Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Algorithms for Communications Systems and their Applications

Algorithms for Communications Systems and their Applications

Nevio Benvenuto
University of Padova, Italy

Giovanni Cherubini
IBM Zurich Research Laboratory, Switzerland



Copyright © 2002

John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester,
West Sussex PO19 8SQ, England

Telephone (+44) 1243 779777

Email (for orders and customer service enquiries): cs-books@wiley.co.uk
Visit our Home Page on www.wileyeurope.com or www.wiley.com

Reprinted with corrections March 2003

All Rights Reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except under the terms of the Copyright, Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd, 90 Tottenham Court Road, London W1T 4LP, UK, without the permission in writing of the Publisher. Requests to the Publisher should be addressed to the Permissions Department, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England, or emailed to permreq@wiley.co.uk, or faxed to (+44) 1243 770571.

Neither the author(s) nor John Wiley & Sons, Ltd accept any responsibility or liability for loss or damage occasioned to any person or property through using the material, instructions methods or ideas contained herein, or acting or refraining from acting as a result of such use. The author(s) and Publisher expressly disclaim all implied warranties, including merchantability of fitness for any particular purpose.

Designations used by companies to distinguish their products are often claimed as trademarks. In all instances where John Wiley & Sons is aware of a claim, the product names appear in initial capital or capital letters. Readers, however, should contact the appropriate companies for more complete information regarding trademarks and registration.

Other Wiley Editorial Offices

John Wiley & Sons Inc., 111 River Street, Hoboken, NJ 07030, USA

Jossey-Bass, 989 Market Street, San Francisco, CA 94103-1741, USA

Wiley-VCH Verlag GmbH, Boschstr. 12, D-69469 Weinheim, Germany

John Wiley & Sons Australia Ltd, 33 Park Road, Milton, Queensland 4064, Australia

John Wiley & Sons (Asia) Pte Ltd, 2 Clementi Loop #02-01, Jin Xing Distripark, Singapore 129809

John Wiley & Sons Canada Ltd, 22 Worcester Road, Etobicoke, Ontario, Canada M9W 1L1

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

ISBN 0-470-84389-6

Produced from LaTeX files supplied by the authors, processed by Laserwords Private Limited, Chennai, India
Printed and bound in Great Britain by Biddles Ltd, Guildford and King's Lynn
This book is printed on acid-free paper responsibly manufactured from sustainable forestry in which at least two trees are planted for each one used for paper production.

To Adriana, and to Antonio, Claudia, and Mariuccia

Contents

Preface **xxix**

Acknowledgements **xxxi**

1 Elements of signal theory **1**

1.1 Signal space . . . 1
Properties of a linear space . . . 1
Inner product . . . 3
1.2 Discrete signal representation . . . 4
The principle of orthogonality . . . 6
Signal representation . . . 6
Gram–Schmidt orthonormalization procedure . . . 8
1.3 Continuous-time linear systems . . . 13
1.4 Discrete-time linear systems . . . 17
Discrete Fourier transform (DFT) . . . 19
The DFT operator . . . 20
Circular and linear convolution via DFT . . . 21
Convolution by the overlap-save method . . . 23
IIR and FIR filters . . . 25
1.5 Signal bandwidth . . . 28
The sampling theorem . . . 30
Heaviside conditions for the absence of signal distortion . . . 32
1.6 Passband signals . . . 33
Complex representation . . . 33
Relation between x and $x^{(bb)}$ . . . 34
Baseband equivalent of a transformation . . . 42
Envelope and instantaneous phase and frequency . . . 43
1.7 Second-order analysis of random processes . . . 44
1.7.1 Correlation . . . 45
Properties of the autocorrelation function . . . 46
1.7.2 Power spectral density . . . 46
Spectral lines in the PSD . . . 47
Cross-power spectral density . . . 48

Properties of the PSD 48
PSD of processes through linear transformations 49
PSD of processes through filtering 50
1.7.3 PSD of discrete-time random processes 50
Spectral lines in the PSD 51
PSD of processes through filtering 52
Minimum-phase spectral factorization 53
1.7.4 PSD of passband processes 54
PSD of the quadrature components of a random process 54
Cyclostationary processes 56
1.8 The autocorrelation matrix 63
Definition 63
Properties 63
Eigenvalues 63
Other properties 64
Eigenvalue analysis for Hermitian matrices 65
1.9 Examples of random processes 67
1.10 Matched filter 73
Matched filter in the presence of white noise 74
1.11 Ergodic random processes 76
1.11.1 Mean value estimators 78
Rectangular window 80
Exponential filter 81
General window 82
1.11.2 Correlation estimators 82
Unbiased estimate 82
Biased estimate 83
1.11.3 Power spectral density estimators 84
Periodogram or instantaneous spectrum 84
Welch periodogram 85
Blackman and Tukey correlogram 86
Windowing and window closing 86
1.12 Parametric models of random processes 90
ARMA(p, q) model 90
MA(q) model 91
AR(N) model 91
Spectral factorization of an AR(N) model 94
Whitening filter 94
Relation between ARMA, MA and AR models 94
1.12.1 Autocorrelation of AR processes 96
1.12.2 Spectral estimation of an AR(N) process 98
Some useful relations 99
AR model of sinusoidal processes 101
1.13 Guide to the bibliography 102

Bibliography . . . 103
Appendices . . . 104
1.A Multirate systems . . . 104
1.A.1 Fundamentals . . . 104
1.A.2 Decimation . . . 106
1.A.3 Interpolation . . . 109
1.A.4 Decimator filter . . . 110
1.A.5 Interpolator filter . . . 112
1.A.6 Rate conversion . . . 113
1.A.7 Time interpolation . . . 116
Linear interpolation . . . 116
Quadratic interpolation . . . 118
1.A.8 The noble identities . . . 118
1.A.9 The polyphase representation . . . 119
Efficient implementations . . . 120
1.B Generation of Gaussian noise . . . 127

2 The Wiener filter and linear prediction 129
2.1 The Wiener filter . . . 129
Matrix formulation . . . 130
Determination of the optimum filter coefficients . . . 132
The principle of orthogonality . . . 134
Expression of the minimum mean-square error . . . 135
Characterization of the cost function surface . . . 135
The Wiener filter in the z-domain . . . 136
2.2 Linear prediction . . . 140
Forward linear predictor . . . 140
Optimum predictor coefficients . . . 141
Forward "prediction error filter" . . . 142
Relation between linear prediction and AR models . . . 143
First and second order solutions . . . 144
2.2.1 The Levinson–Durbin algorithm . . . 145
Lattice filters . . . 146
2.2.2 The Delsarte–Genin algorithm . . . 147
2.3 The least squares (LS) method . . . 148
Data windowing . . . 149
Matrix formulation . . . 149
Correlation matrix $\mathbf{\Phi}$ . . . 150
Determination of the optimum filter coefficients . . . 150
2.3.1 The principle of orthogonality . . . 151
Expressions of the minimum cost function . . . 152
The normal equation using the $\mathbf{T}$ matrix . . . 152
Geometric interpretation: the projection operator . . . 153
2.3.2 Solutions to the LS problem . . . 154
Singular value decomposition of $\mathbf{T}$ . . . 155
Minimum norm solution . . . 157

Bibliography . . . 158
Appendices . . . 159
2.A The estimation problem . . . 159
The estimation problem for random variables . . . 159
MMSE estimation . . . 159
Extension to multiple observations . . . 160
MMSE linear estimation . . . 161
MMSE linear estimation for random vectors . . . 162

3 Adaptive transversal filters 165
3.1 Adaptive transversal filter: MSE criterion . . . 166
3.1.1 Steepest descent or gradient algorithm . . . 166
Stability of the steepest descent algorithm . . . 168
Conditions for convergence . . . 169
Choice of the adaptation gain for fastest convergence . . . 170
Transient behavior of the MSE . . . 171
3.1.2 The least mean-square (LMS) algorithm . . . 173
Implementation . . . 173
Computational complexity . . . 175
Canonical model . . . 175
Conditions for convergence . . . 175
3.1.3 Convergence analysis of the LMS algorithm . . . 177
Convergence of the mean . . . 178
Convergence in the mean-square sense (real scalar case) . . . 179
Convergence in the mean-square sense (general case) . . . 180
Basic results . . . 183
Observations . . . 184
Final remarks . . . 186
3.1.4 Other versions of the LMS algorithm . . . 186
Leaky LMS . . . 187
Sign algorithm . . . 187
Sigmoidal algorithm . . . 188
Normalized LMS . . . 189
Variable adaptation gain . . . 189
LMS for lattice filters . . . 191
3.1.5 Example of application: the predictor . . . 191
3.2 The recursive least squares (RLS) algorithm . . . 197
Normal equation . . . 198
Derivation of the RLS algorithm . . . 199
Initialization of the RLS algorithm . . . 201
Recursive form of $\mathcal{E}_{min}$ . . . 202
Convergence of the RLS algorithm . . . 203
Computational complexity of the RLS algorithm . . . 203
Example of application: the predictor . . . 203
3.3 Fast recursive algorithms . . . 204
3.3.1 Comparison of the various algorithms . . . 205

3.4 Block adaptive algorithms in the frequency domain 205
3.4.1 Block LMS algorithm in the frequency domain: the basic scheme 206
Computational complexity of the block LMS algorithm via FFT 206
3.4.2 Block LMS algorithm in the frequency domain: the FLMS algorithm 207
Computational complexity of the FLMS algorithm 209
Convergence in the mean of the coefficients for the FLMS algorithm 211
3.5 LMS algorithm in a transformed domain 211
3.5.1 Basic scheme 212
On the speed of convergence 214
3.5.2 Normalized FLMS algorithm 214
3.5.3 LMS algorithm in the frequency domain 214
3.5.4 LMS algorithm in the DCT domain 215
3.5.5 General observations 216
3.6 Examples of application 216
3.6.1 System identification 216
Linear case 217
Finite alphabet case 220
3.6.2 Adaptive cancellation of interfering signals 221
General solution 222
3.6.3 Cancellation of a sinusoidal interferer with known frequency . . 224
3.6.4 Disturbance cancellation for speech signals 224
3.6.5 Echo cancellation in subscriber loops 225
3.6.6 Adaptive antenna arrays 226
3.6.7 Cancellation of a periodic interfering signal 227
Bibliography 229
Appendices 233
3.A PN sequences 233
Maximal-length sequences 233
CAZAC sequences 235
Gold sequences 236
3.B Identification of a FIR system by PN sequences 239
3.B.1 Correlation method 239
Signal-to-estimation error ratio 241
3.B.2 Methods in the frequency domain 242
System identification in the absence of noise 242
System identification in the presence of noise 243
3.B.3 The LS method 244
Formulation using the data matrix 246
Computation of the signal-to-estimation error ratio 246
3.B.4 The LMMSE method 249
3.B.5 Identification of a continuous-time system 251

4 Transmission media **255**
4.1 Electrical characterization of a transmission system 255
Simplified scheme of a transmission system 255
Characterization of an active device 257
Conditions for the absence of signal distortion 259
Characterization of a 2-port network 259
Measurement of signal power 262
4.2 Noise generated by electrical devices and networks 263
Thermal noise 263
Shot noise 265
Noise in diodes and transistors 265
Noise temperature of a two-terminal device 265
Noise temperature of a 2-port network 266
Equivalent-noise models 267
Noise figure of a 2-port network 268
Cascade of 2-port networks 270
4.3 Signal-to-noise ratio (SNR) 272
SNR for a two-terminal device 272
SNR for a 2-port network 273
Relation between noise figure and SNR 274
4.4 Transmission lines 275
4.4.1 Fundamentals of transmission line theory 275
Ideal transmission line 276
Non-ideal transmission line 279
Frequency response 279
Conditions for the absence of signal distortion 282
Impulse response of a non-ideal transmission line 282
Secondary constants of some transmission lines 283
4.4.2 Cross-talk 286
Near-end cross-talk 288
Far-end cross-talk 290
4.5 Optical fibers 291
Description of a fiber-optic transmission system 292
4.6 Radio links 294
4.6.1 Frequency ranges for radio transmission 295
Radiation masks 296
4.6.2 Narrowband radio channel model 296
Equivalent circuit at the receiver 299
Multipath 299
4.6.3 Doppler shift 303
4.6.4 Propagation of wideband signals 305
Channel parameters in the presence of multipath 307
Statistical description of fading channels 307
4.6.5 Continuous-time channel model 309
Power delay profile 310

Doppler spectrum . . . 311
Doppler spectrum models . . . 313
Shadowing . . . 313
Final remarks . . . 314
4.6.6 Discrete-time model for fading channels . . . 315
Generation of a process with a pre-assigned spectrum . . . 316
4.7 Telephone channel . . . 318
4.7.1 Characteristics . . . 318
Linear distortion . . . 319
Noise sources . . . 319
Non-linear distortion . . . 319
Frequency offset . . . 319
Phase jitter . . . 321
Echo . . . 321
4.8 Transmission channel: general model . . . 322
Power amplifier (HPA) . . . 322
Transmission medium . . . 326
Additive noise . . . 326
Phase noise . . . 326
Bibliography . . . 328

5 Digital representation of waveforms 331
5.1 Analog and digital access . . . 331
5.1.1 Digital representation of speech . . . 332
Some waveforms . . . 332
Speech coding . . . 337
The interpolator filter as a holder . . . 338
Sizing of the binary channel parameters . . . 340
5.1.2 Coding techniques and applications . . . 341
5.2 Instantaneous quantization . . . 344
5.2.1 Parameters of a quantizer . . . 344
5.2.2 Uniform quantizers . . . 346
Quantization error . . . 347
Relation between Δ, $\mathfrak{b}$ and τ_{sat} . . . 350
Statistical description of the quantization noise . . . 350
Statistical power of the quantization error . . . 352
Design of a uniform quantizer . . . 353
Signal-to-quantization error ratio . . . 354
Implementations of uniform PCM encoders . . . 357
5.3 Non-uniform quantizers . . . 358
Three examples of implementation . . . 359
5.3.1 Companding techniques . . . 360
Signal-to-quantization error ratio . . . 364
Digital compression . . . 365
Signal-to-quantization noise ratio mask . . . 366

5.3.2 Optimum quantizer in the MSE sense . . . 366
Max algorithm . . . 369
Lloyd algorithm . . . 370
Expression of Λ_q for a very fine quantization . . . 371
Performance of non-uniform quantizers . . . 374
5.4 Adaptive quantization . . . 377
General scheme . . . 377
5.4.1 Feedforward adaptive quantizer . . . 379
Performance . . . 380
5.4.2 Feedback adaptive quantizers . . . 381
Estimate of $\sigma_s(k)$ . . . 382
5.5 Differential coding (DPCM) . . . 385
5.5.1 Configuration with feedback quantizer . . . 386
5.5.2 Alternative configuration . . . 389
5.5.3 Expression of the optimum coefficients . . . 391
Effects due to the presence of the quantizer . . . 392
5.5.4 Adaptive predictors . . . 393
Adaptive feedforward predictors . . . 394
Sequential adaptive feedback predictors . . . 394
Performance . . . 398
5.5.5 Alternative structures for the predictor . . . 398
All-pole predictor . . . 398
All-zero predictor . . . 399
Pole-zero predictor . . . 399
Pitch predictor . . . 400
APC . . . 401
5.6 Delta modulation . . . 404
5.6.1 Oversampling and quantization error . . . 404
5.6.2 Linear delta modulation (LDM) . . . 407
LDM implementation . . . 408
Choice of system parameters . . . 408
5.6.3 Adaptive delta modulation (ADM) . . . 410
Continuously variable slope delta modulation (CVSDM) . . . 411
ADM with second-order predictors . . . 412
5.6.4 PCM encoder via LDM . . . 412
5.6.5 Sigma delta modulation (ΣDM) . . . 413
5.7 Coding by modeling . . . 413
Vocoder or LPC . . . 414
RPE coding . . . 415
CELP coding . . . 416
Multipulse coding . . . 417
5.8 Vector quantization (VQ) . . . 417
5.8.1 Characterization of VQ . . . 418
Parameters determining VQ performance . . . 418
Comparison between VQ and scalar quantization . . . 420

5.8.2 Optimum quantization . . . 421
Generalized Lloyd algorithm . . . 422
5.8.3 LBG algorithm . . . 424
Choice of the initial codebook . . . 425
Description of the LBG algorithm with splitting procedure . . . 426
Selection of the training sequence . . . 426
5.8.4 Variants of VQ . . . 429
Tree search VQ . . . 429
Multistage VQ . . . 430
Product code VQ . . . 430
5.9 Other coding techniques . . . 432
Adaptive transform coding (ATC) . . . 433
Sub-band coding (SBC) . . . 433
5.10 Source coding . . . 433
5.11 Speech and audio standards . . . 434
Bibliography . . . 435

6 Modulation theory 437
6.1 Theory of optimum detection . . . 437
Statistics of the random variables $\{w_i\}$ . . . 439
Sufficient statistics . . . 440
Decision criterion . . . 440
Theorem of irrelevance . . . 442
Implementations of the maximum likelihood criterion . . . 445
Error probability . . . 447
6.1.1 Examples of binary signalling . . . 449
Antipodal signals ($\rho = -1$) . . . 449
Orthogonal signals ($\rho = 0$) . . . 450
Binary FSK . . . 452
6.1.2 Limits on the probability of error . . . 454
Upper limit . . . 454
Lower limit . . . 455
6.2 Simplified model of a transmission system and definition of binary channel . . . 456
Parameters of a transmission system . . . 458
Relations among parameters . . . 459
6.3 Pulse amplitude modulation (PAM) . . . 461
6.4 Phase-shift keying (PSK) . . . 465
Binary PSK (BPSK) . . . 470
Quadrature PSK (QPSK) . . . 472
6.5 Differential PSK (DPSK) . . . 474
6.5.1 Error probability for an M-DPSK system . . . 475
6.5.2 Differential encoding and coherent demodulation . . . 477
Binary case ($M = 2$, differentially encoded BPSK) . . . 477
Multilevel case . . . 478

6.6 AM-PM or quadrature amplitude modulation (QAM) 480
Comparison between PSK and QAM 485
6.7 Modulation methods using orthogonal and biorthogonal signals 486
6.7.1 Modulation with orthogonal signals 486
Probability of error 489
Limit of the probability of error for M increasing to infinity . 492
6.7.2 Modulation with biorthogonal signals 493
Probability of error 494
6.8 Binary sequences and coding 496
Optimum receiver 498
6.9 Comparison between coherent modulation methods 499
Trade-offs for QAM systems 502
Comparison of modulation methods 502
6.10 Limits imposed by information theory 503
Capacity of a system using amplitude modulation 504
Coding strategies depending on the signal-to-noise ratio 506
Coding gain 508
Cut-off rate 509
6.11 Optimum receivers for signals with random phase 509
ML criterion 510
Implementation of a non-coherent ML receiver 512
Error probability for a non-coherent binary FSK system 516
Performance comparison of binary systems 519
6.12 Binary modulation systems in the presence of flat fading 520
Diversity 521
6.13 Transmission methods 522
6.13.1 Transmission methods between two users 522
Three methods 523
6.13.2 Channel sharing: deterministic access methods 523
Bibliography 525
Appendices 527
6.A Gaussian distribution function and Marcum function 527
6.A.1 The Q function 527
6.A.2 The Marcum function 529
6.B Gray coding 531
6.C Baseband PPM and PDM 532
Signal-to-noise ratio 532
6.D Walsh codes 536

7 Transmission over dispersive channels 539
7.1 Baseband digital transmission (PAM systems) 539
Transmitter 539
Transmission channel 541
Receiver 542
Power spectral density of a PAM signal 543

7.2 Passband digital transmission (QAM systems) . . . 544
Transmitter . . . 544
Power spectral density of a QAM signal . . . 546
Three equivalent representations of the modulator . . . 547
Coherent receiver . . . 548
7.3 Baseband equivalent model of a QAM system . . . 549
7.3.1 Signal analysis . . . 550
Signal-to-noise ratio . . . 552
7.3.2 Characterization of system elements . . . 553
Transmitter . . . 553
Transmission channel . . . 553
Receiver . . . 555
7.3.3 Intersymbol interference . . . 556
Discrete-time equivalent system . . . 556
Nyquist pulses . . . 559
Eye diagram . . . 562
7.3.4 Performance analysis . . . 565
Symbol error probability in the absence of ISI . . . 565
Matched filter receiver . . . 567
7.4 Carrierless AM/PM (CAP) modulation . . . 568
7.5 Regenerative PCM repeaters . . . 571
7.5.1 PCM signals over a binary channel . . . 571
Linear PCM coding of waveforms . . . 572
Overall system performance . . . 573
7.5.2 Regenerative repeaters . . . 575
Analog transmission . . . 576
Digital transmission . . . 577
Comparison between analog and digital transmission . . . 578
Bibliography . . . 581
Appendices . . . 583
7.A Line codes for PAM systems . . . 583
7.A.1 Line codes . . . 583
Non-return-to-zero (NRZ) format . . . 583
Return-to-zero (RZ) format . . . 584
Biphase (B-ϕ) format . . . 584
Delay modulation or Miller code . . . 585
Block line codes . . . 585
Alternate mark inversion (AMI) . . . 586
7.A.2 Partial response systems . . . 587
The choice of the PR polynomial . . . 590
Symbol detection and error probability . . . 594
Precoding . . . 596
Error probability with precoding . . . 597
Alternative interpretation of PR systems . . . 599

7.B Computation of P_e for some cases of interest . . . 602
7.B.1 P_e in the absence of ISI . . . 602
7.B.2 P_e in the presence of ISI . . . 604
Exhaustive method . . . 604
Gaussian approximation . . . 605
Worst-case limit . . . 605
Saltzberg limit . . . 606
GQR method . . . 607
7.C Coherent PAM-DSB transmission . . . 608
General scheme . . . 608
Transmit signal PSD . . . 609
Signal-to-noise ratio . . . 609
7.D Implementation of a QAM transmitter . . . 611
7.E Simulation of a QAM system . . . 613

8 Channel equalization and symbol detection 619
8.1 Zero-forcing equalizer (LE-ZF) . . . 619
8.2 Linear equalizer (LE) . . . 620
8.2.1 Optimum receiver in the presence of noise and ISI . . . 620
Alternative derivation of the IIR equalizer . . . 622
Signal-to-noise ratio γ . . . 626
8.3 LE with a finite number of coefficients . . . 627
Adaptive LE . . . 628
8.4 Fractionally spaced equalizer (FSE) . . . 630
Adaptive FSE . . . 633
8.5 Decision feedback equalizer (DFE) . . . 635
Adaptive DFE . . . 638
Design of a DFE with a finite number of coefficients . . . 639
Design of a fractionally spaced DFE (FS-DFE) . . . 642
Signal-to-noise ratio γ . . . 644
Remarks . . . 645
8.6 Convergence behavior of adaptive equalizers . . . 645
Adaptive LE . . . 646
Adaptive DFE . . . 648
8.7 LE-ZF with a finite number of coefficients . . . 648
8.8 DFE: alternative configurations . . . 649
DFE-ZF . . . 649
DFE-ZF as a noise predictor . . . 655
DFE as ISI and noise predictor . . . 655
8.9 Benchmark performance for two equalizers . . . 657
Performance comparison . . . 657
Equalizer performance for two channel models . . . 658
8.10 Optimum methods for data detection . . . 659
8.10.1 Maximum likelihood sequence detection . . . 662
Lower limit to error probability using the MLSD criterion . . . 663

The Viterbi algorithm (VA) 663
Computational complexity of the VA 667
8.10.2 Maximum *a posteriori* probability detector 668
Statistical description of a sequential machine 668
The forward-backward algorithm (FBA) 670
Scaling 673
Likelihood function in the absence of ISI 674
Simplified version of the MAP algorithm (Max-Log-MAP) . . 675
Relation between Max-Log-MAP and Log-MAP 677
8.11 Optimum receivers for transmission over dispersive channels 678
Ungerboeck's formulation of the MLSD 680
8.12 Error probability achieved by MLSD 682
Computation of the minimum distance 686
8.13 Reduced state sequence detection 691
Reduced state trellis diagram 691
RSSE algorithm 694
Further simplification: DFSE 695
8.14 Passband equalizers 697
8.14.1 Passband receiver structure 698
Joint optimization of equalizer coefficients and carrier phase offset 700
Adaptive method 701
8.14.2 Efficient implementations of voiceband modems 703
8.15 LE for voiceband modems 705
Detection of the training sequence 706
Computations of the coefficients of a cyclic equalizer 707
Transition from training to data mode 709
Example of application: a simple modem 709
8.16 LE and DFE in the frequency domain with data frames using cyclic prefix 710
8.17 Numerical results obtained by simulations 713
QPSK transmission over a minimum phase channel 713
QPSK transmission over a non-minimum phase channel 715
8-PSK transmission over a minimum phase channel 716
8-PSK transmission over a non-minimum phase channel . . . 716
8.18 Diversity combining techniques 717
Antenna arrays 718
Combining techniques 719
Equalization and diversity 722
Diversity in transmission 722
Bibliography 726
Appendices 731
8.A Calculus of variations and receiver optimization 731
8.A.1 Calculus of variations 731
Linear functional 731
Quadratic functional 732

8.A.2 Receiver optimization . . . 735
8.A.3 Joint optimization of transmitter and receiver . . . 739
8.B DFE design: matrix formulations . . . 741
8.B.1 Method based on correlation sequences . . . 741
8.B.2 Method based on the channel impulse response and i.i.d. symbols . . . 744
8.B.3 Method based on the channel impulse response and any symbol statistic . . . 746
8.B.4 FS-DFE . . . 747
8.C Equalization based on the peak value of ISI . . . 749
8.D Description of a finite state machine (FSM) . . . 751

9 Orthogonal frequency division multiplexing 753
9.1 OFDM systems . . . 753
9.2 Orthogonality conditions . . . 755
Time domain . . . 755
Frequency domain . . . 755
z-transform domain . . . 755
9.3 Efficient implementation of OFDM systems . . . 756
OFDM implementation employing matched filters . . . 757
Orthogonality conditions in terms of the polyphase components . . . 759
OFDM implementation employing a prototype filter . . . 760
9.4 Non-critically sampled filter banks . . . 764
9.5 Examples of OFDM systems . . . 769
Discrete multitone (DMT) . . . 770
Filtered multitone (FMT) . . . 771
Discrete wavelet multitone (DWMT) . . . 771
9.6 Equalization of OFDM systems . . . 773
Interpolator filter and virtual subchannels . . . 773
Equalization of DMT systems . . . 775
Equalization of FMT systems . . . 777
9.7 Synchronization of OFDM systems . . . 779
9.8 Passband OFDM systems . . . 780
Passband DWMT systems . . . 780
Passband DMT and FMT systems . . . 781
Comparison between OFDM and QAM systems . . . 781
9.9 DWMT modulation . . . 782
Transmit and receive filter banks . . . 783
Approximate interchannel interference suppression . . . 786
Perfect interchannel interference suppression . . . 788
Bibliography . . . 793

10 Spread spectrum systems 795
10.1 Spread spectrum techniques . . . 795
10.1.1 Direct sequence systems . . . 795

Classification of CDMA systems . . . 802
Synchronization . . . 804
10.1.2 Frequency hopping systems . . . 804
Classification of FH systems . . . 806
10.2 Applications of spread spectrum systems . . . 807
10.2.1 Anti-jam communications . . . 808
10.2.2 Multiple-access systems . . . 810
10.2.3 Interference rejection . . . 811
10.3 Chip matched filter and rake receiver . . . 811
Number of resolvable rays in a multipath channel . . . 811
Chip matched filter (CMF) . . . 813
10.4 Interference . . . 816
Detection strategies for multiple-access systems . . . 818
10.5 Equalizers for single-user detection . . . 818
Chip equalizer (CE) . . . 818
Symbol equalizer (SE) . . . 819
10.6 Block equalizer for multiuser detection . . . 820
10.7 Maximum likelihood multiuser detector . . . 823
Correlation matrix approach . . . 823
Whitening filter approach . . . 824
Bibliography . . . 824

11 Channel codes **827**
11.1 System model . . . 828
11.2 Block codes . . . 830
11.2.1 Theory of binary codes with group structure . . . 830
Properties . . . 830
Parity check matrix . . . 833
Code generator matrix . . . 836
Decoding of binary parity check codes . . . 837
Cosets . . . 837
Two conceptually simple decoding methods . . . 838
Syndrome decoding . . . 839
11.2.2 Fundamentals of algebra . . . 842
Modulo q arithmetic . . . 843
Polynomials with coefficients from a field . . . 845
The concept of modulo in the arithmetic of polynomials . . . 846
Devices to sum and multiply elements in a finite field . . . 849
Remarks on finite fields . . . 851
Roots of a polynomial . . . 854
Minimum function . . . 857
Methods to determine the minimum function . . . 859
Properties of the minimum function . . . 861
11.2.3 Cyclic codes . . . 862
The algebra of cyclic codes . . . 862

Properties of cyclic codes . . . 864
Encoding method using a shift register of length r . . . 869
Encoding method using a shift register of length k . . . 870
Hard decoding of cyclic codes . . . 871
Hamming codes . . . 872
Burst error detection . . . 875
11.2.4 Simplex cyclic codes . . . 875
Relation to PN sequences . . . 877
11.2.5 BCH codes . . . 878
An alternative method to specify the code polynomials . . . 878
Bose–Chaudhuri–Hocquenhem (BCH) codes . . . 880
Binary BCH codes . . . 883
Reed–Solomon codes . . . 885
Decoding of BCH codes . . . 887
Efficient decoding of BCH codes . . . 891
11.2.6 Performance of block codes . . . 899
11.3 Convolutional codes . . . 900
11.3.1 General description of convolutional codes . . . 903
Parity check matrix . . . 905
Generator matrix . . . 906
Transfer function . . . 907
Catastrophic error propagation . . . 910
11.3.2 Decoding of convolutional codes . . . 912
Interleaving . . . 913
Two decoding models . . . 913
Viterbi algorithm . . . 915
Forward-backward algorithm . . . 915
Sequential decoding . . . 917
11.3.3 Performance of convolutional codes . . . 919
11.4 Concatenated codes . . . 921
Soft-output Viterbi algorithm (SOVA) . . . 921
11.5 Turbo codes . . . 924
Encoding . . . 924
The basic principle of iterative decoding . . . 929
The forward-backward algorithm revisited . . . 930
Iterative decoding . . . 939
Performance evaluation . . . 941
11.6 Iterative detection and decoding . . . 943
11.7 Low-density parity check codes . . . 946
Encoding procedure . . . 948
Decoding algorithm . . . 948
Example of application . . . 953
Performance and coding gain . . . 954
Bibliography . . . 956
Appendices . . . 960

11.A Nonbinary parity check codes . . . 960
Linear codes . . . 961
Parity check matrix . . . 962
Code generator matrix . . . 963
Decoding of nonbinary parity check codes . . . 964
Coset . . . 964
Two conceptually simple decoding methods . . . 965
Syndrome decoding . . . 965

12 Trellis coded modulation **967**
12.1 Linear TCM for one- and two-dimensional signal sets . . . 968
12.1.1 Fundamental elements . . . 968
Basic TCM scheme . . . 970
Example . . . 970
12.1.2 Set partitioning . . . 973
12.1.3 Lattices . . . 975
12.1.4 Assignment of symbols to the transitions in the trellis . . . 980
12.1.5 General structure of the encoder/bit-mapper . . . 985
Computation of d_{free} . . . 987
12.2 Multidimensional TCM . . . 990
Encoding . . . 990
Decoding . . . 993
12.3 Rotationally invariant TCM schemes . . . 995
Bibliography . . . 996

13 Precoding and coding techniques for dispersive channels **999**
13.1 Capacity of a dispersive channel . . . 999
13.2 Techniques to achieve capacity . . . 1002
Bit loading for OFDM . . . 1002
Discrete-time model of a single carrier system . . . 1003
Achieving capacity with a single carrier system . . . 1007
13.3 Precoding and coding for dispersive channels . . . 1008
13.3.1 Tomlinson–Harashima (TH) precoding . . . 1009
13.3.2 TH precoding and TCM . . . 1012
13.3.3 Flexible precoding . . . 1018
Bibliography . . . 1025

14 Synchronization **1027**
14.1 The problem of synchronization for QAM systems . . . 1027
14.2 The phase-locked loop . . . 1029
14.2.1 PLL baseband model . . . 1031
Linear approximation . . . 1032
14.2.2 Analysis of the PLL in the presence of additive noise . . . 1034
Noise analysis using the linearity assumption . . . 1035
14.2.3 Analysis of a second-order PLL . . . 1036
14.3 Costas loop . . . 1040

14.3.1 PAM signals . . . 1040
14.3.2 QAM signals . . . 1042
14.4 The optimum receiver . . . 1044
Timing recovery . . . 1046
Carrier phase recovery . . . 1050
14.5 Algorithms for timing and carrier phase recovery . . . 1051
14.5.1 ML criterion . . . 1051
Assumption of slow time varying channel . . . 1051
14.5.2 Taxonomy of algorithms using the ML criterion . . . 1051
Feedback estimators . . . 1053
Early-late estimators . . . 1055
14.5.3 Timing estimators . . . 1055
Non-data aided . . . 1055
Non-data aided via spectral estimation . . . 1057
Data-aided and data-directed . . . 1059
Data- and phase-directed with feedback: differentiator scheme . . . 1062
Data- and phase-directed with feedback: Mueller & Muller scheme . . . 1064
Non-data aided with feedback . . . 1065
14.5.4 Phasor estimators . . . 1066
Data- and timing-directed . . . 1066
Non-data aided for M-PSK signals . . . 1066
Data- and timing-directed with feedback . . . 1067
14.6 Algorithms for carrier frequency recovery . . . 1068
14.6.1 Frequency offset estimators . . . 1069
Non-data aided . . . 1069
Non-data aided and timing-independent with feedback . . . 1071
Non-data aided and timing-directed with feedback . . . 1071
14.6.2 Estimators operating at the modulation rate . . . 1072
Data-aided and data-directed . . . 1073
Non-data aided for M-PSK . . . 1073
14.7 Second-order digital PLL . . . 1074
14.8 Synchronization in spread spectrum systems . . . 1074
14.8.1 The transmission system . . . 1074
Transmitter . . . 1074
Optimum receiver . . . 1075
14.8.2 Timing estimators with feedback . . . 1076
Non-data aided: non-coherent DLL . . . 1076
Non-data aided MCTL . . . 1077
Data- and phase-directed: coherent DLL . . . 1077
Bibliography . . . 1081

15 Self-training equalization **1083**
15.1 Problem definition and fundamentals . . . 1083
Minimization of a special function . . . 1086

15.2 Three algorithms for PAM systems 1090
The Sato algorithm 1090
Benveniste–Goursat algorithm 1091
Stop-and-go algorithm 1092
Remarks 1092
15.3 The contour algorithm for PAM systems 1093
Simplified realization of the contour algorithm 1095
15.4 Self-training equalization for partial response systems 1096
The Sato algorithm for partial response systems 1096
Contour algorithm for partial response systems 1098
15.5 Self-training equalization for QAM systems 1100
The Sato algorithm for QAM systems 1100
15.5.1 Constant modulus algorithm 1101
The contour algorithm for QAM systems 1102
Joint contour algorithm and carrier phase tracking 1104
15.6 Examples of applications 1106
Bibliography 1111
Appendices 1113
15.A On the convergence of the contour algorithm 1113

16 Applications of interference cancellation 1115
16.1 Echo and near–end cross-talk cancellation for PAM systems 1116
Cross-talk cancellation and full duplex transmission 1117
Polyphase structure of the canceller 1118
Canceller at symbol rate 1119
Adaptive canceller 1120
Canceller structure with distributed arithmetic 1121
16.2 Echo cancellation for QAM systems 1124
16.3 Echo cancellation for OFDM systems 1128
16.4 Multiuser detection for VDSL 1131
16.4.1 Upstream power back-off 1136
16.4.2 Comparison of PBO methods 1137
Bibliography 1142

17 Wired and wireless network technologies 1145
17.1 Wired network technologies 1145
17.1.1 Transmission over unshielded twisted pairs in the customer service area 1145
Modem 1145
Digital subscriber line 1146
17.1.2 High speed transmission over unshielded twisted pairs in local area networks 1152
17.1.3 Hybrid fiber/coaxial cable networks 1156
Ranging and power adjustment for uplink transmission 1158

17.2 Wireless network technologies 1161
17.2.1 Wireless local area networks 1162
Medium access control protocols 1164
17.2.2 MMDS and LMDS 1165
Bibliography 1167
Appendices 1170
17.A Standards for wireless systems 1170
17.A.1 General observations 1171
Wireless systems 1171
Modulation techniques 1171
Parameters of the modulator 1171
Cells in a wireless system 1172
17.A.2 GSM standard 1172
System characteristics 1172
Radio subsystem 1175
GSM-EDGE 1177
17.A.3 IS-136 standard 1177
17.A.4 JDC standard 1180
17.A.5 IS-95 standard 1180
17.A.6 DECT standard 1182
17.A.7 HIPERLAN standard 1185

18 Modulation techniques for wireless systems 1189
18.1 Analog front-end architectures 1189
Conventional superheterodyne receiver 1189
Alternative architectures 1190
Direct conversion receiver 1190
Single conversion to low-IF 1191
Double conversion and wideband IF 1192
18.2 Three non-coherent receivers for phase modulation systems 1192
18.2.1 Baseband differential detector 1192
18.2.2 IF-band (1 Bit) differential detector (1BDD) 1194
Performance of M-DPSK 1196
18.2.3 FM discriminator with integrate and dump filter (LDI) 1197
18.3 Variants of QPSK 1198
18.3.1 Basic schemes 1198
QPSK 1198
Offset QPSK or staggered QPSK 1200
Differential QPSK (DQPSK) 1201
$\pi/4$-DQPSK 1202
18.3.2 Implementations 1203
QPSK, OQPSK, and DQPSK modulators 1203
$\pi/4$-DQPSK modulators 1203
18.4 Frequency shift keying (FSK) 1207
18.4.1 Power spectrum of M-FSK 1207

Power spectrum of non-coherent binary FSK 1208
Power spectrum of coherent M-FSK 1209
18.4.2 FSK receivers and corresponding performance 1212
Coherent demodulator 1212
Non-coherent demodulator 1213
Limiter-discriminator FM demodulator 1213
18.5 Minimum shift keying (MSK) 1214
18.5.1 Power spectrum of continuous-phase FSK (CPFSK) 1217
18.5.2 The MSK signal viewed from two perspectives 1217
Phase of an MSK signal 1217
MSK as binary CPFSK 1219
MSK as OQPSK 1220
Complex notation of an MSK signal 1222
18.5.3 Implementations of an MSK scheme 1224
18.5.4 Performance of MSK demodulators 1224
MSK with differential precoding 1227
18.5.5 Remarks on spectral containment 1228
18.6 Gaussian MSK (GMSK) 1229
18.6.1 GMSK via CPFSK 1229
18.6.2 Power spectrum of GMSK 1231
18.6.3 Implementation of a GMSK scheme 1234
Configuration I 1234
Configuration II 1234
Configuration III 1236
18.6.4 Linear approximation of a GMSK signal 1238
Performance of GMSK demodulators 1238
Performance of a GSM receiver in the presence of multipath 1243
Bibliography 1244
Appendices 1246
18.A Continuous phase modulation (CPM) 1246
Alternative definition of CPM 1246
Advantages of CPM 1248

19 Design of high speed transmission systems over unshielded twisted pair cables 1249
19.1 Design of a quaternary partial response class-IV system for data transmission at 125 Mbit/s 1249
Analog filter design 1249
Received signal and adaptive gain control 1250
Near-end cross-talk cancellation 1251
Decorrelation filter 1251
Adaptive equalizer 1252
Compensation of the timing phase drift 1252

Adaptive equalizer coefficient adaptation 1253
Convergence behavior of the various algorithms 1253
19.1.1 VLSI implementation 1255
Adaptive digital NEXT canceller 1255
Adaptive digital equalizer 1258
Timing control 1261
Viterbi detector 1262
19.2 Design of a dual duplex transmission system at 100 Mbit/s 1263
Dual duplex transmission 1263
Physical layer control 1265
Coding and decoding 1266
19.2.1 Signal processing functions 1269
The 100BASE-T2 transmitter 1269
The 100BASE-T2 receiver 1270
Computational complexity of digital receive filters 1272
Bibliography 1273
Appendices 1274
19.A Interference suppression 1274

Index **1277**

Preface

The motivation for this book is twofold. On the one hand, we provide a didactic tool to students of communications systems. On the other hand, we present a discussion of fundamental algorithms and structures for telecommunication technologies. The contents reflect our experience in teaching courses on Algorithms for Telecommunications at the University of Padova, Italy, as well as our professional experience acquired in industrial research laboratories.

The text explains the procedures for solving problems posed by the design of systems for reliable communications over wired or wireless channels. In particular, we focus on fundamental developments in the field in order to provide the reader with the necessary insight to design essential elements of various communications systems.

The book is divided into nineteen chapters. We briefly indicate four tracks corresponding to specific areas and course work offered.

Track 1. Track 1 includes the basic elements for a first course on telecommunications, which we regard as an introduction to the remaining tracks. It covers Chapter 1, which recalls fundamental concepts on signals and random processes, with an emphasis on second-order statistical descriptions. A discussion of the characteristics of transmission media follows in Chapter 4. In this track we focus on the description of noise in electronic devices and on the laws of propagation in transmission lines and radio channels. The representation of waveforms by sequences of binary symbols is treated in Chapter 5; for a first course it is suggested that emphasis be placed on PCM. Next, Chapter 6 examines the fundamental principles of a digital transmission system, where a sequence of information symbols is sent over a transmission channel. We refer to Shannon theorem to establish the maximum bit rate that can be transmitted reliably over a noisy channel. Signal dispersion caused by a transmission channel is then analyzed in Chapter 7. Examples of elementary and practical implementations of transmission systems are presented, together with a brief introduction to computer simulations. The first three sections of Chapter 11, where we introduce methods for increasing transmission reliability by exploiting the redundancy added to the information bits, conclude the first track.

Track 2. Track 2, which is an extension of Track 1, focuses on modulation techniques. First, parametric models of random processes are analyzed in Chapter 1. The Wiener filter and the linear prediction theory, which constitute fundamental elements for receiver design, are dealt with in Chapter 2. Chapter 3 lists iterative methods to achieve the objectives stated

in Chapter 2, as well as various applications of the Wiener filter, for example channel identification and interference cancellation. These applications are further developed in the first two sections of Chapter 16.

In the first part of Chapter 8, channel equalization is examined as a further application of the Wiener filter. In the second part of the chapter, more sophisticated methods of equalization and symbol detection, which rely on the Viterbi algorithm and on the forward-backward algorithm, are analyzed. Initially single-carrier modulation systems are considered. In Chapter 9, we introduce multicarrier modulation techniques, which are preferable for transmission over very dispersive channels and/or applications that require flexibility in spectral allocation. In Chapter 10 spread spectrum systems are examined, with emphasis to applications for simultaneous channel access by several users that share a wideband channel. The inherent narrowband interference rejection capabilities of spread spectrum systems, as well as their implementations, are also discussed. This is followed by Chapter 18, which illustrates specific modulation techniques developed for mobile radio applications.

Track 3. We observe the trend towards implementing transceiver functions using digital signal processors. Therefore the algorithmic aspects of a transmission system are becoming increasingly important. Hardware devices are assigned wherever possible only the functions of analog front-end, fixed filtering, and digital-to-analog and analog-to-digital conversion. This approach enhances the flexibility of transceivers, which can be utilized for more than one transmission standard, and considerably reduces development time.

In line with the above considerations, Track 3 begins with a review of Chapters 2 and 3, which illustrate the fundamental principles of transmission system design, and of Chapter 8, which investigates individual building blocks for channel equalization and symbol detection. The assumption that the transmission channel characteristics are known a priori is removed in Chapter 15, where blind equalization techniques are discussed. Channel coding techniques to improve the reliability of transmission are investigated in depth in Chapters 11 and 12. A further method to mitigate channel dispersion is precoding. The operations of systems that employ joint precoding and channel coding are explained in Chapter 13. Because of electromagnetic coupling, the desired signal at the receiver is often disturbed by other transmissions taking place simultaneously. Cancellation techniques to suppress interference signals are treated in Chapter 16.

Track 4. Track 4 addresses various challenges encountered in designing wired and wireless communications systems. The elements introduced in Chapters 2 and 3, as well as the algorithms introduced in Chapter 8, are essential for this track. The principles of multicarrier and spread spectrum modulation techniques, which are increasingly being adopted in communications systems, are investigated in depth in Chapters 9 and 10, respectively. The design of the receiver front-end, as well as various methods for timing and carrier recovery, are dealt with in Chapter 14. Applications of interference cancellation and multi-user detection are addressed in Chapter 16. An overview of wired and wireless access technologies appears in Chapter 17, and specific examples of system design are given in Chapters 18 and 19.

Acknowledgements

We gratefully acknowledge all who have made the realization of this book possible. In particular, the editing of the various chapters would never have been completed without the contributions of numerous students in our courses on Algorithms for Telecommunications. Although space limitations preclude mentioning them all by name, we nevertheless express our sincere gratitude.

We also thank Christian Bolis and Chiara Paci for their support in developing the software for the book, Charlotte Bolliger and Lilli M. Pavka for their assistance in administering the project, and Urs Bitterli and Darja Kropaci for their help with the graphics editing. For text processing of the Italian version, the contribution of Barbara Sicoli was indispensable; our thanks also go to Jane Frankenfield Zanin for her help in translating the text into English.

We are pleased to thank the following colleagues for their invaluable assistance throughout the revision of the book: Antonio Assalini, Paola Bisaglia, Alberto Bononi, Giancarlo Calvagno, Giulio Colavolpe, Roberto Corvaja, Elena Costa, Andrea Galtarossa, Antonio Mian, Carlo Monti, Ezio Obetti, Riccardo Rahely, Roberto Rinaldo, Antonio Salloum, Fortunato Santucci, Andrea Scaggiante, Giovanna Sostrato, Stefano Tomasin, and Luciano Tomba. We gratefully acknowledge our colleague and mentor Jack Wolf for letting us include his lecture notes in the chapter on channel codes. A special acknowledgment goes to our colleagues Werner Bux and Evangelos Eleftheriou of the IBM Zurich Research Laboratory, and Silvano Pupolin of the University of Padua, for their continuing support.

Nevio Benvenuto

Giovanni Cherubini

To make the reading of the adopted symbols easier, a table containing the Greek alphabet is included.

The Greek alphabet

α	A	alpha	ν	N	nu
β	B	beta	ξ	Ξ	xi
γ	Γ	gamma	o	O	omicron
δ	Δ	delta	π	Π	pi
ϵ, ε	E	epsilon	ρ, ϱ	P	rho
ζ	Z	zeta	σ, ς	$\sum$	sigma
η	H	eta	τ	T	tau
θ, ϑ	Θ	theta	υ	Y	upsilon
ι	I	iota	ϕ, φ	Φ	phi
κ	K	kappa	χ	X	chi
λ	Λ	lambda	ψ	Ψ	psi
μ	M	mu	ω	Ω	omega

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 1

Elements of signal theory

In the present chapter we recall fundamental concepts of signal theory and random processes. A majority of readers will simply find this chapter a review of known principles, while others will find it a useful incentive for further in-depth study, for which we recommend the items in the bibliography. In any event, we will begin with the definition of signal space and its discrete representation, then move to the study of discrete-time linear systems (discrete Fourier transforms, IIR and FIR impulse responses) and signals (complex representation of passband signals and the baseband equivalent). We will conclude with the study of random processes, with emphasis on the statistical estimation of first- and second-order ergodic processes (periodogram, correlogram, ARMA, MA and especially AR models).

1.1 Signal space

Definition 1.1
A *linear space* is a set of elements called vectors, together with two operators defined over the elements of the set, the sum between vectors and the multiplication of a vector by a scalar.

The Euclidean space is an example of linear space in which the sum of two vectors coincides with the vector obtained by adding the individual components, and the product of a vector by a scalar coincides with the vector obtained by multiplying each component for that scalar. In our case of particular interest is the set of complex vectors, i.e., those with complex-valued components, in an Euclidean space.

Properties of a linear space

Let $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$ and $\mathbf{0}$ be elements of a *linear space*, and α and β be complex numbers (scalars).

1. *Addition is commutative*

$$\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x} \tag{1.1}$$

2. *Addition is associative*

$$\mathbf{x} + (\mathbf{y} + \mathbf{z}) = (\mathbf{x} + \mathbf{y}) + \mathbf{z} \tag{1.2}$$

3. *There exists a unique vector* $\mathbf{0}$*, called null, such that*

$$\mathbf{0} + \mathbf{x} = \mathbf{x} \tag{1.3}$$

4. *For each* $\mathbf{x}$*, there is a unique vector* $-\mathbf{x}$*, called additive inverse, such that*

$$\mathbf{x} + (-\mathbf{x}) = \mathbf{0} \tag{1.4}$$

5. *Multiplication by scalars is associative*

$$\alpha(\beta\mathbf{x}) = (\alpha\beta)\mathbf{x} \tag{1.5}$$

In particular, we have

$$1\mathbf{x} = \mathbf{x} \qquad 0\mathbf{x} = \mathbf{0} \tag{1.6}$$

6. *Distributive laws*

$$\alpha(\mathbf{x} + \mathbf{y}) = \alpha\mathbf{x} + \alpha\mathbf{y} \tag{1.7}$$

$$(\alpha + \beta)\mathbf{x} = \alpha\mathbf{x} + \beta\mathbf{x} \tag{1.8}$$

A geometrical interpretation of the two elementary operations in a two-dimensional Euclidean space is given in Figure 1.1.

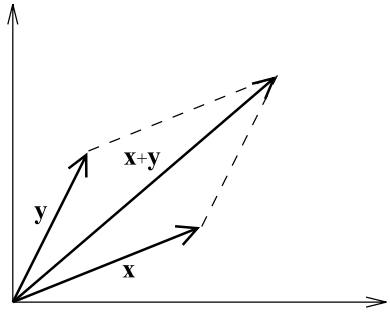
As previously mentioned, the Euclidean space is an example of a *linear space*. Two other examples of linear spaces are: the *discrete-time signal space* (an Euclidean space with infinite dimensions), whose elements are the signals

$$\{x(kT_c)\} \qquad k \text{ integer} \tag{1.9}$$

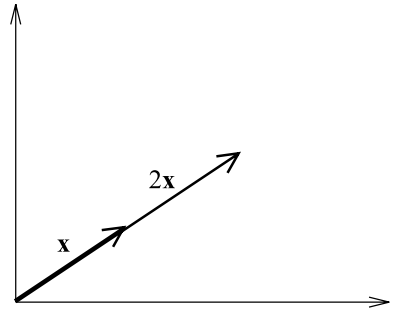
where T_c is the sampling period or interval,[1] and the *continuous-time signal space*, whose elements are the signals

$$x(t) \qquad t \in \Re \tag{1.10}$$

where $\Re$ denotes the set of real numbers.



(a) Sum of two vectors.



(b) Product of a vector by a scalar.

Figure 1.1. Geometrical interpretation in the two-dimensional space of the sum of two vectors and the multiplication of a vector by a scalar.

[1] Later a discrete-time signal will be indicated simply as $\{x(k)\}$, omitting the indication of the sampling period. In general, we will indicate by $\{x_k\}$ a sequence of real or complex numbers *not* necessarily generated at instants kT_c.

Inner product

In an I-dimensional Euclidean space,[2] given the two vectors $\mathbf{x} = [x_1, \ldots, x_I]^T$ and $\mathbf{y} = [y_1, \ldots, y_I]^T$, we indicate with $\langle \mathbf{x}, \mathbf{y} \rangle$ the inner product:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{I} x_i y_i^* \tag{1.11}$$

If $\langle \mathbf{x}, \mathbf{y} \rangle$ is real, there is an important geometrical interpretation of the inner product in the Euclidean space, represented in Figure 1.2, that is obtained from the relation:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \|\mathbf{x}\| \, \|\mathbf{y}\| \cos\theta \tag{1.12}$$

where $\|\mathbf{x}\|$ denotes the norm or length of the vector $\mathbf{x}$. Note that

$$\langle \mathbf{x}, \mathbf{x} \rangle = \sum_{i=1}^{I} |x_i|^2 = \|\mathbf{x}\|^2 \tag{1.13}$$

Observation 1.1
From (1.12),

$$\frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{y}\|} = \|\mathbf{x}\| \cos\theta \tag{1.14}$$

is the length of the projection of $\mathbf{x}$ onto $\mathbf{y}$.

Definition 1.2
Two vectors $\mathbf{x}$ and $\mathbf{y}$ are orthogonal ($\mathbf{x} \perp \mathbf{y}$) if $\langle \mathbf{x}, \mathbf{y} \rangle = 0$, that is if the angle they form is 90°.

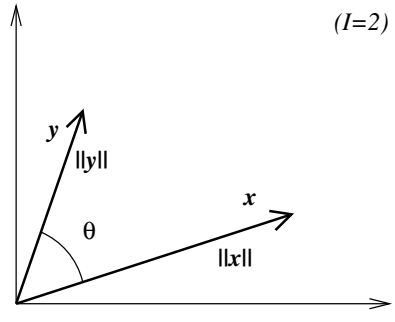


Figure 1.2. Geometrical representation of the inner product between two vectors. $\|\mathbf{x}\|$ is the norm of $\mathbf{x}$, that is the vector length.

[2] Henceforth: ^{T} stands for transpose, * for complex conjugate and ^{H} for transpose complex conjugate or Hermitian.

We can extend these concepts to a *signal space*, defining the inner product as

$$\sum_{k=-\infty}^{+\infty} x(k)\, y^*(k) \tag{1.15}$$

for discrete-time signals, and

$$\int_{-\infty}^{+\infty} x(t)\, y^*(t)\, dt \tag{1.16}$$

for continuous-time signals. In both cases it is assumed that the energy of signals is finite. Hence, for continuous-time signals it must be:

$$\int_{-\infty}^{+\infty} |x(t)|^2\, dt < \infty \quad \text{and} \quad \int_{-\infty}^{+\infty} |y(t)|^2\, dt < \infty \tag{1.17}$$

Recall that the inner product enjoys the following properties:

1. $\langle \mathbf{x} + \mathbf{y}, \mathbf{z} \rangle = \langle \mathbf{x}, \mathbf{z} \rangle + \langle \mathbf{y}, \mathbf{z} \rangle$.
2. $\langle \alpha \mathbf{x}, \mathbf{y} \rangle = \alpha \langle \mathbf{x}, \mathbf{y} \rangle$.
3. $\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{y}, \mathbf{x} \rangle^*$.
4. $\langle \mathbf{x}, \mathbf{x} \rangle > \mathbf{0} \qquad \forall \mathbf{x} \neq 0$.
5. (Schwarz inequality) $|\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\| \|\mathbf{y}\|$. Equality holds if and only if $\mathbf{x} = \mathtt{K}\mathbf{y}$, with $\mathtt{K}$ a complex scalar.

1.2 Discrete signal representation

Let us consider the problem of associating a sequence (possibly finite) of numbers with a continuous-time signal.[3] A basis of orthonormal signals (orthogonal signals with unit norm) $\{\phi_i(t)\}$, $t \in \Re$, $i \in \mathcal{I}$, where $\mathcal{I}$ is a finite or numerable set, is defined by

$$\langle \phi_i, \phi_j \rangle = \int_{-\infty}^{+\infty} \phi_i(t)\, \phi_j^*(t)\, dt = \delta_{i-j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{1.18}$$

In this text, δ_n is the Kronecker delta, whereas $\delta(t)$ denotes the Dirac delta.

Given a finite-energy signal $x(t)$, $t \in \Re$,

$$E_x = \langle x, x \rangle = \int_{-\infty}^{+\infty} |x(t)|^2\, dt < \infty \tag{1.19}$$

[3] This procedure can easily be extended to discrete-time signals.

we want to express $x(t)$, $t \in \Re$, as a linear combination of the functions $\{\phi_i(t)\}$, $i \in \mathcal{I}$. Consider the signal

$$\hat{x}(t) = \sum_{i \in \mathcal{I}} x_i \, \phi_i(t) \tag{1.20}$$

If we define the error as

$$e(t) = x(t) - \hat{x}(t) \tag{1.21}$$

the most common method to determine the coefficients $\{x_i\}$ in (1.20) is by minimizing the energy of e, defined as

$$E_e = \langle e, e \rangle = \int_{-\infty}^{+\infty} \left| x(t) - \sum_{i \in \mathcal{I}} x_i \, \phi_i(t) \right|^2 dt \tag{1.22}$$

Let

$$c_i = \langle x, \phi_i \rangle \qquad i \in \mathcal{I} \tag{1.23}$$

and express E_e as

$$\begin{aligned} E_e &= \langle x - \hat{x}, x - \hat{x} \rangle \\ &= \langle x, x \rangle - \langle \hat{x}, x \rangle - \langle x, \hat{x} \rangle + \langle \hat{x}, \hat{x} \rangle \\ &= E_x + \sum_{i \in \mathcal{I}} (x_i x_i^* - x_i c_i^* - x_i^* c_i) \end{aligned} \tag{1.24}$$

By adding and subtracting $|c_i|^2$, one finds

$$E_e = E_x + \sum_{i \in \mathcal{I}} |x_i - c_i|^2 - \sum_{i \in \mathcal{I}} |c_i|^2 \tag{1.25}$$

as $|x_i|^2 - x_i c_i^* - x_i^* c_i + |c_i|^2 = |x_i - c_i|^2 = (x_i - c_i)(x_i - c_i)^*$. The minimum of (1.25) is obtained if the second term is zero. Hence the coefficients to be determined are given by

$$x_i = c_i = \langle x, \phi_i \rangle = \int_{-\infty}^{+\infty} x(t) \phi_i^*(t) \, dt \qquad i \in \mathcal{I} \tag{1.26}$$

and

$$E_e = E_x - E_{\hat{x}} \tag{1.27}$$

where

$$E_{\hat{x}} = \sum_{i \in \mathcal{I}} |x_i|^2 \tag{1.28}$$

If $E_e = 0$, then $\{\phi_i(t)\}$, $i \in \mathcal{I}$, is a *complete basis* for $x(t)$, $t \in \Re$, and $x(t)$ can be expressed as

$$x(t) = \sum_{i \in \mathcal{I}} x_i \, \phi_i(t) \tag{1.29}$$

where equality must be intended in terms of quadratic norm. Moreover, from (1.29)

$$E_x = \sum_{i \in \mathcal{I}} |x_i|^2 \tag{1.30}$$

that is the energy of the signal coincides with the sum of the squares of the coefficient amplitudes.

The principle of orthogonality

The vector identified by the optimum coefficients (1.26) satisfies the following important property:

$$\langle e, \phi_i \rangle = \langle x - \hat{x}, \phi_i \rangle = x_i - x_i = 0 \qquad \forall i \in \mathcal{I} \tag{1.31}$$

that is $e \perp \phi_i$, $\forall i \in \mathcal{I}$.

As an example, for a generic signal x and for a basis formed by two signals ϕ_1 and ϕ_2, one gets the geometrical representation in Figure 1.3.

Signal representation

Definition 1.3
The signal $\hat{x}(t)$, $t \in \Re$, given by (1.20) with $\{x_i = c_i\}$, $i \in \mathcal{I}$, is called the *projection* of $x(t)$, $t \in \Re$, onto the space spanned by the signals $\{\phi_i(t)\}$, $i \in \mathcal{I}$, that is, the space whose signals are expressed as linear combinations of $\{\phi_i(t)\}$, $i \in \mathcal{I}$.

If $E_e = 0$, then $x(t)$, $t \in \Re$, belongs to the space spanned by $\{\phi_i(t)\}$, $i \in \mathcal{I}$. Therefore, given a sequence of orthonormal signals, which form a complete basis for $x(t)$, the signal $x(t)$ can be represented by a sequence of numbers $\{x_i\}$, $i \in \mathcal{I}$, given by

$$x_i = \langle x, \phi_i \rangle \tag{1.32}$$

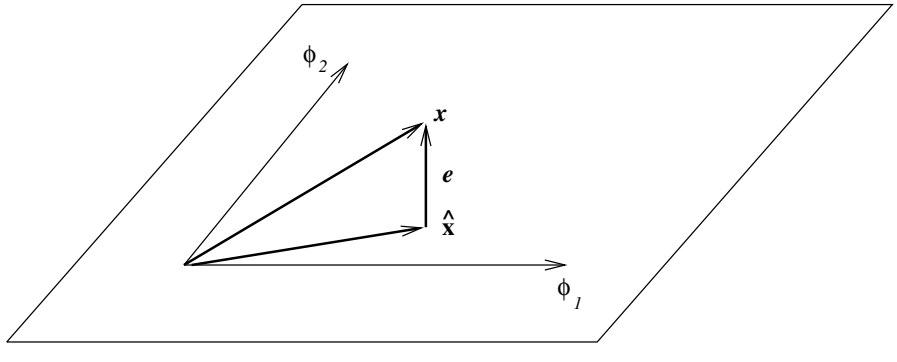


Figure 1.3. Geometrical representation of the projection of **x** onto the space spanned by ϕ_1 and ϕ_2.

In short, we have the following correspondence between a signal and its vector representation:

$$x(t) \qquad t \in \Re \longleftrightarrow \mathbf{x} = [\ldots, x_{-1}, x_0, x_1, \ldots]^T \tag{1.33}$$

It is useful to analyze the inner product between signals in terms of the corresponding vector representations. Let

$$x(t) = \sum_{i \in \mathcal{I}} x_i \phi_i(t) \quad \text{and} \quad y(t) = \sum_{i \in \mathcal{I}} y_i \phi_i(t) \tag{1.34}$$

then

$$\langle x, y \rangle = \int_{-\infty}^{+\infty} x(t)\, y^*(t)\, dt = \langle \mathbf{x}, \mathbf{y} \rangle \tag{1.35}$$

In particular,

$$E_x = \int_{-\infty}^{+\infty} |x(t)|^2\, dt = \sum_{i \in \mathcal{I}} |x_i|^2 = \langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2 = E_{\mathbf{x}} \tag{1.36}$$

We introduce now the Euclidean distance between two signals:

$$d(x, y) = \sqrt{E_{x-y}} = \left(\int_{-\infty}^{+\infty} |x(t) - y(t)|^2\, dt \right)^{\frac{1}{2}} \tag{1.37}$$

$$= \left(\sum_{i \in \mathcal{I}} |x_i - y_i|^2 \right)^{\frac{1}{2}} = d(\mathbf{x}, \mathbf{y}) \tag{1.38}$$

In other words, the Euclidean distance between two signals coincides with the Euclidean distance between the corresponding vectors.

Moreover, the following relation holds:[4]

$$d^2(x, y) = E_x + E_y - 2Re\left[\langle x, y \rangle\right] \tag{1.39}$$

In particular we have $E_e = d^2(x, \hat{x})$. Let

$$\rho = \frac{\langle x, y \rangle}{\|x\|\, \|y\|} \tag{1.40}$$

be the *correlation coefficient* between x and y. If ρ is real, observing (1.12) $\rho = \cos\theta$, with θ the angle formed by $\mathbf{x}$ and $\mathbf{y}$. Then (1.39) becomes:

$$d^2(x, y) = E_x + E_y - 2\rho\sqrt{E_x E_y} \tag{1.41}$$

We note that if ρ is real, the signals are not necessarily real, it is sufficient that their inner product is real.

[4] The symbols $Re\,[c]$ and $Im\,[c]$ denote, respectively, the real and the imaginary part of c.

Example 1.2.1
Resorting to the *sampling theorem*, it can be shown that for a real valued signal $x(t)$, $t \in \Re$, with finite bandwidth B (see (1.140)), the sequence of functions

$$\phi_i(t) = \frac{\sin\left(\pi \frac{1}{T_c}(t - iT_c)\right)}{\pi \frac{1}{T_c}(t - iT_c)} \qquad T_c = \frac{1}{2B} \tag{1.42}$$

forms a complete orthogonal basis for x. The coefficients are given by the samples of $x(t)$ at the time instants $t = iT_c$,

$$x_i = x(iT_c) \tag{1.43}$$

Gram–Schmidt orthonormalization procedure

Given a set of M signals

$$\{s_m(t)\} \qquad m = 1, 2, \ldots, M \tag{1.44}$$

a procedure to derive an orthonormal basis for this set is now outlined. We indicate by $\{\phi'_i(t)\}$ a set of orthogonal functions and by

$$\left\{ \phi_i(t) = \frac{\phi'_i(t)}{\sqrt{E_{\phi'_i}}} \right\} \tag{1.45}$$

the orthonormal functions obtained from $\{\phi'_i(t)\}$.

1. Let

$$\phi'_1(t) = s_1(t) \tag{1.46}$$

Then it follows

$$\phi_1(t) = \frac{\phi'_1(t)}{\sqrt{E_{\phi'_1}}} \tag{1.47}$$

2. Let

$$\phi'_2(t) = s_2(t) - \langle s_2, \phi_1 \rangle \phi_1(t) \tag{1.48}$$

it is easy to see that $\phi'_2 \perp \phi_1$; in fact,

$$\langle \phi'_2, \phi_1 \rangle = \langle s_2 - \langle s_2, \phi_1 \rangle \phi_1, \phi_1 \rangle = \langle s_2, \phi_1 \rangle - \langle s_2, \phi_1 \rangle = 0 \tag{1.49}$$

As illustrated in Figure 1.4, in (1.48) $\langle s_2, \phi_1 \rangle \phi_1(t)$ is the projection of s_2 upon ϕ_1. Then, from (1.48)

$$\phi_2(t) = \frac{\phi'_2(t)}{\sqrt{E_{\phi'_2}}} \tag{1.50}$$

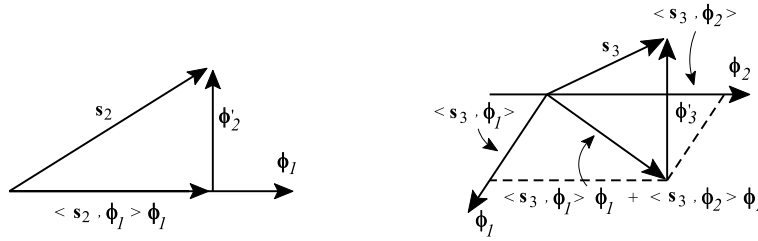


Figure 1.4. Geometrical representation of the Gram--Schmidt orthonormalization procedure.

3. Let

$$\phi_3'(t) = s_3(t) - \langle s_3, \phi_1 \rangle \phi_1(t) - \langle s_3, \phi_2 \rangle \phi_2(t) \tag{1.51}$$

then one gets $\phi_3' \perp \phi_1$ and $\phi_3' \perp \phi_2$.

In general

$$\phi_i'(t) = s_i(t) - \sum_{j=1}^{i-1} \langle s_i, \phi_j \rangle \, \phi_j(t) \tag{1.52}$$

and if $\phi_i'(t)$ is not identically zero, we choose

$$\phi_i(t) = \frac{\phi_i'(t)}{\sqrt{E_{\phi_i'}}} \tag{1.53}$$

It follows that $\phi_i \perp \phi_j$ for $j = 1, 2, \ldots, i-1$. The procedure is represented geometrically in Figure 1.4, limited to $\boldsymbol{\phi}_2'$ and $\boldsymbol{\phi}_3'$.

Observation 1.2
The set of $\{\phi_i(t)\}$ is not unique, in any case the reciprocal distances between signals remain unchanged.

Observation 1.3
The number of dimensions I of $\{\phi_i(t)\}$ can be lower than M if the signals $\{s_m(t)\}$, $m = 1, \ldots, M$, are linearly dependent, that is if there exists a set of coefficients, not all equal to zero, such that

$$\sum_{m=1}^{M} c_m s_m(t) = 0 \qquad \forall t \tag{1.54}$$

In such a case, it happens that in (1.52) for some i the signal $\{\phi_i'(t)\}$ is identically zero. Obviously, a null signal cannot be an element of the basis.

Let us look at a few examples of discrete representation of a set of signals.

Example 1.2.2
For the three signals

$$s_1(t) = \begin{cases} A \sin 2\pi \dfrac{t}{T} & 0 < t < \dfrac{T}{2} \\ 0 & \text{elsewhere} \end{cases} \tag{1.55}$$

$$s_2(t) = \begin{cases} A \sin 2\pi \dfrac{t}{T} & 0 < t < T \\ 0 & \text{elsewhere} \end{cases} \tag{1.56}$$

$$s_3(t) = \begin{cases} A \sin 2\pi \dfrac{t}{T} & \dfrac{T}{2} < t < T \\ 0 & \text{elsewhere} \end{cases} \tag{1.57}$$

which are depicted in Figure 1.5, an orthonormal basis, represented in Figure 1.6, is the following:

$$\phi_1(t) = \begin{cases} \dfrac{2}{\sqrt{T}} \sin 2\pi \dfrac{t}{T} & 0 < t < \dfrac{T}{2} \\ 0 & \text{elsewhere} \end{cases} \tag{1.58}$$

$$\phi_2(t) = \begin{cases} \dfrac{2}{\sqrt{T}} \sin 2\pi \dfrac{t}{T} & \dfrac{T}{2} < t < T \\ 0 & \text{elsewhere} \end{cases} \tag{1.59}$$

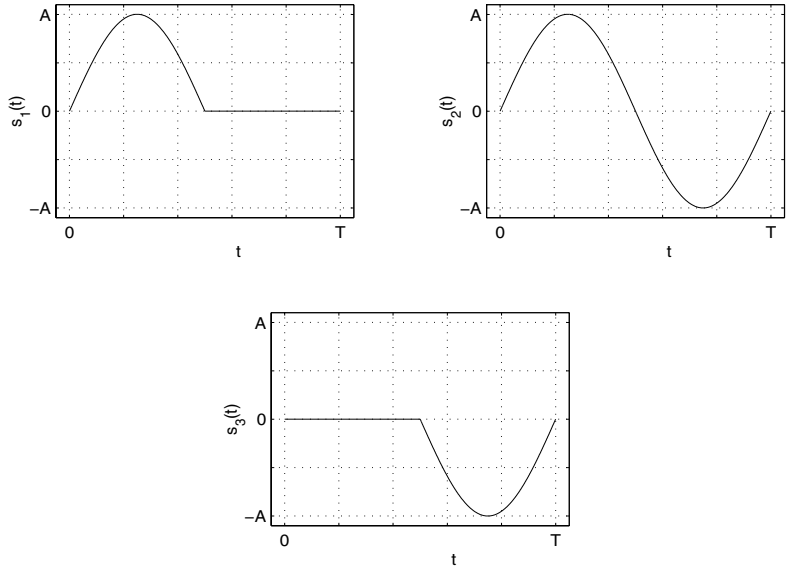


Figure 1.5. The three signals.

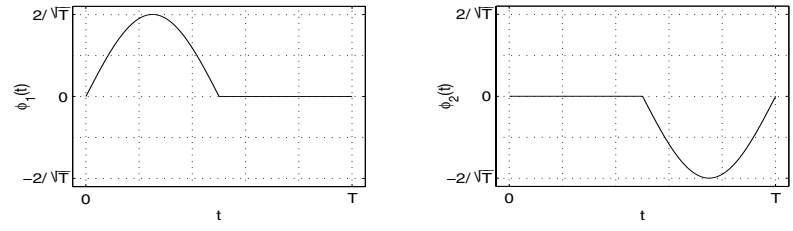


Figure 1.6. Orthonormal basis for signals of Figure 1.5.

Moreover,

$$s_1(t) = \frac{A}{2}\sqrt{T}\phi_1(t) \tag{1.60}$$

$$s_2(t) = \frac{A}{2}\sqrt{T}\phi_1(t) + \frac{A}{2}\sqrt{T}\phi_2(t) \tag{1.61}$$

$$s_3(t) = \frac{A}{2}\sqrt{T}\phi_2(t) \tag{1.62}$$

from which the correspondence between signals and their vector representation is (see Figure 1.7):

$$s_1(t) \longleftrightarrow \mathbf{s}_1 = \left[\frac{A}{2}\sqrt{T}, 0\right]^T \tag{1.63}$$

$$s_2(t) \longleftrightarrow \mathbf{s}_2 = \left[\frac{A}{2}\sqrt{T}, \frac{A}{2}\sqrt{T}\right]^T \tag{1.64}$$

$$s_3(t) \longleftrightarrow \mathbf{s}_3 = \left[0, \frac{A}{2}\sqrt{T}\right]^T \tag{1.65}$$

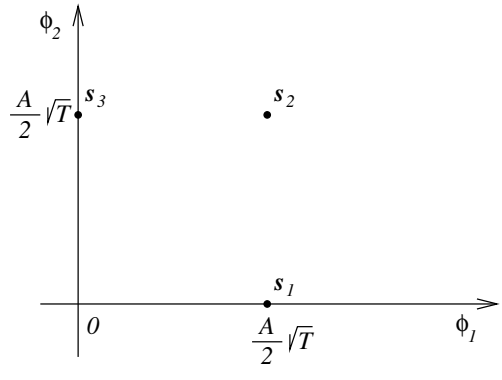


Figure 1.7. Vector representation or constellation of signals of Figure 1.5.

We note that the three signals are represented as a linear combination of only two functions ($I = 2$).

Definition 1.4
The vector representation of a set of M signals is often called a *signal constellation*.

Example 1.2.3 (4-PSK)
Given the set of four signals

$$s_m(t) = \begin{cases} A\cos\left(2\pi f_0 t + \left(m - \frac{1}{2}\right)\frac{\pi}{2}\right) & 0 < t < T \qquad m = 1, 2, 3, 4 \\ 0 & \text{elsewhere} \end{cases} \tag{1.66}$$

depicted in Figure 1.8, to determine the basis functions we write $s_m(t)$ as

$$s_m(t) = A\cos\left[\left(m - \frac{1}{2}\right)\frac{\pi}{2}\right]\cos(2\pi f_0 t) - A\sin\left[\left(m - \frac{1}{2}\right)\frac{\pi}{2}\right]\sin(2\pi f_0 t) \tag{1.67}$$

We choose the following basis:

$$\phi_1(t) = \begin{cases} +\sqrt{\frac{2}{T}}\cos(2\pi f_0 t) & 0 < t < T \\ 0 & \text{elsewhere} \end{cases}$$

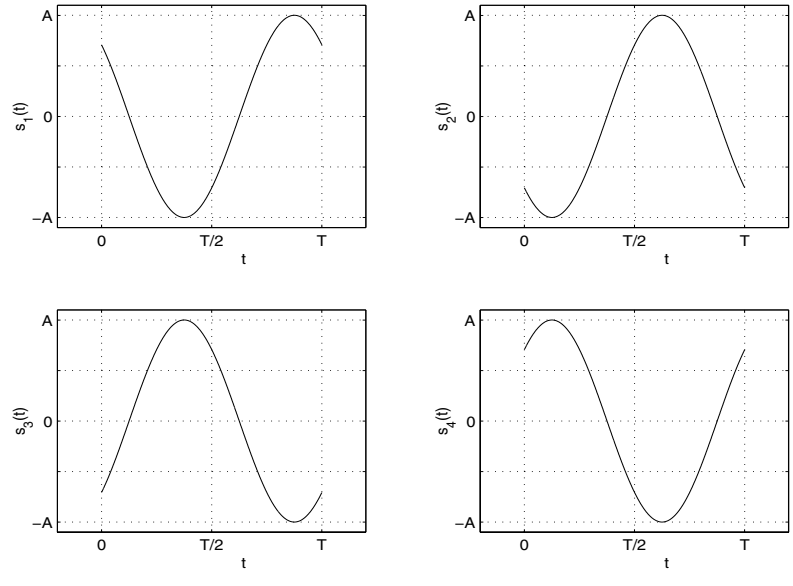


Figure 1.8. Modulated 4-PSK signals for $f_0 = 1/T$.

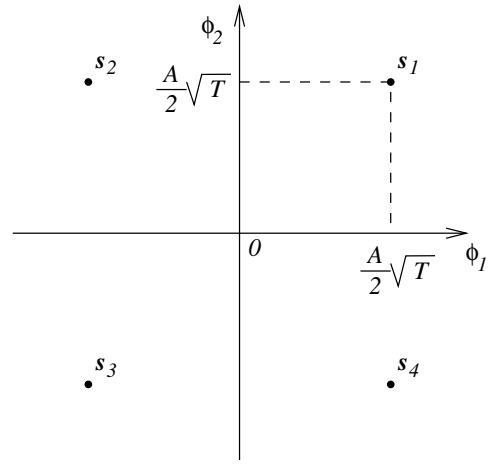


Figure 1.9. 4-PSK constellation.

and

$$\phi_2(t) = \begin{cases} -\sqrt{\frac{2}{T}}\sin(2\pi f_0 t) & 0 < t < T \\ 0 & \text{elsewhere} \end{cases} \tag{1.68}$$

One finds that

$$E_{\phi_1} = 1 + \frac{\sin \pi 4 f_0 T}{\pi 4 f_0 T} \qquad E_{\phi_2} = 1 - \frac{\sin \pi 4 f_0 T}{\pi 4 f_0 T} \tag{1.69}$$

and

$$\langle \phi_1, \phi_2 \rangle = \frac{\sin^2 2\pi f_0 T}{2\pi f_0 T} \tag{1.70}$$

Hence if

$$f_0 = \frac{k}{2T} \ (k \text{ integer}) \quad \text{or} \quad f_0 \gg \frac{1}{T} \tag{1.71}$$

then it results $\langle \phi_1, \phi_2 \rangle \simeq 0$, and $E_{\phi_i} \simeq 1$ for $i = 1, 2$. Under these conditions for f_0, $\phi_1(t)$ and $\phi_2(t)$ form an othonormal basis for the four signals in Figure 1.8, whose constellation is given in Figure 1.9.

1.3 Continuous-time linear systems

A time-invariant continuous-time continuous-amplitude linear system, also called analog filter, is represented in Figure 1.10, where x and y are the input and output signals, respectively, and h denotes the filter impulse response.

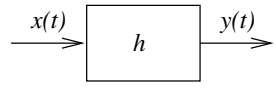


Figure 1.10. Analog filter as a time-invariant linear system with continuous domain.

The output at a certain instant $t \in \Re$ is given by the convolution integral

$$y(t) = \int_{-\infty}^{\infty} h(t-\tau)\, x(\tau)\, d\tau = \int_{-\infty}^{\infty} h(\tau)\, x(t-\tau)\, d\tau \tag{1.72}$$

In short we will use the notation

$$y(t) = x * h(t) \tag{1.73}$$

where the symbol '$*$' denotes the convolution operation (1.72).

We also introduce the Fourier transform of the signal $x(t)$, $t \in \Re$,

$$\mathcal{X}(f) = \mathcal{F}[x(t)] = \int_{-\infty}^{+\infty} x(t)\, e^{-j2\pi ft}\, dt \qquad f \in \Re \tag{1.74}$$

The inverse Fourier transform is given by

$$x(t) = \int_{-\infty}^{\infty} \mathcal{X}(f)\, e^{j2\pi ft}\, df \tag{1.75}$$

In the frequency domain, (1.73) becomes

$$\mathcal{Y}(f) = \mathcal{X}(f)\, \mathcal{H}(f) \qquad f \in \Re \tag{1.76}$$

where $\mathcal{H}$ is the filter frequency response. The magnitude of the frequency response, $|\mathcal{H}(f)|$, is usually called the *magnitude response* or *amplitude response*.

General properties of the Fourier transform are given in Table 1.1.[5]

Definition 1.5

We introduce two functions that will be extensively used:

$$\text{rect}\left(\frac{f}{F}\right) = \begin{cases} 1 & |f| < \dfrac{F}{2} \\ 0 & \text{elsewhere} \end{cases} \tag{1.77}$$

[5] Two important functions that will be used very often are:

$$\text{step function:} \quad 1(t) = \begin{cases} 1 & t > 0 \\ 0 & t < 0 \end{cases}$$

$$\text{sign function:} \quad \text{sgn}(t) = \begin{cases} 1 & t > 0 \\ -1 & t < 0 \end{cases}$$

Table 1.1 Some general properties of the Fourier transform.

Property	Signal	Fourier transform
	$x(t)$	$\mathcal{X}(f)$
linearity	$a\,x(t) + b\,y(t)$	$a\,\mathcal{X}(f) + b\,\mathcal{Y}(f)$
duality	$\mathcal{X}(t)$	$x(-f)$
time inverse	$x(-t)$	$\mathcal{X}(-f)$
complex conjugate	$x^*(t)$	$\mathcal{X}^*(-f)$
real part	$Re[x(t)] = \dfrac{x(t) + x^*(t)}{2}$	$\dfrac{1}{2}\,[\mathcal{X}(f) + \mathcal{X}^*(-f)]$
imaginary part	$Im[x(t)] = \dfrac{x(t) - x^*(t)}{2j}$	$\dfrac{1}{2j}\,[\mathcal{X}(f) - \mathcal{X}^*(-f)]$
scaling	$x(at),\ a \neq 0$	$\dfrac{1}{\lvert a\rvert}\,\mathcal{X}\left(\dfrac{f}{a}\right)$
time shift	$x(t - t_0)$	$e^{-j2\pi f t_0}\,\mathcal{X}(f)$
frequency shift	$x(t)\,e^{j2\pi f_0 t}$	$\mathcal{X}(f - f_0)$
modulation	$x(t)\cos(2\pi f_0 t + \varphi)$	$\dfrac{1}{2}\,[e^{j\varphi}\mathcal{X}(f - f_0) + e^{-j\varphi}\mathcal{X}(f + f_0)]$
	$x(t)\sin(2\pi f_0 t + \varphi)$	$\dfrac{1}{2j}\,[e^{j\varphi}\mathcal{X}(f - f_0) - e^{-j\varphi}\mathcal{X}(f + f_0)]$
	$Re[x(t)\,e^{j(2\pi f_0 t + \varphi)}]$	$\dfrac{1}{2}[e^{j\varphi}\mathcal{X}(f - f_0) + e^{-j\varphi}\mathcal{X}^*(-f - f_0)]$
differentiation	$\dfrac{d}{dt}\,x(t)$	$j2\pi f\,\mathcal{X}(f)$
integration	$\displaystyle\int_{-\infty}^{t} x(\tau)\,d\tau = 1 * x(t)$	$\dfrac{1}{j2\pi f}\,\mathcal{X}(f) + \dfrac{\mathcal{X}(0)}{2}\,\delta(f)$
convolution	$[x(\tau) * y(\tau)](t)$	$\mathcal{X}(f)\,\mathcal{Y}(f)$
correlation	$[x(\tau) * y^*(-\tau)](t)$	$\mathcal{X}(f)\,\mathcal{Y}^*(f)$
product	$x(t)\,y(t)$	$[\mathcal{X}(\eta) * \mathcal{Y}(\eta)](f)$
real signal	$x(t) = x^*(t)$	$\mathcal{X}(f) = \mathcal{X}^*(-f)$, $\mathcal{X}$ Hermitian, $Re[\mathcal{X}(f)]$ even, $Im[\mathcal{X}(f)]$ odd, $\lvert\mathcal{X}(f)\rvert^2$ even
imaginary signal	$x(t) = -x^*(t)$	$\mathcal{X}(f) = -\mathcal{X}^*(-f)$
real and even signal	$x(t) = x^*(t) = x(-t)$	$\mathcal{X}(f) = \mathcal{X}^*(f) = \mathcal{X}(-f)$, $\mathcal{X}$ real and even
real and odd signal	$x(t) = x^*(t) = -x(-t)$	$\mathcal{X}(f) = -\mathcal{X}^*(f) = -\mathcal{X}(-f)$, $\mathcal{X}$ imaginary and odd
Parseval's theorem	$E_x = \displaystyle\int_{-\infty}^{+\infty} \lvert x(t)\rvert^2\,dt = \int_{-\infty}^{+\infty} \lvert\mathcal{X}(f)\rvert^2\,df = E_{\mathcal{X}}$	
Poisson sum formula	$\displaystyle\sum_{k=-\infty}^{+\infty} x(kT_c) = \frac{1}{T_c}\sum_{\ell=-\infty}^{+\infty} \mathcal{X}\left(\frac{\ell}{T_c}\right)$	

$$\text{sinc}(t) = \frac{\sin(\pi t)}{\pi t} \tag{1.78}$$

The following relation holds:

$$\mathcal{F}[\text{sinc}(Ft)] = \frac{1}{F}\,\text{rect}\left(\frac{f}{F}\right) \tag{1.79}$$

as illustrated in Figure 1.11.

Further examples of signals and relative Fourier transforms are given in Table 1.2. We reserve the notation $H(s)$ to indicate the Laplace transform of $h(t)$, $t \in \Re$:

$$H(s) = \int_{-\infty}^{+\infty} h(t)e^{-st}\,dt \tag{1.80}$$

with s complex variable; $H(s)$ is also called the *transfer function* of the filter. A class of functions $H(s)$ often used in practice is characterized by the ratio of two polynomials in s, each with a finite number of coefficients.

It is easy to observe that if the curve $s = j2\pi f$ in the s-plane belongs to the convergence region of the integral in (1.80), then $\mathcal{H}(f)$ is related to $H(s)$ by

$$\mathcal{H}(f) = H(s)|_{s=j2\pi f} \tag{1.81}$$

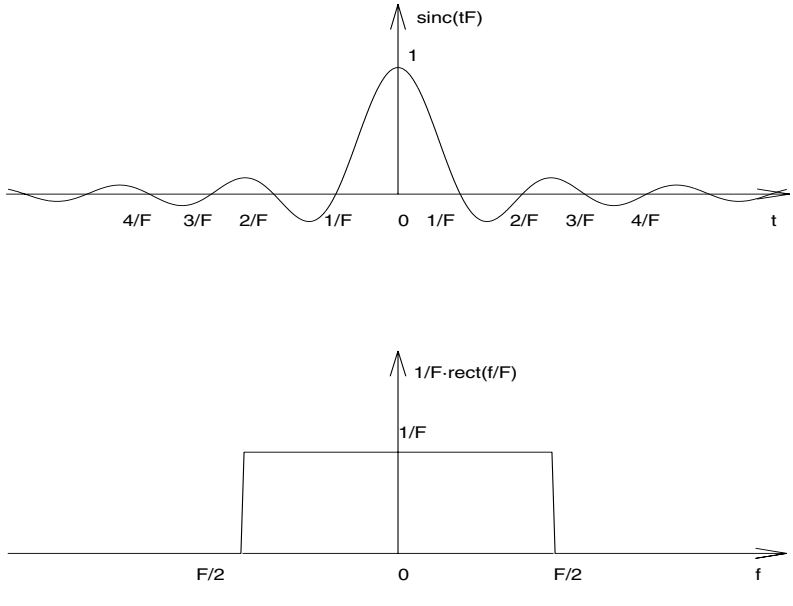


Figure 1.11. Example of signal and Fourier transform pair.

Table 1.2 Examples of Fourier transform signal pairs.

Signal	Fourier transform
$x(t)$	$\mathcal{X}(f)$
$\delta(t)$	1
1	$\delta(f)$
$e^{j2\pi f_0 t}$	$\delta(f - f_0)$
$\cos(2\pi f_0 t)$	$\frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)]$
$\sin(2\pi f_0 t)$	$\frac{1}{2j}[\delta(f - f_0) - \delta(f + f_0)]$
$1(t)$	$\frac{1}{2}\delta(f) + \frac{1}{j2\pi f}$
$\mathrm{sgn}(t)$	$\frac{1}{j\pi f}$
$\mathrm{rect}\left(\frac{t}{T}\right)$	$T\,\mathrm{sinc}(fT)$
$\mathrm{sinc}\left(\frac{t}{T}\right)$	$T\,\mathrm{rect}(fT)$
$\left(1 - \frac{\lvert t\rvert}{T}\right)\mathrm{rect}\left(\frac{t}{2T}\right)$	$T\,\mathrm{sinc}^2(fT)$
$e^{-at}\,1(t),\ a > 0$	$\frac{1}{a + j2\pi f}$
$t\,e^{-at}\,1(t),\ a > 0$	$\frac{1}{(a + j2\pi f)^2}$
$e^{-a\lvert t\rvert},\ a > 0$	$\frac{2a}{a^2 + (2\pi f)^2}$
$e^{-at^2},\ a > 0$	$\sqrt{\frac{\pi}{a}}\exp\left(-\pi\,\frac{\pi}{a}f^2\right)$

1.4 Discrete-time linear systems

A discrete–time time-invariant linear system, with sampling period T_c, is shown in Figure 1.12, where $x(k)$ and $y(k)$ are respectively the input and output signals at the time instants kT_c, $k \in \mathcal{Z}$, where $\mathcal{Z}$ denotes the set of integers. The impulse response of the system is denoted by $\{h(k)\}$, $k \in \mathcal{Z}$, or more simply by h.

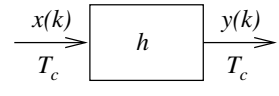


Figure 1.12. Discrete-time linear system (filter).

The relation between the input sequence $\{x(k)\}$ and the output sequence $\{y(k)\}$ is given by the convolution operation:

$$y(k) = [x(m) * h(m)](k) = \sum_{n=-\infty}^{+\infty} h(k-n)x(n) \tag{1.82}$$

In short, we will use the notation $y(k) = x * h(k)$.

We list some definitions that are valid for time-invariant linear systems. We say the system is *causal* (*anticausal*) if $h(k) = 0$, $k < 0$ (if $h(k) = 0$, $k > 0$).

We define as *transfer function* of the filter the z-transform[6] of the impulse response h, given by

$$H(z) = \sum_{k=-\infty}^{+\infty} h(k)z^{-k} \tag{1.83}$$

We indicate with $\mathcal{H}(f)$ the *frequency response* of the filter, defined as

$$\mathcal{H}(f) = \mathcal{F}[h(k)] = \sum_{k=-\infty}^{+\infty} h(k)e^{-j2\pi fkT_c} = H(z)_{z=e^{j2\pi fT_c}} \tag{1.84}$$

The inverse Fourier transform of the frequency response yields

$$h(k) = T_c \int_{-\frac{1}{2T_c}}^{+\frac{1}{2T_c}} \mathcal{H}(f)e^{j2\pi fkT_c}\, df \tag{1.85}$$

We note the property that, for $x(k) = b^k$, where b is a complex constant, the output is given by $y(k) = H(b)\ b^k$. In Table 1.3 some further properties of the z-transform are summarized. For discrete-time linear systems, in the frequency domain (1.82) becomes

$$\mathcal{Y}(f) = \mathcal{X}(f)\mathcal{H}(f) \tag{1.86}$$

where all functions are periodic of period $1/T_c$.

Example 1.4.1

A fundamental example of z–transform is that of the sequence:

$$h(k) = \begin{cases} a^k & k \geq 0 \\ 0 & k < 0 \end{cases} \qquad |a| < 1 \tag{1.87}$$

Applying the transform formula (1.83) we find

$$H(z) = \frac{1}{1 - az^{-1}} = \frac{z}{z - a} \tag{1.88}$$

under the condition $|a/z| < 1$.

---

[6] Sometimes the D transform is used instead of the z-transform, where $D = z^{-1}$, and $H(z)$ is replaced by $h(D) = \sum_{k=-\infty}^{+\infty} h(k)D^k$.

Table 1.3 Properties of the z-transform.

Property	Sequence	z transform
	$x(k),\ y(k)$	$X(z),\ Y(z)$
linearity	$ax(k) + by(k)$	$aX(z) + bY(z)$
delay	$x(k-m)$	$z^{-m}X(z)$
complex conjugate	$x^*(k)$	$X^*(z^*)$
inverse time	$x(-k)$	$X\left(\frac{1}{z}\right)$
	$x^*(-k)$	$X^*\left(\frac{1}{z^*}\right)$
scaling	$a^{-k}x(k)$	$X(az)$
convolution	$[x(m) * y(m)](k)$	$X(z)Y(z)$
correlation	$[x(m) * y^*(-m)](k)$	$X(z)Y^*\left(\frac{1}{z^*}\right)$
real sequence	$x(k) = x^*(k)$	$X(z) = X^*(z^*)$

Example 1.4.2
Let $q(t)$, $t \in \Re$, be a continuous-time signal with Fourier transform $\mathcal{Q}(f)$, $f \in \Re$. We now consider the sequence obtained by sampling $q(t)$, that is

$$h_k = q(kT_c) \qquad k \in \mathcal{Z} \tag{1.89}$$

Using the Poisson formula of Table 1.1, one demonstrates that the Fourier transform of the sequence $\{h_k\}$ is related to $\mathcal{Q}(f)$ by

$$\mathcal{H}(f) = \mathcal{F}[h_k] = H\left(e^{j2\pi f T_c}\right) = \frac{1}{T_c}\sum_{\ell=-\infty}^{\infty}\mathcal{Q}\left(f - l\frac{1}{T_c}\right) \tag{1.90}$$

Discrete Fourier transform (DFT)

For a sequence with a finite number of samples, $\{g_k\}$, $k = 0, 1, \ldots, N-1$, the expression of the Fourier transform becomes

$$\mathcal{G}(f) = \sum_{k=0}^{N-1} g_k e^{-j2\pi f k T_c} \tag{1.91}$$

Evaluating $\mathcal{G}(f)$ at the points $f = m/(NT_c)$, $m = 0, 1, \ldots, N-1$, and setting $\mathcal{G}_m = \mathcal{G}(m/(NT_c))$, we obtain:

$$\mathcal{G}_m = \sum_{k=0}^{N-1} g_k W_N^{km} \qquad W_N = e^{-j\frac{2\pi}{N}} \tag{1.92}$$

The sequence $\{\mathcal{G}_m\}$, $m = 0, 1, \ldots, N-1$, is called the DFT of $\{g_k\}$, $k = 0, 1, \ldots, N-1$. The inverse of (1.92) is given by

$$g_k = \frac{1}{N} \sum_{m=0}^{N-1} \mathcal{G}_m W_N^{-km} \qquad k = 0, 1, \ldots, N-1 \tag{1.93}$$

We note that, besides the factor $1/N$, the expression of the inverse DFT (IDFT) coincides with that of the DFT, provided W_N^{-1} is substituted with W_N.

We also observe that direct computation of (1.92) requires $N(N-1)$ complex additions and N^2 complex multiplications; however, the algorithm known as *fast Fourier transform* (FFT) allows computation of the DFT by $N \log_2 N$ complex additions and $\left(\frac{N}{2} \log_2 N - N\right)$ complex multiplications.[7]

The DFT operator

The DFT operator can be expressed in matrix form as

$$\mathbf{F} = \begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ 1 & W_N & W_N^2 & \ldots & W_N^{(N-1)} \\ 1 & W_N^2 & W_N^4 & \ldots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{(N-1)} & W_N^{(N-1)2} & \ldots & W_N^{(N-1)(N-1)} \end{bmatrix} \tag{1.94}$$

with elements $[\mathbf{F}]_{i,n} = W_N^{in}$, $i, n = 0, 1, \ldots, N-1$. The inverse operator (IDFT) is given by

$$\mathbf{F}^{-1} = \frac{1}{N} \mathbf{F}^* \tag{1.95}$$

We note that $\mathbf{F} = \mathbf{F}^T$, and $(1/\sqrt{N})\mathbf{F}$ is a unitary matrix.[8]

The following property holds: if $\mathbf{C}$ is a right circulant square matrix whose rows are obtained by successive shift to the right of the first row, then $\mathbf{FCF}^{-1}$ is a diagonal matrix whose elements are given by the DFT of the first row of $\mathbf{C}$.

Introducing the vector formed by the samples of the sequence $\{g_k\}$, $k = 0, 1, \ldots, N-1$,

$$\mathbf{g}^T = [g_0, g_1, \ldots, g_{N-1}] \tag{1.96}$$

and the vector of transform coefficients

$$\boldsymbol{\mathcal{G}}^T = [\mathcal{G}_0, \mathcal{G}_1, \ldots, \mathcal{G}_{N-1}] = \mathrm{DFT}[\mathbf{g}] \tag{1.97}$$

observing (1.92) it is immediate to verify the following relation:

$$\boldsymbol{\mathcal{G}} = \mathbf{Fg} \tag{1.98}$$

---

[7] The computational complexity of the FFT is often expressed as $N \log_2 N$.

[8] A square matrix $\mathbf{A}$ is unitary if $\mathbf{A}^H \mathbf{A} = \mathbf{I}$ where $\mathbf{I}$ is the identity matrix, i.e. a matrix for which all elements are zero except the elements on the main diagonal that are all equal to one.

Moreover, based on (1.95), we obtain

$$\mathbf{g} = \frac{1}{N}\mathbf{F}^{*}\boldsymbol{\mathcal{G}} \tag{1.99}$$

Circular and linear convolution via DFT

Let the two sequences x and h have a finite support of L_x and N samples respectively (see Figure 1.13) with $L_x > N$:

$$x(k) = 0 \qquad k < 0 \qquad k > L_x - 1 \tag{1.100}$$

and

$$h(k) = 0 \qquad k < 0 \qquad k > N - 1 \tag{1.101}$$

We define the periodic signals of period L,

$$x_{\mathrm{rep}_L}(k) = \sum_{\ell=-\infty}^{+\infty} x(k - \ell L) \tag{1.102}$$

and

$$h_{\mathrm{rep}_L}(k) = \sum_{\ell=-\infty}^{+\infty} h(k - \ell L) \tag{1.103}$$

where in order to avoid *time aliasing* it must be

$$L \geq L_x \quad \text{ed} \quad L \geq N \tag{1.104}$$

Definition 1.6
The *circular convolution* between x and h is a periodic sequence of period L defined as

$$y^{(\mathrm{circ})}(k) = h \overset{L}{\otimes} x(k) = \sum_{i=0}^{L-1} h_{\mathrm{rep}_L}(i)\, x_{\mathrm{rep}_L}(k - i) \tag{1.105}$$

with *main period* corresponding to $k = 0, 1, \ldots, L - 1$.

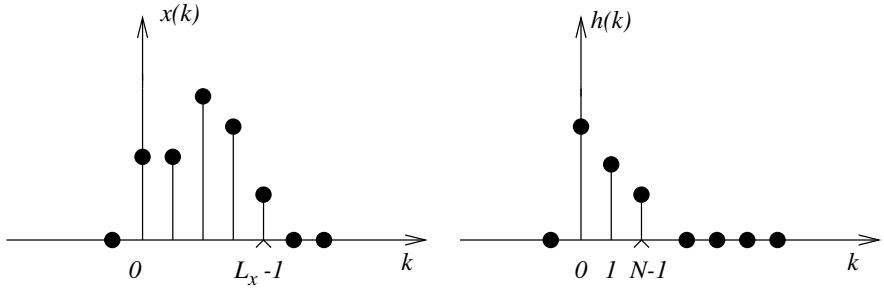


Figure 1.13. Time-limited signals: $\{x(k)\}$, $k = 0, 1, \ldots, L_x - 1$, and $\{h(k)\}$, $k = 0, 1, \ldots, N - 1$.

Then, if we indicate with $\{\mathcal{X}_m\}$, $\{\mathcal{H}_m\}$, and $\{\mathcal{Y}_m^{(\text{circ})}\}$, $m = 0, 1, \ldots, L-1$, the L-point DFT of sequences x, h, and $y^{(\text{circ})}$, respectively, we obtain

$$\mathcal{Y}_m^{(\text{circ})} = \mathcal{X}_m \mathcal{H}_m \qquad m = 0, 1, \ldots, L-1 \tag{1.106}$$

In vector notation (1.97), (1.106) becomes[9]

$$\boldsymbol{\mathcal{Y}}^{(\text{circ})} = \left[\mathcal{Y}_0^{(\text{circ})}, \mathcal{Y}_1^{(\text{circ})}, \ldots, \mathcal{Y}_{L-1}^{(\text{circ})}\right]^T = \text{diag}\{\text{DFT}[\boldsymbol{x}]\}\boldsymbol{\mathcal{H}} \tag{1.107}$$

where $\boldsymbol{\mathcal{H}}$ is the column vector given by the L-point DFT of the sequence h.

We are often interested in the *linear convolution* between x and h given by (1.82):

$$y(k) = x * h(k) = \sum_{i=0}^{N-1} h(i)x(k-i) \tag{1.108}$$

whose support is $k = 0, 1, \ldots, L_x + N - 2$. By comparing (1.108) with (1.105), it is easy to see that only if

$$L \geq L_x + N - 1 \tag{1.109}$$

then

$$y(k) = y^{(\text{circ})}(k) \qquad k = 0, 1, \ldots, L-1 \tag{1.110}$$

To compute the convolution between the two finite-length sequences x and h, (1.109) and (1.110) require that both sequences be completed with zeros (zero padding) to get a length of $L = L_x + N - 1$ samples. Then, taking the L-point DFT of the two sequences, performing the product (1.106), and taking the inverse transform of the result, one obtains the desired linear convolution.

We give below two relations between the circular convolution $y^{(\text{circ})}$ and the linear convolution y; in both cases we use

$$L = L_x \tag{1.111}$$

with $L > N$.

Relation 1. We verify that the two convolutions $y^{(\text{circ})}$ and y coincide only for the instants $k = N-1, N, \ldots, L-1$, and we write

$$y^{(\text{circ})}(k) = y(k) \qquad \text{only for } k = N-1, N, \ldots, L-1 \tag{1.112}$$

Indeed, with reference to Figure 1.14, the result of circular convolution coincides with $\{y(k)\}$, output of the linear convolution, only for a delay k such that it is avoided the product between non-zero samples of the two periodic sequences h_{rep_L} and x_{rep_L}, indicated by $\bullet$ and $\circ$, respectively. This is achieved only for $k \geq N-1$ and $k \leq L-1$.

[9] The notation $\text{diag}\{\boldsymbol{v}\}$ denotes a diagonal matrix whose elements on the diagonal are equal to the components of the vector $\boldsymbol{v}$.

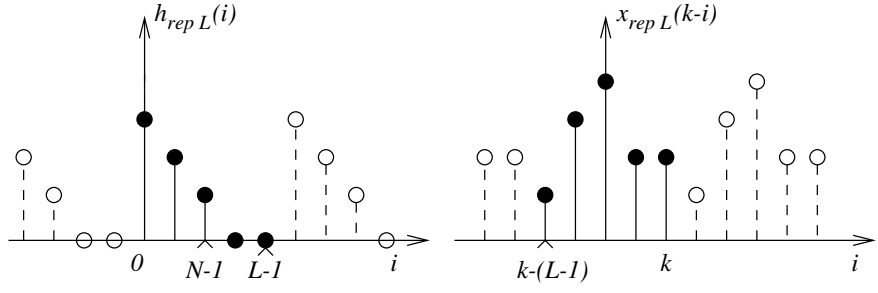


Figure 1.14. Illustration of the *circular convolution* operation between $\{x(k)\}$, $k = 0, 1, \ldots, L-1$, and $\{h(k)\}$, $k = 0, 1, \ldots, N-1$.

Relation 2. An alternative to (1.112) is to consider, instead of the finite sequence x, an extended sequence $x^{(px)}$ that is obtained by partially repeating x with a *cyclic prefix* of N_{px} samples:

$$x^{(px)}(k) = \begin{cases} x(k) & k = 0, 1, \ldots, L_x - 1 \\ x(L_x + k) & k = -N_{px}, \ldots, -2, -1 \end{cases} \tag{1.113}$$

Let $y^{(px)}$ be the *linear convolution* between $x^{(px)}$ and h, with support $\{-N_{px}, \ldots, L_x + N - 2\}$. If $N_{px} \geq N - 1$, it is easy to prove the following relation

$$y^{(px)}(k) = y^{(\text{circ})}(k) \qquad k = 0, 1, \ldots, L_x - 1 \tag{1.114}$$

Let us define

$$z(k) = \begin{cases} y^{(px)}(k) & k = 0, 1, \ldots, L_x - 1 \\ 0 & \text{elsewhere} \end{cases} \tag{1.115}$$

then from (1.114) and (1.106) the following relation between the corresponding L_x–point DFTs is obtained:

$$\mathcal{Z}_m = \mathcal{X}_m \mathcal{H}_m \qquad m = 0, 1, \ldots, L_x - 1 \tag{1.116}$$

Convolution by the overlap-save method

For a very long sequence x, the application of (1.112) leads to the *overlap-save* method to determine the linear convolution between x and h. It is not restrictive to assume that the first $(N-1)$ samples of the sequence $\{y(k)\}$ are zero. If this were not true it would be sufficient to shift the input by $(N-1)$ samples, and neglect the first $(N-1)$ samples of $\{y(k)\}$. Let us now subdivide the sequence $\{x(k)\}$ into blocks of L samples such that adjacent blocks are characterized by an overlapping of $(N-1)$ samples. A fast procedure to compute the linear convolution $\{y(k)\}$ for instants $k = N-1, N, \ldots, L-1$ is the following:[10]

[10] In this section the superscript $'$ indicates a vector of L components.

1. *Loading*

$$\boldsymbol{h}'^T = [h(0), h(1), \ldots, h(N-1), \overbrace{0, \ldots, 0}^{L-N \text{ zeros}}] \tag{1.117}$$

$$\boldsymbol{x}'^T = [x(0), x(1), \ldots, x(N-1), x(N), \ldots, x(L-1)] \tag{1.118}$$

in which we have assumed $x(k) = 0$, $k = 0, 1, \ldots, N-2$.

2. *Transform*

$$\boldsymbol{\mathcal{H}}' = \text{DFT}[\boldsymbol{h}'] \qquad \text{vector} \tag{1.119}$$

$$\boldsymbol{\mathcal{X}}' = \text{diag}\{\text{DFT}[\boldsymbol{x}']\} \qquad \text{matrix} \tag{1.120}$$

3. *Matrix product*

$$\boldsymbol{\mathcal{Y}}' = \boldsymbol{\mathcal{X}}'\boldsymbol{\mathcal{H}}' \qquad \text{vector} \tag{1.121}$$

4. *Inverse transform*

$$\boldsymbol{y}'^T = \text{DFT}^{-1}\left[\boldsymbol{\mathcal{Y}}'^T\right] = [\overbrace{\sharp, \ldots, \sharp}^{N-1 \text{ terms}}, y(N-1), y(N), \ldots, y(L-1)] \tag{1.122}$$

where the symbol $\sharp$ denotes a component that is neglected.

The second loading contains

$$\boldsymbol{x}'^T = [x((L-1)-(N-2)), \ldots, x(2(L-1)-(N-2))] \tag{1.123}$$

and the desired output samples will be

$$y(k) \qquad k = L, \ldots, 2(L-1)-(N-2) \tag{1.124}$$

The third loading contains

$$\boldsymbol{x}'^T = [x(2(L-1)-2(N-2)), \ldots, x(3(L-1)-2(N-2))] \tag{1.125}$$

and will yield the desired output samples

$$y(k) \qquad k = 2(L-1)-(N-2)+1, \ldots, 3(L-1)-2(N-2) \tag{1.126}$$

The algorithm proceeds until the entire input sequence is processed.

IIR and FIR filters

An important class of linear systems is identified by the input–output relation

$$\sum_{n=0}^{p} \mathtt{a}_n y(k-n) = \sum_{n=0}^{q} \mathtt{b}_n x(k-n) \tag{1.127}$$

where we can set $\mathtt{a}_0 = 1$ without loss of generality. If the system is causal, (1.127) becomes

$$y(k) = -\sum_{n=1}^{p} \mathtt{a}_n y(k-n) + \sum_{n=0}^{q} \mathtt{b}_n x(k-n) \qquad k \geq 0 \tag{1.128}$$

and the transfer function for such systems assumes the form

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\displaystyle\sum_{n=0}^{q} \mathtt{b}_n z^{-n}}{1 + \displaystyle\sum_{n=1}^{p} \mathtt{a}_n z^{-n}} = \frac{\mathtt{b}_0 \displaystyle\prod_{n=1}^{q} (1 - \mathtt{z}_n z^{-1})}{\displaystyle\prod_{n=1}^{p} (1 - \mathtt{p}_n z^{-1})} \tag{1.129}$$

where $\{\mathtt{z}_n\}$ and $\{\mathtt{p}_n\}$ are, respectively, the zeros and poles of $H(z)$. Equation (1.129) generally defines an *infinite impulse response* (IIR) filter. In the case in which $\mathtt{a}_n = 0$, $n = 1, 2, \ldots, p$, (1.129) reduces to

$$H(z) = \sum_{n=0}^{q} \mathtt{b}_n z^{-n} \tag{1.130}$$

and we obtain a *finite impulse response* (FIR) filter with $h(n) = \mathtt{b}_n$, $n = 0, 1, \ldots, q$. To get the impulse response coefficients, assuming known the z-transform $H(z)$, we can expand $H(z)$ in partial fractions and apply the linear property of the z-transform (see Table 1.3, page 19). If $q < p$ and assuming that all poles are distinct, we obtain

$$H(z) = \sum_{n=1}^{p} \frac{r_n}{1 - \mathtt{p}_n z^{-1}} \implies h(k) = \begin{cases} \displaystyle\sum_{n=1}^{p} r_n \mathtt{p}_n^k & k \geq 0 \\ 0 & k < 0 \end{cases} \tag{1.131}$$

where

$$r_n = H(z)[1 - \mathtt{p}_n z^{-1}]|_{z=\mathtt{p}_n} \tag{1.132}$$

We give now two definitions.

Definition 1.7
A causal system is *stable* (*bounded input-bounded output* stability) if $|\mathtt{p}_n| < 1$, $\forall n$.

Definition 1.8
The system is *minimum phase* (*maximum phase*) if $|\mathsf{p}_n| < 1$ and $|\mathsf{z}_n| \leq 1$ ($|\mathsf{p}_n| > 1$ and $|\mathsf{z}_n| > 1$), $\forall n$.

Among all systems having the same magnitude response $|\mathcal{H}(e^{j2\pi f T_c})|$, the minimum (maximum) phase system presents a phase response, $\arg\mathcal{H}(e^{j2\pi f T_c})$, which is below (above) the phase response of all other systems.

Example 1.4.3
It is interesting to determine the phase of a system for a given impulse response. Let us consider the system with transfer function $H_1(z)$ and impulse response $h_1(k)$ shown in Figure 1.15a. After determining the zeros of the transfer function, we factorize $H_1(z)$ as:

$$H_1(z) = \mathsf{b}_0 \prod_{n=1}^{4} (1 - \mathsf{z}_n z^{-1}) \tag{1.133}$$

As shown in Figure 1.15a, $H_1(z)$ is minimum phase. We now observe that the magnitude of the frequency response does not change if $1/z_n^*$ is replaced with z_n in (1.133). If we move all the zeros outside the unit circle, we get a maximum-phase system $H_2(z)$ whose impulse response is shown in Figure 1.15b. A general case, that is a transfer function with some zeros inside and others outside the unit circle, is given in Figure 1.15c. The coefficients of the impulse responses h_1, h_2, and h_3 are given in Table 1.4. The coefficients are normalized so that the three impulse responses have equal energy.

We define the *partial energy* of a causal impulse response as

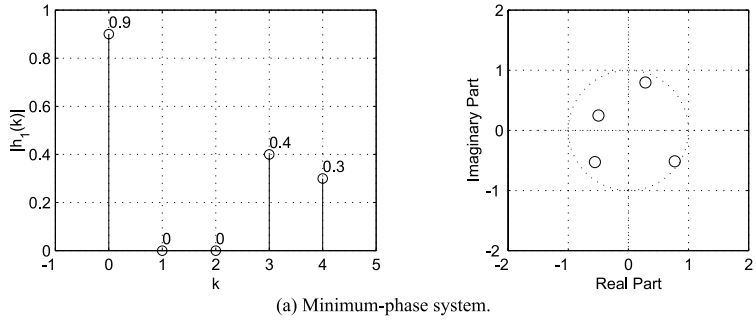
$$E(k) = \sum_{i=0}^{k} |h(i)|^2 \tag{1.134}$$

Comparing the partial-energy sequences for the three impulse responses of Figure 1.15, one finds that the minimum (maximum) phase system yields the largest (smallest) $\{E(k)\}$. In other words, the magnitude of the frequency responses being equal, a minimum (maximum) phase system concentrates all its energy on the first (last) samples of the impulse response.

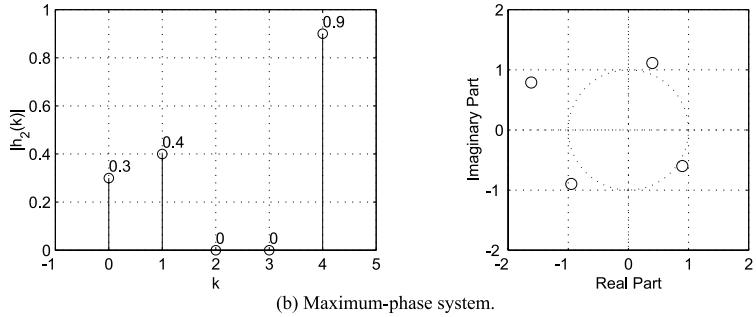
Extending our previous considerations also to IIR filters, if h_1 is a causal minimum-phase filter, i.e. $H_1(z) = H_{min}(z)$ is a ratio of polynomials in z^{-1} with poles and zeros

Table 1.4 Impulse responses of systems having the same magnitude of the frequency response.

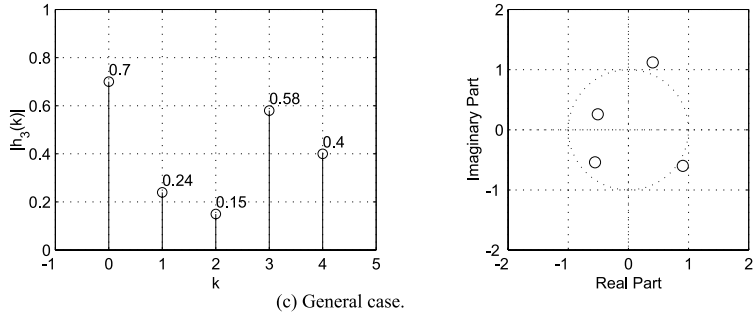
	$h(0)$	$h(1)$	$h(2)$	$h(3)$	$h(4)$
h_1 (minimum phase)	$0.9e^{-j1.57}$	0	0	$0.4e^{-j0.31}$	$0.3e^{-j0.63}$
h_2 (maximum phase)	$0.3e^{j0.63}$	$0.4e^{j0.31}$	0	0	$0.9e^{j1.57}$
h_3 (general case)	$0.7e^{-j1.57}$	$0.24e^{j2.34}$	$0.15e^{-j1.66}$	$0.58e^{-j0.51}$	$0.4e^{-j0.63}$



(a) Minimum-phase system.



(b) Maximum-phase system.



(c) General case.

Figure 1.15. Impulse response magnitudes and zero locations for three systems having the same frequency response magnitude.

inside the unit circle, then $H_{max}(z) = K\, H^*_{min}\left(\frac{1}{z^*}\right)$, where K is a constant, is an anti-causal maximum-phase filter, i.e. $H_{max}(z)$ is a ratio of polynomials in z with poles and zeros outside the unit circle.

In the case of a minimum-phase FIR filter with impulse response $h_{min}(n), n = 0, 1, \ldots, q$, $H_2(z) = z^{-q}\, H^*_{min}\left(\frac{1}{z^*}\right)$ is a causal maximum-phase filter. Moreover, the relation $\{h_2(n)\} = \{h_1^*(q-n)\}, n = 0, 1, \ldots, q$, is satisfied. In this text we use the notation $\{h_2(n)\} = \{h_1^{B*}(n)\}$, where B is the *backward* operator that orders the elements of a sequence from the last to the first.

In Appendix 1.A multirate transformations for systems are described, in which the time domain of the input is different from that of the output. In particular, decimator and interpolator filters are introduced, together with their efficient implementations.

1.5 Signal bandwidth

Definition 1.9
The *support* of a signal $x(\xi)$, $\xi \in \Re$, is the set of values $\xi \in \Re$ for which $|x(\xi)| \neq 0$.

Let us consider a filter with impulse response h and frequency response $\mathcal{H}$. If h assumes real values, then $\mathcal{H}$ is Hermitian, $\mathcal{H}(-f) = \mathcal{H}^*(f)$, and $|\mathcal{H}(f)|$ is an even function. Depending on the support of $|\mathcal{H}(f)|$, the classification of Figure 1.16 is usually done. If h assumes complex values, the terminology is less standard. We adopt the classification of Figure 1.17, in which the filter is a lowpass filter (LPF) if the support $|\mathcal{H}(f)|$ includes the origin, otherwise it is a passband filter (PBF).

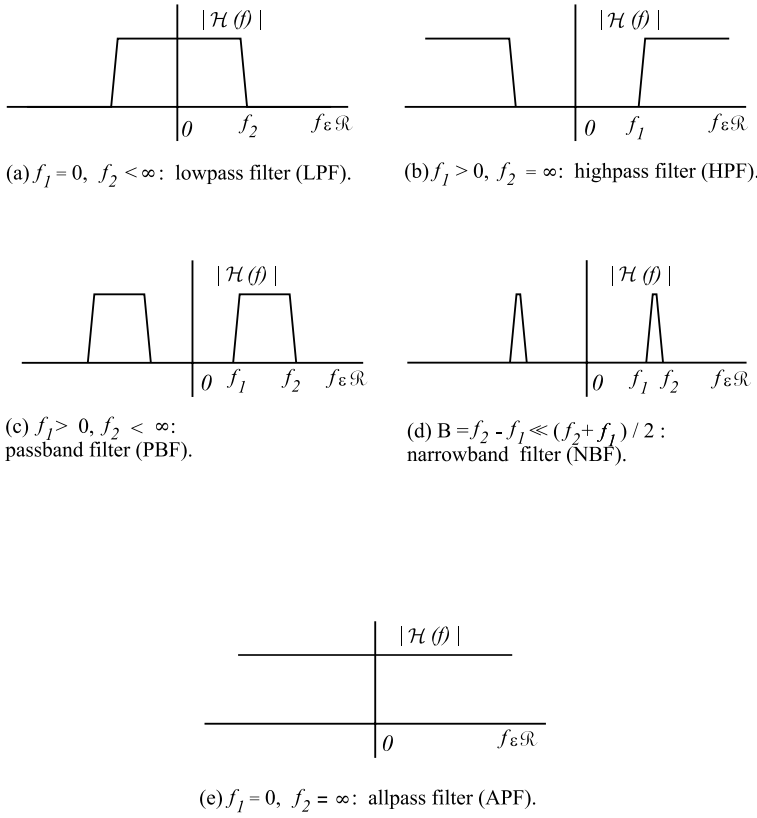


Figure 1.16. Classification of real valued analog filters on the basis of the support of $|\mathcal{H}(f)|$.

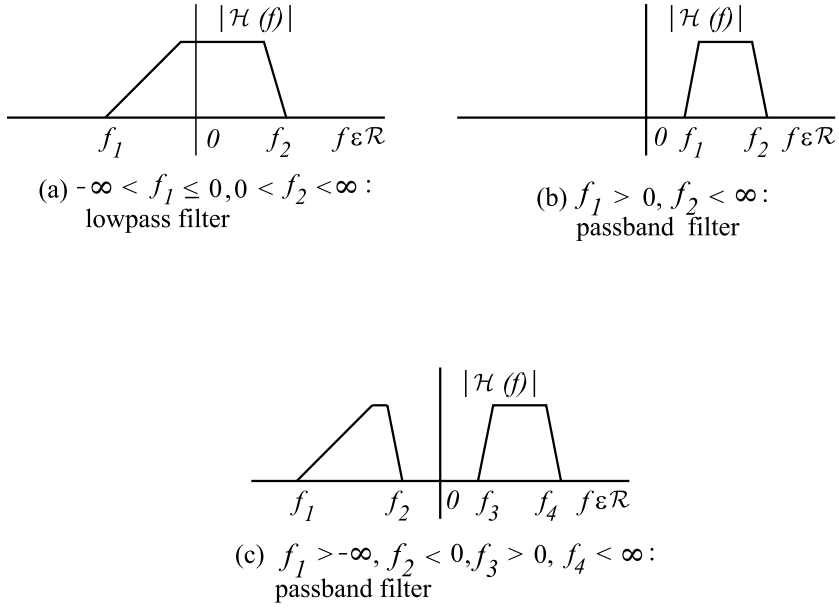


Figure 1.17. Classification of complex valued analog filters on the basis of support of $|\mathcal{H}(f)|$.

Analogously, for a signal x, we will use the same denomination and we will say that x is a baseband (BB) or passband (PB) signal depending on whether the support of $|\mathcal{X}(f)|$, $f \in \Re$, includes or not the origin.

Definition 1.10
In general, for a *real-valued signal* x, the set of positive frequencies such that $|\mathcal{X}(f)| \neq 0$ is called *passband* or simply *band* $\mathcal{B}$:

$$\mathcal{B} = \{f \geq 0 : |\mathcal{X}(f)| \neq 0\} \tag{1.135}$$

As $|\mathcal{X}(f)|$ is an even function, we have $|\mathcal{X}(-f)| \neq 0$, $f \in \mathcal{B}$. We note that $\mathcal{B}$ is equivalent to the support of $\mathcal{X}$ limited to positive frequencies. The bandwidth[11] of x is given by the measure of $\mathcal{B}$:

$$B = \int_{\mathcal{B}} df \tag{1.140}$$

[11] The signal *bandwidth* may be given different definitions. Let us consider an LPF having frequency response $\mathcal{H}(f)$. The filter gain $\mathcal{H}_0$ is usually defined as $\mathcal{H}_0 = |\mathcal{H}(0)|$; other definitions are as average gain of the filter in the passband $\mathcal{B}$, or as $\max_f |\mathcal{H}(f)|$. We give the following four definitions for the bandwidth B of h.

a) *First zero*:

$$B = \min\{f > 0 : \mathcal{H}(f) = 0\} \tag{1.136}$$

For example, with regard to the signals of Figure 1.16, we have that for an LPF $B = f_2$, whereas for a PBF $B = f_2 - f_1$. In the case of a *complex-valued signal* x, $\mathcal{B}$ is equivalent to the support of $\mathcal{X}$, and B is thus given by the measure of the entire support.

For discrete-time filters, for which $\mathcal{H}$ is periodic of period $1/T_c$, the same definitions hold, with the caution of considering the support of $|\mathcal{H}(f)|$ within a period, let's say between $-1/(2T_c)$ and $1/(2T_c)$. In the case of discrete-time highpass filters (HPF), the passband will extend from a certain frequency f_1 to $1/(2T_c)$.

As discrete-time signals are often obtained by sampling continuous-time signals, we will state the following fundamental theorem.

The sampling theorem

Let $q(t)$, $t \in \Re$ be a continuous-time signal, in general complex-valued, whose Fourier transform $\mathcal{Q}(f)$ has support within an interval $\mathcal{B}$ of finite measure B_0. The samples of the signal $q(t)$, taken with period T_c,

$$h_k = q(kT_c) \tag{1.141}$$

univocally represent the signal $q(t)$, $t \in \Re$, under the condition that the *sampling frequency* $1/T_c$ satisfies the relation

$$\frac{1}{T_c} \geq B_0 \tag{1.142}$$

For the proof, which is based on the relation (1.90) between a signal and its samples, we refer the reader to [1].

B_0 is often referred to as the minimum sampling frequency. If $1/T_c < B_0$ the signal cannot be perfectly reconstructed from its samples, originating the so-called *aliasing phenomenon* in the frequency-domain signal representation.

---

b) *Based on amplitude*, bandwidth at A dB:

$$B = \max\left\{ f > 0 : \frac{|\mathcal{H}(f)|}{\mathcal{H}_0} = 10^{-\frac{A}{20}} \right\} \tag{1.137}$$

Typically $A = 3, 40$, or 60.

c) *Based on energy*, bandwidth at $p\%$:

$$\frac{\int_0^B |\mathcal{H}(f)|^2 \, df}{\int_0^\infty |\mathcal{H}(f)|^2 \, df} = \frac{p}{100} \tag{1.138}$$

Typically $p = 90$ or 99.

d) *Equivalent noise bandwidth*:

$$B = \frac{\int_0^\infty |\mathcal{H}(f)|^2 \, df}{\mathcal{H}_0^2} \tag{1.139}$$

Figure 1.18 illustrates the various definitions for a particular $|\mathcal{H}(f)|$.

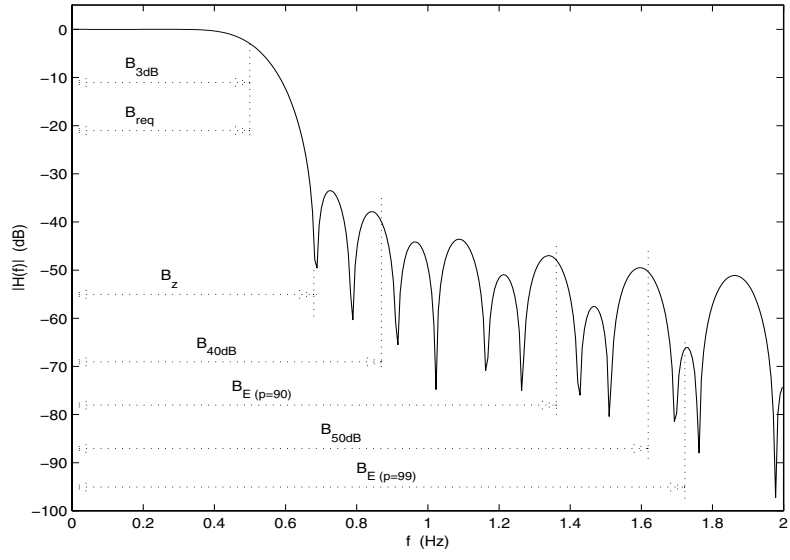


Figure 1.18. The real signal bandwidth following the definitions of:
1) Bandwidth at first zero: $B_Z = 0.652$ Hz.
2) Amplitude-based bandwidth: $B_{3\,dB} = 0.5$ Hz, $B_{40\,dB} = 0.87$ Hz, $B_{50\,dB} = 1.62$ Hz.
3) Energy-based bandwidth: $B_{E(p=90)} = 1.362$ Hz, $B_{E(p=99)} = 1.723$ Hz.
4) Equivalent noise bandwidth: $B_{\text{req}} = 0.5$ Hz.

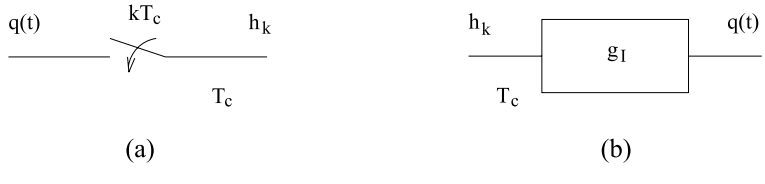


Figure 1.19. Operation of (a) sampling and (b) interpolation.

In turn, the signal $q(t)$, $t \in \Re$, can be reconstructed from its samples $\{h_k\}$ according to the scheme of Figure 1.19, where it is employed as an interpolation filter having an ideal frequency response given by

$$\mathcal{G}_I(f) = \begin{cases} 1 & f \in \mathcal{B} \\ 0 & \text{elsewhere} \end{cases} \tag{1.143}$$

We note that for *real-valued baseband signals* $B_0 = 2B$. For passband signals, care must be taken in the choice of $B_0 \geq 2B$ to avoid *aliasing* between the positive and negative frequency components of $\mathcal{Q}(f)$.

Heaviside conditions for the absence of signal distortion

Let us consider a filter having frequency response $\mathcal{H}(f)$ (see Figure 1.10 or Figure 1.12) given by

$$\mathcal{H}(f) = \mathcal{H}_0 e^{-j2\pi f t_0} \qquad f \in \mathcal{B} \tag{1.144}$$

where $\mathcal{H}_0$ and t_0 are two non-negative constants, and $\mathcal{B}$ is the *passband* of the filter input signal x. Then the output is given by

$$\mathcal{Y}(f) = \mathcal{H}(f)\mathcal{X}(f) = \mathcal{H}_0 \mathcal{X}(f)\, e^{-j2\pi f t_0} \tag{1.145}$$

or, in the time domain,

$$y(t) = \mathcal{H}_0 x(t - t_0) \tag{1.146}$$

In other words, for a filter of the type (1.144), the signal at the input is reproduced at the output with a gain factor $\mathcal{H}_0$ and a delay t_0.

A filter of the type (1.144) satisfies the *Heaviside conditions for the absence of signal distortion* and is characterized by

1. *constant magnitude*

$$|\mathcal{H}(f)| = \mathcal{H}_0 \qquad f \in \mathcal{B} \tag{1.147}$$

2. *linear phase*[12]

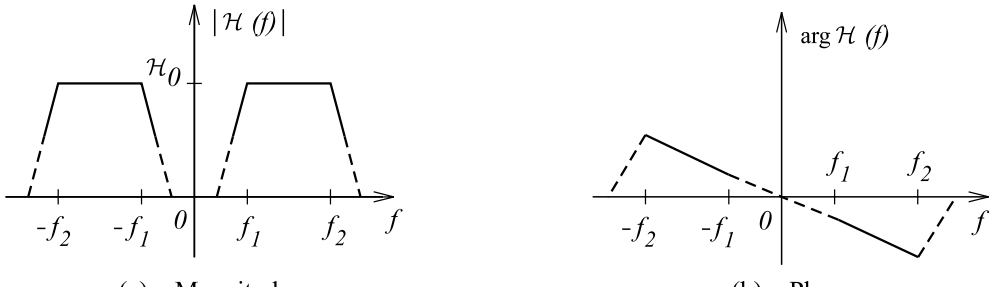
$$\arg \mathcal{H}(f) = -2\pi f t_0 \qquad f \in \mathcal{B} \tag{1.148}$$

3. *constant group delay*, also called *envelope delay*

$$\tau(f) = -\frac{1}{2\pi}\frac{d}{df} \arg \mathcal{H}(f) = t_0 \qquad f \in \mathcal{B} \tag{1.149}$$

We emphasize that it is sufficient that the Heaviside conditions are verified within the support of $\mathcal{X}$; as $|\mathcal{X}(f)| = 0$ outside the support, the filter frequency response may be arbitrary.

We show in Figure 1.20 the frequency response of a PBF, with bandwidth $B = f_2 - f_1$, that satisfies the conditions stated by Heaviside.



(a) Magnitude (b) Phase

Figure 1.20. Characteristics of a filter satisfying the conditions for the absence of signal distortion in the frequency interval (f_1, f_2).

[12] For a complex number c, $\arg c$ denotes the phase of c (see note 3, page 441).

1.6 Passband signals

Complex representation

For a passband signal x it is convenient to introduce an equivalent representation in terms of a baseband signal $x^{(bb)}$.

Let x be a PB real-valued signal with Fourier transform as illustrated in Figure 1.21. The following two procedures can be adopted to obtain $x^{(bb)}$.

PB filter. Referring to Figure 1.21 and to the transformations illustrated in Figure 1.22, given x we extract its positive frequency components using an *analytic filter* or *phase splitter*, $h^{(a)}$, having the following ideal frequency response

$$\mathcal{H}^{(a)}(f) = 2 \cdot 1(f) = \begin{cases} 2 & f > 0 \\ 0 & f < 0 \end{cases} \tag{1.150}$$

In practice, it is sufficient that $h^{(a)}$ be a complex PB filter, with passband, in which $\mathcal{H}^{(a)}(f) \simeq 2$, that extends from f_1 to f_2, equal to that of $\mathcal{X}(f)$, and stopband, in which $|\mathcal{H}^{(a)}(f)| \simeq 0$, that extends from $-f_2$ to $-f_1$. The signal $x^{(a)}$ is called the analytic signal or pre-envelope of x.

It is now convenient to introduce a suitable frequency f_0, called *reference carrier frequency*, which usually belongs to the passband (f_1, f_2) of x. The filter output, $x^{(a)}$, is

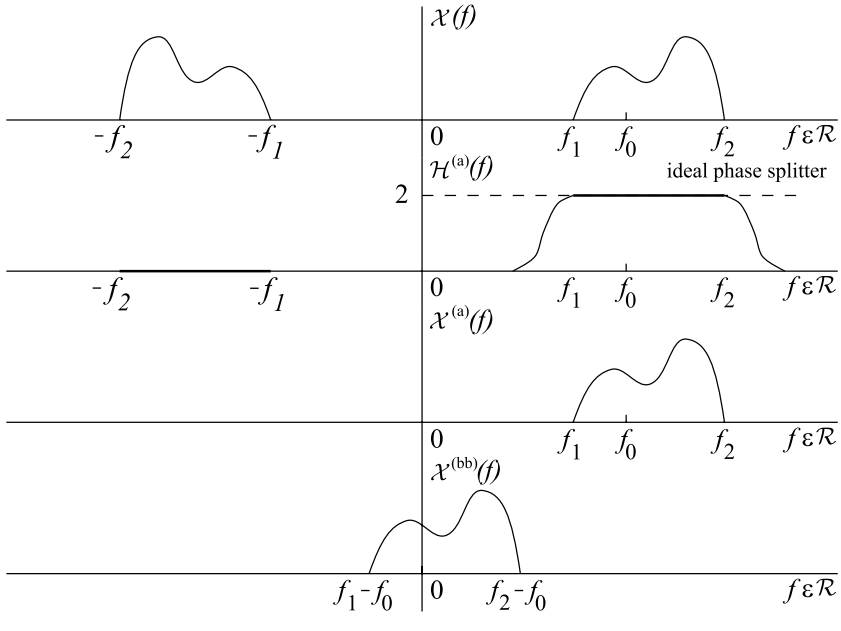


Figure 1.21. Illustration of transformations to obtain the baseband equivalent signal $x^{(bb)}$ around the carrier frequency f_0 using a phase splitter.

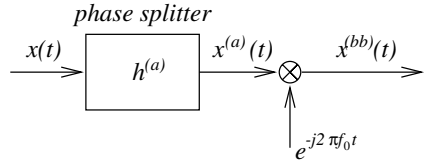


Figure 1.22. Transformations to obtain the baseband equivalent signal $x^{(bb)}$ around the carrier frequency f_0 using a phase splitter.

frequency shifted by f_0 to obtain a BB signal, $x^{(bb)}$. The signal $x^{(bb)}$ is the *baseband equivalent* of x, also called *complex envelope of x around the carrier frequency* f_0.

Analytically, we have

$$x^{(a)}(t) = x * h^{(a)}(t) \qquad \xleftrightarrow{\mathcal{F}} \quad \mathcal{X}^{(a)}(f) = \mathcal{X}(f)\mathcal{H}^{(a)}(f) \tag{1.151}$$

$$x^{(bb)}(t) = x^{(a)}(t)\, e^{-j2\pi f_0 t} \xleftrightarrow{\mathcal{F}} \mathcal{X}^{(bb)}(f) = \mathcal{X}^{(a)}(f + f_0) \tag{1.152}$$

and in the frequency domain

$$\mathcal{X}^{(bb)}(f) = \begin{cases} 2\mathcal{X}(f + f_0) & \text{for } f > -f_0 \\ 0 & \text{for } f < -f_0 \end{cases} \tag{1.153}$$

In other words, $x^{(bb)}$ is given by the components of x at positive frequencies, scaled by 2 and frequency shifted by f_0.

BB filter. One gets the same result using a frequency shift of x followed by a lowpass filter (see Figures 1.23 and 1.24). It is immediate to determine the relation between the frequency responses of the filters of Figure 1.21 and Figure 1.23:

$$\mathcal{H}(f) = \mathcal{H}^{(a)}(f + f_0) \tag{1.154}$$

From (1.154) one can derive the relation between the impulse response of the analytic filter and the impulse response of the lowpass filter:

$$h^{(a)}(t) = h(t)\, e^{j2\pi f_0 t} \tag{1.155}$$

Relation between x and $x^{(bb)}$

A simple analytical relation exists between a *real signal* x and its complex envelope. In fact, making use of the property $\mathcal{X}(-f) = \mathcal{X}^*(f)$, it follows

$$\mathcal{X}(f) = \mathcal{X}(f)1(f) + \mathcal{X}(f)1(-f) = \mathcal{X}(f)1(f) + \mathcal{X}^*(-f)1(-f) \tag{1.156}$$

or, equivalently,

$$x(t) = \frac{x^{(a)}(t) + x^{(a)*}(t)}{2} = Re[x^{(a)}(t)] \tag{1.157}$$

Using (1.152) it also follows

$$x(t) = Re[x^{(bb)}(t)e^{j2\pi\, f_0 t}] \tag{1.158}$$

as illustrated in Figure 1.25.

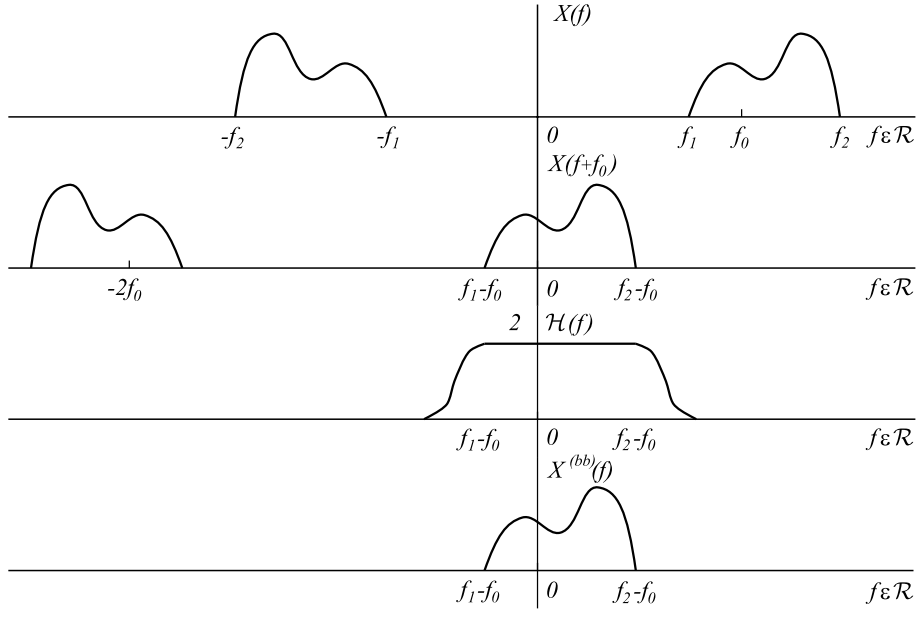


Figure 1.23. Illustration of transformations to obtain the baseband equivalent signal $x^{(bb)}$ around the carrier frequency f_0 using a lowpass filter.

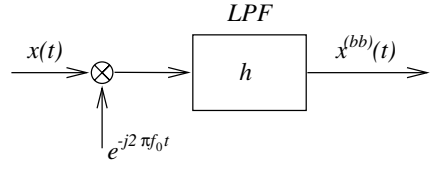


Figure 1.24. Transformations to obtain the baseband equivalent signal $x^{(bb)}$ around the carrier frequency f_0 using a lowpass filter.

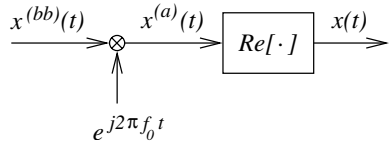


Figure 1.25. Relation between a signal, its complex envelope and the analytic signal.

Baseband components of a PB signal. We introduce the notation

$$x^{(bb)}(t) = x_I^{(bb)}(t) + jx_Q^{(bb)}(t) \tag{1.159}$$

where

$$x_I^{(bb)}(t) = Re[x^{(bb)}(t)] \tag{1.160}$$

and

$$x_Q^{(bb)}(t) = Im[x^{(bb)}(t)] \tag{1.161}$$

are real-valued baseband signals, called *in-phase* and *quadrature components* of x, respectively. Substituting (1.159) in (1.158) we obtain

$$x(t) = x_I^{(bb)}(t)\cos 2\pi f_0 t - x_Q^{(bb)}(t)\sin 2\pi f_0 t \tag{1.162}$$

as illustrated in Figure 1.26.

Conversely, given x, one can use the scheme of Figure 1.24 and the relations (1.160) and (1.161) to get the baseband components. If the frequency response $\mathcal{H}(f)$ has Hermitian-symmetric characteristics with respect to the origin, h is real and the scheme of Figure 1.27a holds. The scheme of Figure 1.27b employs instead an ideal Hilbert filter with frequency response given by

$$\mathcal{H}^{(h)}(f) = -j\,\mathrm{sgn}(f) = e^{-j\frac{\pi}{2}\mathrm{sgn}(f)} \tag{1.163}$$

Magnitude and phase of $\mathcal{H}^{(h)}(f)$ are shown in Figure 1.28. We note that $h^{(h)}$ phase-shifts by $-\pi/2$ the positive-frequency components of the input and by $\pi/2$ the negative-frequency components. In practice these filter specifications are imposed only on the passband of the input signal.[13] To simplify the notation, in block diagrams a Hilbert filter is indicated as "$-\pi/2$".

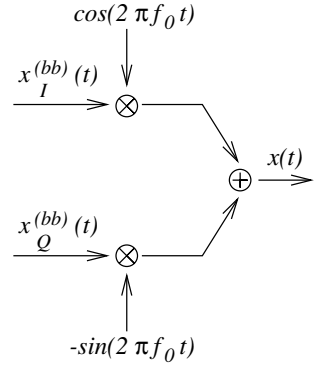
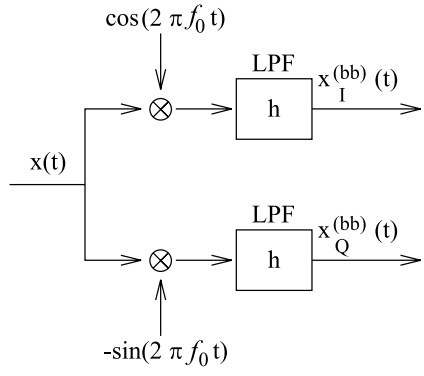


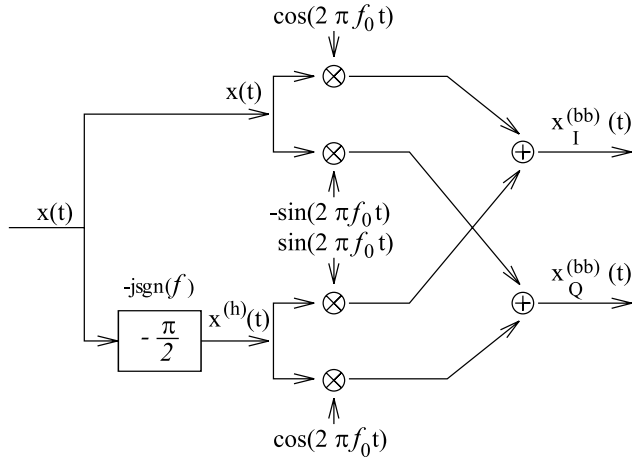
Figure 1.26. Relation between a signal and its baseband components.

[13] We note that the ideal Hilbert filter in Figure 1.28 has an impulse response given by (see Table 1.2 on page 17):

$$h^{(h)}(t) = \frac{1}{\pi t} \tag{1.164}$$



(a) Implementation using LPF.



(b) Implementation using Hilbert filter.

Figure 1.27. Relations to derive the baseband signal components.

Comparing the frequency responses of the analytic filter (1.150) and of the Hilbert filter (1.163), we obtain the relation

$$\mathcal{H}^{(a)}(f) = 1 + j\mathcal{H}^{(h)}(f) \tag{1.167}$$

Consequently, if x is the input signal, the output of the Hilbert filter (also denoted as Hilbert transform of x) is

$$x^{(h)}(t) = \frac{1}{\pi} \int_{-\infty}^{+\infty} \frac{x(\tau)}{t-\tau}\, d\tau \tag{1.165}$$

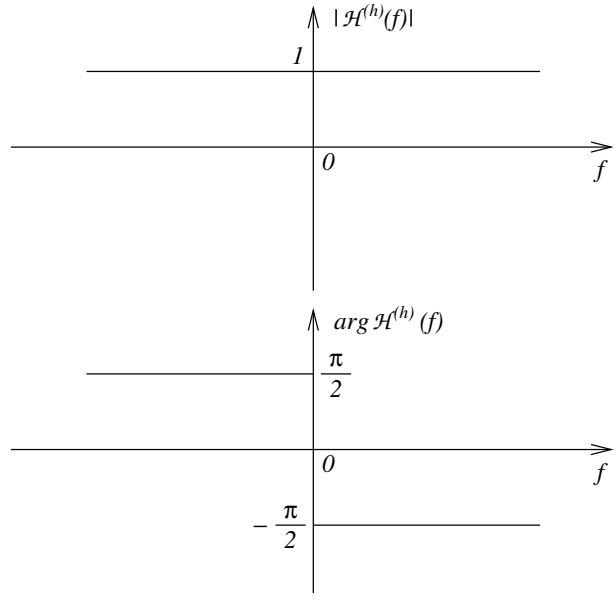


Figure 1.28. Magnitude and phase responses of the ideal Hilbert filter.

Then, letting

$$x^{(h)}(t) = x * h^{(h)}(t) \tag{1.168}$$

the analytic signal can be expressed as

$$x^{(a)}(t) = x(t) + jx^{(h)}(t) \tag{1.169}$$

Consequently, from (1.152), (1.160) and (1.161):

$$x_I^{(bb)}(t) = x(t)\cos 2\pi f_0 t + x^{(h)}(t)\sin 2\pi f_0 t \tag{1.170}$$

$$x_Q^{(bb)}(t) = x^{(h)}(t)\cos 2\pi f_0 t - x(t)\sin 2\pi f_0 t \tag{1.171}$$

as illustrated in Figure 1.27b.[14]

We note that in practical *systems*, transformations to obtain, e.g., the analytic signal, the complex envelope, or the Hilbert transform of a given signal, are implemented by

Moreover, noting that from (1.163) $(-j\,\mathrm{sgn}\,f)(-j\,\mathrm{sgn}\,f) = -1$, taking the Hilbert transform of a signal we get the initial signal with the sign changed. Then it results:

$$x(t) = -\frac{1}{\pi}\int_{-\infty}^{+\infty}\frac{x^{(h)}(\tau)}{t-\tau}\,d\tau \tag{1.166}$$

[14] We recall that the design of a filter, and in particular of a Hilbert filter, requires the introduction of a suitable delay. In other words, we are only able to produce an output with a delay t_D, $x^{(h)}(t - t_D)$. Consequently, in the block diagram of Figure 1.27, also $x(t)$ and the various sinusoidal waveforms must be delayed.

filtering operations. However, it is usually more convenient to perform signal *analysis* in the frequency domain by the Fourier transform. In the following two examples we use frequency-domain techniques to obtain the complex envelope of a PB signal.

Example 1.6.1
Let $x(t)$ be a sinusoidal signal,

$$x(t) = A\cos(2\pi f_0 t + \varphi_0) \tag{1.172}$$

Then

$$\mathcal{X}(f) = \frac{A}{2}e^{j\varphi_0}\delta(f - f_0) + \frac{A}{2}e^{-j\varphi_0}\delta(f + f_0) \tag{1.173}$$

The analytic signal is given by:

$$\mathcal{X}^{(a)}(f) = Ae^{j\varphi_0}\delta(f - f_0) \xleftrightarrow{\mathcal{F}^{-1}} x^{(a)}(t) = Ae^{j\varphi_0}e^{j2\pi f_0 t} \tag{1.174}$$

and

$$\mathcal{X}^{(bb)}(f) = Ae^{j\varphi_0}\delta(f) \xleftrightarrow{\mathcal{F}^{-1}} x^{(bb)}(t) = Ae^{j\varphi_0} \tag{1.175}$$

We note that we have chosen as reference carrier frequency of the complex envelope the same carrier frequency as in (1.172).

Example 1.6.2
Let

$$x(t) = A\,\mathrm{sinc}(Bt)\cos(2\pi f_0 t) \tag{1.176}$$

with the Fourier transform given by

$$\mathcal{X}(f) = \frac{A}{2B}\left[\mathrm{rect}\left(\frac{f - f_0}{B}\right) + \mathrm{rect}\left(\frac{f + f_0}{B}\right)\right] \tag{1.177}$$

as illustrated in Figure 1.29. Then, using f_0 as reference carrier frequency,

$$\mathcal{X}^{(bb)}(f) = \frac{A}{B}\,\mathrm{rect}\left(\frac{f}{B}\right) \tag{1.178}$$

and

$$x^{(bb)}(t) = A\,\mathrm{sinc}(Bt) \tag{1.179}$$

Another *analytical technique* to get the expression of the signal after the various transformations is obtained by applying the following theorem.

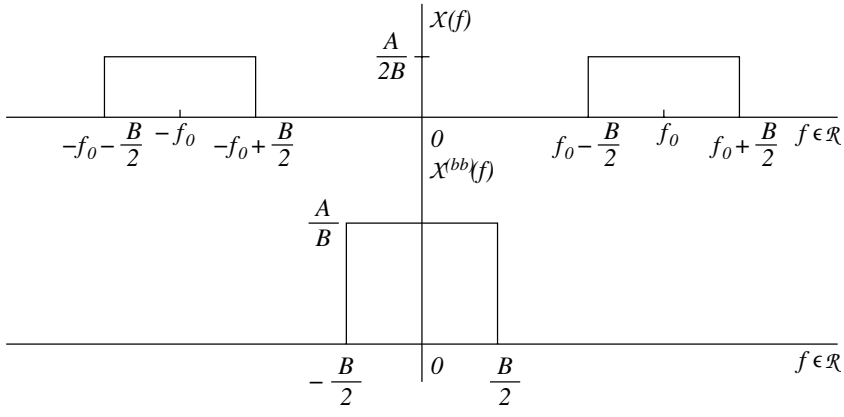


Figure 1.29. Frequency response of a PB signal and corresponding complex envelope.

Theorem 1.1
Let the *product of two real signals* be

$$x(t) = a(t)\, c(t) \tag{1.180}$$

where a is a BB signal with $\mathcal{B}_a = [0, B)$ and c is a PB signal with $\mathcal{B}_c = [f_0, +\infty)$. If $f_0 > B$, then the analytic signal of x is related to that of c by:

$$x^{(a)}(t) = a(t)\, c^{(a)}(t) \tag{1.181}$$

Proof. We consider the general relation (1.157), valid for every real signal

$$c(t) = \tfrac{1}{2}\, c^{(a)}(t) + \tfrac{1}{2}\, c^{(a)*}(t) \tag{1.182}$$

Substituting (1.182) in (1.180) yields

$$x(t) = a(t)\tfrac{1}{2}c^{(a)}(t) + a(t)\tfrac{1}{2}c^{(a)*}(t) \tag{1.183}$$

In the frequency domain the support of the first term in (1.183) is given by the interval $[f_0 - B, +\infty)$, while that of the second is equal to $(-\infty, -f_0 + B]$. Under the hypothesis that $f_0 \geq B$, the two terms in (1.183) have disjoint support in the frequency domain and (1.181) is immediately obtained.

Corollary 1.1
From (1.181) we obtain

$$x^{(h)}(t) = a(t)c^{(h)}(t) \tag{1.184}$$

and

$$x^{(bb)}(t) = a(t)c^{(bb)}(t) \tag{1.185}$$

In fact, from (1.169) we get

$$x^{(h)}(t) = Im[x^{(a)}(t)] \tag{1.186}$$

which substituted in (1.181) yields (1.184). Finally, (1.185) is obtained by substituting (1.152),

$$x^{(bb)}(t) = x^{(a)}(t)e^{-j2\pi f_0 t} \tag{1.187}$$

in (1.181).

An interesting application of (1.186) is in the design of a Hilbert filter $h^{(h)}$ starting from a lowpass filter h. In fact, from (1.155) and (1.186) we get

$$h^{(h)}(t) = h(t)\sin(2\pi f_0 t) \tag{1.188}$$

Example 1.6.3
Let a modulated *double sideband* (DSB) signal be expressed as

$$x(t) = a(t)\cos(2\pi f_0 t + \varphi_0) \tag{1.189}$$

where a is a BB signal with bandwidth B. Then, if $f_0 > B$, from the above theorem we have the following relations:

$$x^{(a)}(t) = a(t)e^{j(2\pi f_0 t + \varphi_0)} \tag{1.190}$$

$$x^{(h)}(t) = a(t)\sin(2\pi f_0 t + \varphi_0) \tag{1.191}$$

$$x^{(bb)}(t) = a(t)e^{j\varphi_0} \tag{1.192}$$

We list in Table 1.5 some properties of the Hilbert transformation (1.168) that are easily obtained by using the Fourier transform and the properties of Table 1.1.

Table 1.5 Some properties of the Hilbert transform.

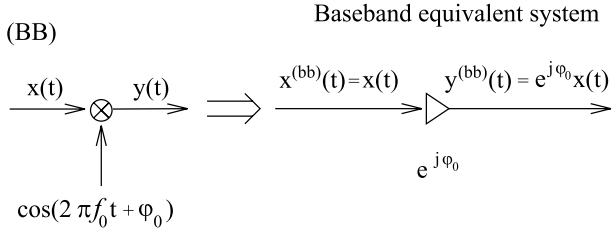
Property	(Real) signal	(Real) Hilbert transform
	$x(t)$	$x^{(h)}(t)$
duality	$x^{(h)}(t)$	$-x(t)$
inverse time	$x(-t)$	$-x^{(h)}(-t)$
even signal	$x(t) = x(-t)$	$x^{(h)}(t) = -x^{(h)}(-t)$, odd
odd signal	$x(t) = -x(-t)$	$x^{(h)}(t) = x^{(h)}(-t)$, even
product (see Theorem 1.1)	$a(t)\,c(t)$	$a(t)\,c^{(h)}(t)$
cosinusoidal signal	$\cos(2\pi f_0 t + \varphi_0)$	$\sin(2\pi f_0 t + \varphi_0)$
energy	$E_x = \int_{-\infty}^{+\infty} \lvert x(t)\rvert^2\, dt = \int_{-\infty}^{+\infty} \lvert x^{(h)}(t)\rvert^2\, dt = E_{x^{(h)}}$	
orthogonality	$\langle x, x^{(h)}\rangle = \int_{-\infty}^{+\infty} x(t)\, x^{(h)}(t)\, dt = 0$	

Baseband equivalent of a transformation

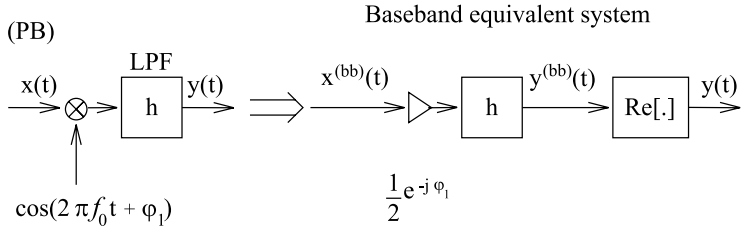
Given a transformation involving also passband signals, it is often useful to determine an equivalent relation between baseband complex representations of input and output signals. Three transformations are given in Figure 1.30, together with their baseband equivalent.

We will prove the relation illustrated in Figure 1.30b. Assuming h is the real-valued impulse response of an LPF and using (1.158),

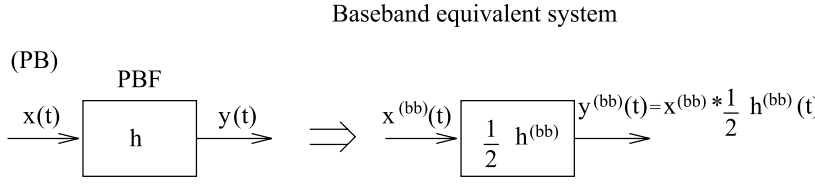
$$y(t) = \{h(\tau) * Re[x^{(bb)}(\tau)e^{j2\pi f_0\tau}(\cos(2\pi f_0\tau + \varphi_1))]\}(t)$$



(a) Modulator



(b) Demodulator



(c) Passband filtering

Figure 1.30. Passband transformations and their baseband equivalent.

$$= Re\left[\left(h(\tau) * x^{(bb)}(\tau)\frac{e^{-j\varphi_1}}{2} + h(\tau) * x^{(bb)}(\tau)\frac{e^{+j(2\pi 2 f_0\tau+\varphi_1)}}{2}\right)(t)\right] \quad (1.193)$$

$$= Re\left[\left(h * x^{(bb)}\frac{e^{-j\varphi_1}}{2}\right)(t)\right]$$

where the last equality follows because the term with frequency components around $2f_0$ is filtered by the LPF.

We note, moreover, that the filter $h^{(bb)}$ in Figure 1.30c has in-phase component $h_I^{(bb)}$ and quadrature component $h_Q^{(bb)}$ that are related to $\mathcal{H}^{(a)}$ by (see (1.160) and (1.161))

$$\begin{aligned}\mathcal{H}_I^{(bb)}(f) &= \tfrac{1}{2}[\mathcal{H}^{(bb)}(f) + \mathcal{H}^{(bb)*}(-f)] \\ &= \tfrac{1}{2}[\mathcal{H}^{(a)}(f+f_0) + \mathcal{H}^{(a)*}(-f+f_0)]\end{aligned} \quad (1.194)$$

and

$$\begin{aligned}\mathcal{H}_Q^{(bb)}(f) &= \frac{1}{2j}[\mathcal{H}^{(bb)}(f) - \mathcal{H}^{(bb)*}(-f)] \\ &= \frac{1}{2j}[\mathcal{H}^{(a)}(f+f_0) - \mathcal{H}^{(a)*}(-f+f_0)]\end{aligned} \quad (1.195)$$

Consequently, if $\mathcal{H}^{(a)}$ has Hermitian symmetry around f_0 then

$$\mathcal{H}_I^{(bb)}(f) = \mathcal{H}_a^{(a)}(f+f_0)$$

and

$$\mathcal{H}_Q^{(bb)}(f) = 0$$

In other words $h^{(bb)}(t) = h_I^{(bb)}(t)$ is real and the realization of the filter $\frac{1}{2}h^{(bb)}$ is simplified. In practice this condition is verified by ensuring that the filter $h^{(a)}$ has symmetrical frequency specifications around f_0.

Envelope and instantaneous phase and frequency

We will conclude this section with a few definitions.
Given a PB signal $x(t)$, with reference to the analytic signal we define:

1. *Envelope*

$$M_x(t) = |x^{(a)}(t)| \quad (1.196)$$

2. *Instantaneous phase*

$$\varphi_x(t) = \arg x^{(a)}(t) \quad (1.197)$$

3. *Instantaneous frequency*

$$f_x(t) = \frac{1}{2\pi}\frac{d}{dt}\varphi_x(t) \tag{1.198}$$

In terms of the complex envelope signal $x^{(bb)}$, from (1.152) the equivalent relations follow:

$$M_x(t) = |x^{(bb)}(t)| \tag{1.199}$$

$$\varphi_x(t) = \arg x^{(bb)}(t) + 2\pi f_0 t \tag{1.200}$$

$$f_x(t) = \frac{1}{2\pi}\frac{d}{dt}[\arg x^{(bb)}(t)] + f_0 \tag{1.201}$$

Then, from the polar representation $x^{(a)}(t) = M_x(t)\, e^{j\varphi_x(t)}$ and from (1.157), a PB signal x can be written as

$$x(t) = Re[x^{(a)}(t)] = M_x(t)\cos(\varphi_x(t)) \tag{1.202}$$

Two simplified methods to get the envelope $M_x(t)$ from the PB signal $x(t)$ are given in Figure 6.58 on page 514. For example if $x(t) = A\cos(2\pi f_0 t + \varphi_0)$ it follows that

$$M_x(t) = A \tag{1.203}$$

$$\varphi_x(t) = 2\pi f_0 t + \varphi_0 \tag{1.204}$$

$$f_x(t) = f_0 \tag{1.205}$$

With reference to the above relations, three other definitions follow.

1. *Envelope deviation*

$$\Delta M_x(t) = |x^{(a)}(t)| - A = |x^{(bb)}(t)| - A \tag{1.206}$$

2. *Phase deviation*

$$\Delta\varphi_x(t) = \varphi_x(t) - (2\pi f_0 t + \varphi_0) = \arg x^{(bb)}(t) - \varphi_0 \tag{1.207}$$

3. *Frequency deviation*

$$\Delta f_x(t) = f_x(t) - f_0 = \frac{1}{2\pi}\frac{d}{dt}\Delta\varphi_x(t) \tag{1.208}$$

Then (1.202) becomes

$$x(t) = [A + \Delta M_x(t)]\cos(2\pi f_0 t + \varphi_0 + \Delta\varphi_x(t)) \tag{1.209}$$

1.7 Second-order analysis of random processes

We recall the functions related to the statistical description of random processes, especially those functions concerning second-order analysis.

1.7.1 Correlation

Let $x(t)$ and $y(t)$, $t \in \Re$, be two continuous-time random processes. We indicate the delay or lag with τ and the expectation operator with E.

1. *Mean value*

$$\mathtt{m}_x(t) = E[x(t)] \tag{1.210}$$

2. *Statistical power*

$$\mathtt{M}_x(t) = E[|x(t)|^2] \tag{1.211}$$

3. *Autocorrelation*

$$\mathtt{r}_x(t, t-\tau) = E[x(t)x^*(t-\tau)] \tag{1.212}$$

4. *Cross-correlation*

$$\mathtt{r}_{xy}(t, t-\tau) = E[x(t)y^*(t-\tau)] \tag{1.213}$$

5. *Autocovariance*

$$\begin{aligned} \mathtt{c}_x(t, t-\tau) &= E[(x(t) - \mathtt{m}_x(t))(x(t-\tau) - \mathtt{m}_x(t-\tau))^*] \\ &= \mathtt{r}_x(t, t-\tau) - \mathtt{m}_x(t)\mathtt{m}_x^*(t-\tau) \end{aligned} \tag{1.214}$$

6. *Cross-covariance*

$$\begin{aligned} \mathtt{c}_{xy}(t, t-\tau) &= E[(x(t) - \mathtt{m}_x(t))(y(t-\tau) - \mathtt{m}_y(t-\tau))^*] \\ &= \mathtt{r}_{xy}(t, t-\tau) - \mathtt{m}_x(t)\mathtt{m}_y^*(t-\tau) \end{aligned} \tag{1.215}$$

Observation 1.4

- x and y are *orthogonal* if $\mathtt{r}_{xy}(t, t-\tau) = 0$, $\forall t, \tau$. In this case we write $x \perp y$.[15]
- x and y are *uncorrelated* if $\mathtt{c}_{xy}(t, t-\tau) = 0$, $\forall t, \tau$.
- if at least one of the two random processes has zero mean, *orthogonality* is equivalent to *uncorrelation*.
- x is *wide-sense stationary* (WSS) if
  1. $\mathtt{m}_x(t) = \mathtt{m}_x$, $\forall t$,
  2. $\mathtt{r}_x(t, t-\tau) = \mathtt{r}_x(\tau)$, $\forall t$.

[15] We observe that the notion of *orthogonality between two random processes* is quite different from that of *orthogonality between two deterministic signals*. In fact, while in the deterministic case, based on Definition 1.2, it is sufficient that the inner product of the signals is zero, in the random case the cross-correlation must be zero for all the delays and not only for zero delay.
In particular, we note that the *two random variables* v_1 and v_2 are orthogonal if condition $E[v_1 v_2^*] = 0$ is satisfied.

- $\mathtt{r}_x(0) = E[|x(t)|^2] = \mathtt{M}_x$ is the *statistical power*, whereas $\mathtt{c}_x(0) = \sigma_x^2 = \mathtt{M}_x - |\mathtt{m}_x|^2$ is the *variance* of $x(t)$.
- $x(t)$ and $y(t)$ are *jointly wide-sense stationary* if
  1. $\mathtt{m}_x(t) = \mathtt{m}_x$, $\mathtt{m}_y(t) = \mathtt{m}_y$, $\forall t$,
  2. $\mathtt{r}_{xy}(t, t-\tau) = \mathtt{r}_{xy}(\tau)$, $\forall t$.

Properties of the autocorrelation function

1. $\mathtt{r}_x(-\tau) = \mathtt{r}_x^*(\tau)$, $\mathtt{r}_x(\tau)$ is a function with Hermitian symmetry.
2. $\mathtt{r}_x(0) \geq |\mathtt{r}_x(\tau)|$.
3. $\mathtt{r}_x(0)\mathtt{r}_y(0) \geq |\mathtt{r}_{xy}(\tau)|^2$.
4. $\mathtt{r}_{xy}(-\tau) = \mathtt{r}_{yx}^*(\tau)$.
5. $\mathtt{r}_{x^*}(\tau) = \mathtt{r}_x^*(\tau)$.

1.7.2 Power spectral density

Given the WSS random process $x(t)$, $t \in \Re$, its *power spectral density* (PSD) is defined as the Fourier transform of the autocorrelation function

$$\mathcal{P}_x(f) = \mathcal{F}[\mathtt{r}_x(\tau)] = \int_{-\infty}^{+\infty} \mathtt{r}_x(\tau) e^{-j2\pi f\tau} d\tau \tag{1.216}$$

The inverse transformation is given by the following formula:

$$\mathtt{r}_x(\tau) = \int_{-\infty}^{+\infty} \mathcal{P}_x(f) e^{j2\pi f\tau}\, df \tag{1.217}$$

In particular from (1.217) one gets the statistical power

$$\mathtt{M}_x = \mathtt{r}_x(0) = \int_{-\infty}^{+\infty} \mathcal{P}_x(f)\, df \tag{1.218}$$

hence the name PSD for the function $\mathcal{P}_x(f)$: it represents the distribution of the statistical power in the frequency domain.

The pair of equations (1.216) and (1.217) are obtained from the *Wiener–Khintchine theorem* [2].

Definition 1.11
The passband $\mathcal{B}$ of a random process x is defined with reference to its PSD function.

Spectral lines in the PSD

In many applications it is important to detect the presence of sinusoidal components in a random process. With this aim in mind we give the following theorem.

Theorem 1.2
The PSD of a WSS process, $\mathcal{P}_x$, can be uniquely decomposed into a component $\mathcal{P}_x^{(c)}$ with no impulses and a discrete component consisting of impulses (spectral lines) $\mathcal{P}_x^{(d)}$, so that

$$\mathcal{P}_x(f) = \mathcal{P}_x^{(c)}(f) + \mathcal{P}_x^{(d)}(f) \tag{1.219}$$

where $\mathcal{P}_x^{(c)}$ is an ordinary (piecewise linear) function and

$$\mathcal{P}_x^{(d)}(f) = \sum_{i \in \mathcal{I}} \mathtt{M}_i \delta(f - f_i) \tag{1.220}$$

where $\mathcal{I}$ identifies a discrete set of frequencies $\{f_i\}$, $i \in \mathcal{I}$.

The inverse Fourier transform of (1.219) yields the relation

$$\mathtt{r}_x(\tau) = \mathtt{r}_x^{(c)}(\tau) + \mathtt{r}_x^{(d)}(\tau) \tag{1.221}$$

with

$$\mathtt{r}_x^{(d)}(\tau) = \sum_{i \in \mathcal{I}} \mathtt{M}_i e^{j2\pi f_i \tau} \tag{1.222}$$

The most interesting consideration is that the following random process decomposition corresponds to the decomposition (1.219) of the PSD:

$$x(t) = x^{(c)}(t) + x^{(d)}(t) \tag{1.223}$$

where $x^{(c)}$ and $x^{(d)}$ are *orthogonal* processes having PSD functions

$$\mathcal{P}_{x^{(c)}}(f) = \mathcal{P}_x^{(c)}(f) \quad \text{and} \quad \mathcal{P}_{x^{(d)}}(f) = \mathcal{P}_x^{(d)}(f) \tag{1.224}$$

Moreover, $x^{(d)}$ is given by

$$x^{(d)}(t) = \sum_{i \in \mathcal{I}} x_i e^{j2\pi f_i t} \tag{1.225}$$

where $\{x_i\}$ are orthogonal random variables (r.v.s) having statistical power

$$E[|x_i|^2] = \mathtt{M}_i \qquad i \in \mathcal{I} \tag{1.226}$$

where $\mathtt{M}_i$ is defined in (1.220).

Observation 1.5
The spectral lines of the PSD identify the periodic components in the process.

Definition 1.12
A WSS random process is said to be *asymptotically uncorrelated* if the following two properties hold:

$$1) \quad \lim_{\tau \to \infty} \mathtt{r}_x(\tau) = |\mathtt{m}_x|^2 \tag{1.227}$$

$$2) \quad \mathtt{c}_x(\tau) = \mathtt{r}_x(\tau) - |\mathtt{m}_x|^2 \text{ is absolutely integrable .} \tag{1.228}$$

The property 1) denotes that $x(t)$ and $x(t-\tau)$ become uncorrelated for $\tau \to \infty$.

For such processes, one can prove that

$$\mathtt{r}_x^{(c)}(\tau) = \mathtt{c}_x(\tau) \quad \text{and} \quad \mathtt{r}_x^{(d)}(\tau) = |\mathtt{m}_x|^2 \tag{1.229}$$

Hence $\mathcal{P}_x^{(d)}(f) = |\mathtt{m}_x|^2 \delta(f)$, and the process exhibits at most a spectral line at the origin.

Cross-power spectral density

One can extend the definition of PSD to two jointly WSS random processes:

$$\mathcal{P}_{xy}(f) = \mathcal{F}[\mathtt{r}_{xy}(k)] \tag{1.230}$$

Since $\mathtt{r}_{xy}(-\tau) \neq \mathtt{r}_{xy}^*(\tau)$, $\mathcal{P}_{xy}(f)$ is in general a complex function.

Properties of the PSD

1. $\mathcal{P}_x(f)$ is a *real-valued function*. This follows from property 1 of the autocorrelation.
2. $\mathcal{P}_x(f)$ is generally *not an even function*. However, if the process $x(t)$ is real valued, then both $\mathtt{r}_x(\tau)$ and $\mathcal{P}_x(f)$ are even functions.
3. $\mathcal{P}_x(f)$ is a *non-negative function*.
4. $\mathcal{P}_{yx}(f) = \mathcal{P}_{xy}^*(f)$.
5. $\mathcal{P}_{x^*}(f) = \mathcal{P}_x(-f)$.

Moreover, the following inequality holds:

$$0 \leq |\mathcal{P}_{xy}(f)|^2 \leq \mathcal{P}_x(f)\mathcal{P}_y(f) \tag{1.231}$$

Definition 1.13 (White random process)
The zero-mean random process $x(t)$, $t \in \Re$, is called *white* if

$$\mathtt{r}_x(\tau) = \sigma_x^2 \delta(\tau) \tag{1.232}$$

In this case

$$\mathcal{P}_x(f) = \sigma_x^2 \tag{1.233}$$

i.e. $\mathcal{P}_x$ is a constant.

PSD of processes through linear transformations

By an example we will show how to determine PSDs of processes in a linear system, assuming the PSDs of the various input processes are known. We consider the scheme of Figure 1.31 in which the inputs x_1 and x_2 have the following PSDs: $\mathcal{P}_{x_1}(f)$, $\mathcal{P}_{x_2}(f)$, and $\mathcal{P}_{x_1x_2}(f)$. To determine the PSDs $\mathcal{P}_{y_1}(f)$, $\mathcal{P}_{y_2}(f)$, and $\mathcal{P}_{y_1y_2}(f)$, the procedure consists of three steps.

1. Determine the frequency response of the various outputs in terms of the inputs. In our specific case, we have

$$\mathcal{Y}_1 = \mathcal{H}_1\mathcal{X}_1 + \mathcal{H}_2\mathcal{X}_2 \tag{1.234}$$

$$\mathcal{Y}_2 = \mathcal{H}_2\mathcal{H}_3\mathcal{X}_2 \tag{1.235}$$

   in which for simplicity we omit the argument f.

2. Construct the different products

$$\mathcal{Y}_1\mathcal{Y}_1^* = |\mathcal{H}_1|^2\mathcal{X}_1\mathcal{X}_1^* + |\mathcal{H}_2|^2\mathcal{X}_2\mathcal{X}_2^* + \mathcal{H}_1\mathcal{H}_2^*\mathcal{X}_1\mathcal{X}_2^* + \mathcal{H}_1^*\mathcal{H}_2\mathcal{X}_1^*\mathcal{X}_2 \tag{1.236}$$

$$\mathcal{Y}_2\mathcal{Y}_2^* = |\mathcal{H}_2\mathcal{H}_3|^2\mathcal{X}_2\mathcal{X}_2^* \tag{1.237}$$

$$\mathcal{Y}_1\mathcal{Y}_2^* = \mathcal{H}_1\mathcal{H}_2^*\mathcal{H}_3^*\mathcal{X}_1\mathcal{X}_2^* + |\mathcal{H}_2|^2\mathcal{H}_3^*\mathcal{X}_2\mathcal{X}_2^* \tag{1.238}$$

3. Substitute the expressions of the products in the previous equations using the rule

$$\mathcal{Y}_i\mathcal{Y}_j^* \longrightarrow \mathcal{P}_{y_iy_j} \tag{1.239}$$

$$\mathcal{X}_\ell\mathcal{X}_m^* \longrightarrow \mathcal{P}_{x_\ell x_m} \tag{1.240}$$

   Then

$$\mathcal{P}_{y_1} = |\mathcal{H}_1|^2\mathcal{P}_{x_1} + |\mathcal{H}_2|^2\mathcal{P}_{x_2} + \mathcal{H}_1\mathcal{H}_2^*\mathcal{P}_{x_1x_2} + \mathcal{H}_1^*\mathcal{H}_2\mathcal{P}_{x_1x_2}^* \tag{1.241}$$

$$\mathcal{P}_{y_2} = |\mathcal{H}_2\mathcal{H}_3|^2\mathcal{P}_{x_2} \tag{1.242}$$

$$\mathcal{P}_{y_1y_2} = \mathcal{H}_1\mathcal{H}_2^*\mathcal{H}_3^*\mathcal{P}_{x_1x_2} + |\mathcal{H}_2|^2\mathcal{H}_3^*\mathcal{P}_{x_2} \tag{1.243}$$

The proof of the above method is based on the relation (1.449).

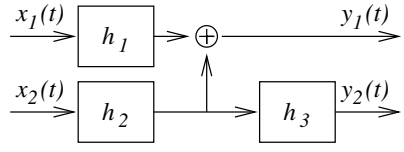


Figure 1.31. PSD of processes through filtering.

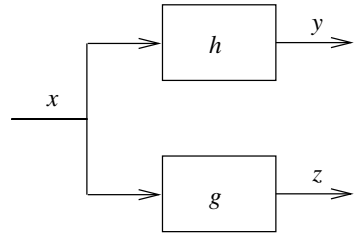


Figure 1.32. Reference scheme of PSD computations.

PSD of processes through filtering

With reference to Figure 1.32, by applying the above method the following relations are easily obtained:

$$\mathcal{P}_{yx}(f) = \mathcal{P}_x(f)\mathcal{H}(f) \tag{1.244}$$

$$\mathcal{P}_y(f) = \mathcal{P}_x(f)|\mathcal{H}(f)|^2 \tag{1.245}$$

$$\mathcal{P}_{yz}(f) = \mathcal{P}_x(f)\mathcal{H}(f)\mathcal{G}^*(f) \tag{1.246}$$

The relation (1.245) is of particular interest since it relates the spectral density of the output process of a filter to the spectral density of the input process, through the frequency response of the filter. In the particular case in which y and z have *disjoint passbands*, i.e. $\mathcal{P}_y(f)\mathcal{P}_z(f) = 0$, then, from (1.231), $\mathtt{r}_{yz}(\tau) = 0$, and $y \perp z$.

1.7.3 PSD of discrete-time random processes

Let $\{x(k)\}$ and $\{y(k)\}$ be two discrete-time random processes. Definitions and properties of Section 1.7.1 remain valid also for discrete-time processes: the only difference is that the correlation is now defined on discrete time and is called *autocorrelation sequence* (ACS). It is, however, interesting to review the properties of PSDs. Given a discrete-time WSS random process x, the PSD is obtained as

$$\mathcal{P}_x(f) = T_c\mathcal{F}[\mathtt{r}_x(n)] = T_c \sum_{n=-\infty}^{+\infty} \mathtt{r}_x(n)e^{-j2\pi fnT_c} \tag{1.247}$$

We note a further property: $\mathcal{P}_x(f)$ is a periodic function of period $1/T_c$. The inverse transformation yields:

$$\mathtt{r}_x(n) = \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \mathcal{P}_x(f)e^{j2\pi fnT_c}\,df \tag{1.248}$$

In particular, the statistical power is given by

$$\mathtt{M}_x = \mathtt{r}_x(0) = \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \mathcal{P}_x(f)\,df \tag{1.249}$$

Definition 1.14 (White random process)
A discrete-time random process $\{x(k)\}$ is *white* if

$$\mathtt{r}_x(n) = \sigma_x^2 \delta_n \tag{1.250}$$

In this case the PSD is a constant:

$$\mathcal{P}_x(f) = \sigma_x^2 T_c \tag{1.251}$$

Definition 1.15
If the samples of the random process $\{x(k)\}$ are statistically independent and identically distributed we say that $\{x(k)\}$ has i.i.d. samples.

Spectral lines in the PSD

Even for a discrete time random process the PSD can be decomposed into ordinary components and spectral lines, provided the decomposition (1.219) is limited to a period of the PSD.

In particular for a discrete-time WSS *asymptotically uncorrelated* random process, the relation (1.229) and the following are true

$$\mathcal{P}_x^{(c)}(f) = T_c \sum_{n=-\infty}^{+\infty} \mathtt{c}_x(n)\, e^{-j2\pi f n T_c} \tag{1.252}$$

$$\mathcal{P}_x^{(d)}(f) = |\mathtt{m}_x|^2 \sum_{\ell=-\infty}^{+\infty} \delta\left(f - \frac{\ell}{T_c}\right) \tag{1.253}$$

We note that, if the process has non-zero mean value, the PSD exhibits lines at multiples of $1/T_c$.

Example 1.7.1
We calculate the PSD of an i.i.d. sequence $\{x(k)\}$. From

$$\mathtt{r}_x(n) = \begin{cases} \mathtt{M}_x & n = 0 \\ |\mathtt{m}_x|^2 & n \neq 0 \end{cases} \tag{1.254}$$

it follows that

$$\mathtt{c}_x(n) = \begin{cases} \sigma_x^2 & n = 0 \\ 0 & n \neq 0 \end{cases} \tag{1.255}$$

Then

$$\mathtt{r}_x^{(c)}(n) = \sigma_x^2 \delta_n \qquad \mathtt{r}_x^{(d)}(n) = |\mathtt{m}_x|^2 \tag{1.256}$$

$$\mathcal{P}_x^{(c)}(f) = \sigma_x^2 T_c \qquad \mathcal{P}_x^{(d)}(f) = |\mathtt{m}_x|^2 \sum_{\ell=-\infty}^{+\infty} \delta\left(f - \frac{\ell}{T_c}\right) \tag{1.257}$$

PSD of processes through filtering

Given the system illustrated in Figure 1.12, we want to find a relation between the PSDs of the input and output signals, assuming these processes are individually as well as jointly WSS. We introduce the z-transform of the correlation sequence:

$$P_x(z) = \sum_{n=-\infty}^{+\infty} \mathtt{r}_x(n) z^{-n} \tag{1.258}$$

From the comparison of (1.258) with (1.247), the PSD of $x(k)$ is related to $P_x(z)$ by

$$\mathcal{P}_x(f) = T_c P_x(e^{j2\pi f T_c}) \tag{1.259}$$

Using Table 1.3, we obtain the relations between ACS and PSD listed in Table 1.6. Let the deterministic autocorrelation of h be defined as [16]

$$\mathtt{r}_h(n) = \sum_{k=-\infty}^{+\infty} h(k) h^*(k-n) = [h(m) * h^*(-m)](n) \tag{1.260}$$

whose z-transform is given by

$$P_h(z) = \sum_{n=-\infty}^{+\infty} \mathtt{r}_h(n)\, z^{-n} = H(z)\, H^*\left(\frac{1}{z^*}\right) \tag{1.261}$$

In case $P_h(z)$ is a rational function, from (1.261) one deduces that, if $P_h(z)$ has a pole (zero) of the type $e^{j\varphi}|a|$, it also has a corresponding pole (zero) of the type $e^{j\varphi}/|a|$. Consequently the poles (and zeros) of $P_h(z)$ come in pairs of the type $e^{j\varphi}|a|$, $e^{j\varphi}/|a|$.

From the last relation in Table 1.6 one gets the relation between the PSDs of input and output signals, given by

$$\mathcal{P}_y(f) = \mathcal{P}_x(f) |H(e^{j2\pi f T_c})|^2 \tag{1.262}$$

In the case of white noise input, then

$$P_y(z) = \sigma_x^2 H(z) H^*\left(\frac{1}{z^*}\right) \tag{1.263}$$

Table 1.6 Relations between ACS and PSD for discrete-time processes through a linear filter.

ACS	PSD
$\mathtt{r}_{yx}(n) = \mathtt{r}_x * h(n)$	$P_{yx}(z) = P_x(z) H(z)$
$\mathtt{r}_{xy}(n) = [\mathtt{r}_x(m) * h^*(-m)](n)$	$P_{xy}(z) = P_x(z) H^*(1/z^*)$
$\mathtt{r}_y(n) = \mathtt{r}_{xy} * h(n)$	$P_y(z) = P_{xy}(z) H(z)$
$= \mathtt{r}_x * \mathtt{r}_h(n)$	$= P_x(z) H(z) H^*(1/z^*)$

[16] In this text we use the same symbol to indicate the correlation between random processes and the correlation between deterministic functions.

and

$$\mathcal{P}_y(f) = T_c \sigma_x^2 |H(e^{j2\pi f T_c})|^2 \tag{1.264}$$

In other words, $\mathcal{P}_y(f)$ has the same shape as the filter frequency response.

In the case of real filters

$$H^*\left(\frac{1}{z^*}\right) = H(z^{-1}) \tag{1.265}$$

Among the various applications of (1.264), it is worth mentioning the *process synthesis*, which deals with the generation of a random process having a pre-assigned PSD. Two methods are shown in Section 4.6.6.

Minimum-phase spectral factorization

In the previous section we introduced the relation between an impulse response $\{h(k)\}$ and its autocorrelation sequence $\{\mathrm{r}_h(n)\}$ in terms of the z-transform. In many practical applications it is interesting to determine the minimum-phase impulse response for a given autocorrelation function: with this intent we state the following theorem [3].

Theorem 1.3 (Spectral factorization for discrete-time processes)
Let the process y be given, with autocorrelation sequence $\{\mathrm{r}_y(n)\}$ having z-transform $P_y(z)$, which satisfies the Paley–Wiener condition for discrete-time systems, i.e.

$$\int_{1/T_c} |\log P_y(e^{j2\pi f T_c})|\, df < \infty \tag{1.266}$$

where the integration is over an arbitrarily chosen interval $1/T_c$. Then the function $P_y(z)$ can be factorized as follows:

$$P_y(z) = f_0^2\, \tilde{F}(z)\, \tilde{F}^*\left(\frac{1}{z^*}\right) \tag{1.267}$$

where

$$\tilde{F}(z) = 1 + \tilde{f}_1 z^{-1} + \tilde{f}_2 z^{-2} + \cdots \tag{1.268}$$

is monic, minimum phase, and associated with a causal sequence $\{1, \tilde{f}_1, \tilde{f}_2, \ldots\}$. The factor f_0 in (1.267) is the geometric mean of $P_y(e^{j2\pi f T_c})$:

$$\log f_0^2 = T_c \int_{1/T_c} \log P_y(e^{j2\pi f T_c})\, df \tag{1.269}$$

The logarithms in (1.266) and (1.269) may have any common base.

The Paley–Wiener criterion implies that $P_y(z)$ may have only a discrete set of zeros on the unit circle, and that the spectral factorization (1.267) (with the constraint that $\tilde{F}(z)$ is causal, monic and minimum phase) is unique. For rational $P_y(z)$, the function $f_0\tilde{F}(z)$ is obtained by extracting the poles and zeros of $P_y(z)$ that lie inside the unit circle (see (1.526) and the considerations relative to (1.261)). Moreover, in (1.267) $f_0\tilde{F}^*(1/z^*)$ is the z-transform of an anti-causal sequence $f_0\{\ldots, \tilde{f}_2^*, \tilde{f}_1^*, 1\}$, associated with poles and zeros of $P_y(z)$ that lie outside the unit circle.

1.7.4 PSD of passband processes

Definition 1.16
A WSS random process x is said to be PB (BB) if its PSD is of PB (BB) type.

PSD of the quadrature components of a random process

Let x be a real PB, WSS process. Our aim is to derive the power spectral density of the in-phase and quadrature components of the process. We assume that x does not have DC components, i.e. frequency components at $f = 0$, hence its mean is zero and consequently also $x^{(a)}$ and $x^{(bb)}$ have zero mean.

We introduce two filters having a non-overlapping passband and ideal frequency response given by

$$\mathcal{H}^{(+)}(f) = 1(f) \quad \text{and} \quad \mathcal{H}^{(-)}(f) = 1(-f) \tag{1.270}$$

For the same input x, the output of the two filters is respectively $x^{(+)}$ and $x^{(-)}$. We find that

$$x(t) = x^{(+)}(t) + x^{(-)}(t) \tag{1.271}$$

with $x^{(-)}(t) = x^{(+)*}(t)$. The following relations are valid

$$\mathcal{P}_{x^{(+)}}(f) = |\mathcal{H}^{(+)}(f)|^2 \mathcal{P}_x(f) = \mathcal{P}_x(f)1(f) \tag{1.272}$$

$$\mathcal{P}_{x^{(-)}}(f) = |\mathcal{H}^{(-)}(f)|^2 \mathcal{P}_x(f) = \mathcal{P}_x(f)1(-f) \tag{1.273}$$

and

$$\mathcal{P}_{x^{(+)}x^{(-)}}(f) = 0 \tag{1.274}$$

as $x^{(+)}$ and $x^{(-)}$ have non-overlapping passbands. Then $x^{(+)} \perp x^{(-)}$, hence (1.271) yields

$$\mathcal{P}_x(f) = \mathcal{P}_{x^{(+)}}(f) + \mathcal{P}_{x^{(-)}}(f) \tag{1.275}$$

where $\mathcal{P}_{x^{(-)}}(f) = \mathcal{P}_{x^{(+)*}}(f) = \mathcal{P}_{x^{(+)}}(-f)$, using Property 5 of the PSD. The analytic signal $x^{(a)}$ is equal to $2x^{(+)}$, hence

$$\mathrm{r}_{x^{(a)}}(\tau) = 4\mathrm{r}_{x^{(+)}}(\tau) \tag{1.276}$$

and

$$\mathcal{P}_{x^{(a)}}(f) = 4\mathcal{P}_{x^{(+)}}(f) \tag{1.277}$$

Moreover, being $x^{(a)*} = 2x^{(-)}$, it follows that $x^{(a)} \perp x^{(a)*}$ and

$$\mathrm{r}_{x^{(a)}x^{(a)*}}(\tau) = 0 \tag{1.278}$$

The complex envelope $x^{(bb)}$ is related to $x^{(a)}$ by (1.152) and

$$\mathrm{r}_{x^{(bb)}}(\tau) = \mathrm{r}_{x^{(a)}}(\tau)e^{-j2\pi f_0 \tau} \tag{1.279}$$

hence

$$\mathcal{P}_{x^{(bb)}}(f) = \mathcal{P}_{x^{(a)}}(f + f_0) = 4\mathcal{P}_{x^{(+)}}(f + f_0) \tag{1.280}$$

Moreover, from (1.278) it follows that $x^{(bb)} \perp x^{(bb)*}$.

Using (1.280), (1.275) can be written as

$$\mathcal{P}_x(f) = \tfrac{1}{4}[\mathcal{P}_{x^{(bb)}}(f - f_0) + \mathcal{P}_{x^{(bb)}}(-f - f_0)] \tag{1.281}$$

Finally, from

$$x_I^{(bb)}(t) = Re[x^{(bb)}(t)] = \frac{x^{(bb)}(t) + x^{(bb)*}(t)}{2} \tag{1.282}$$

and

$$x_Q^{(bb)}(t) = Im[x^{(bb)}(t)] = \frac{x^{(bb)}(t) - x^{(bb)*}(t)}{2j} \tag{1.283}$$

we obtain the following relations:

$$\mathtt{r}_{x_I^{(bb)}}(\tau) = \tfrac{1}{2}Re[\mathtt{r}_{x^{(bb)}}(\tau)] \tag{1.284}$$

$$\mathcal{P}_{x_I^{(bb)}}(f) = \tfrac{1}{4}[\mathcal{P}_{x^{(bb)}}(f) + \mathcal{P}_{x^{(bb)}}(-f)] \tag{1.285}$$

$$\mathtt{r}_{x_Q^{(bb)}}(\tau) = \mathtt{r}_{x_I^{(bb)}}(\tau) \tag{1.286}$$

$$\mathtt{r}_{x_Q^{(bb)}x_I^{(bb)}}(\tau) = \tfrac{1}{2}Im[\mathtt{r}_{x^{(bb)}}(\tau)] \tag{1.287}$$

$$\mathcal{P}_{x_Q^{(bb)}x_I^{(bb)}}(f) = \frac{1}{4j}[\mathcal{P}_{x^{(bb)}}(f) - \mathcal{P}_{x^{(bb)}}(-f)] \tag{1.288}$$

$$\mathtt{r}_{x_I^{(bb)}x_Q^{(bb)}}(\tau) = -\mathtt{r}_{x_Q^{(bb)}x_I^{(bb)}}(\tau) = -\mathtt{r}_{x_I^{(bb)}x_Q^{(bb)}}(-\tau) \tag{1.289}$$

The second equality in (1.289) follows from Property 4 of ACS.

From (1.289) we note that $\mathtt{r}_{x_I^{(bb)}x_Q^{(bb)}}(\tau)$ is an odd function. Moreover, from (1.288) one gets $x_I^{(bb)} \perp x_Q^{(bb)}$ only if $\mathcal{P}_{x^{(bb)}}$ is an even function; in any case the random variables $x_I^{(bb)}(t)$ and $x_Q^{(bb)}(t)$ are always orthogonal since $\mathtt{r}_{x_I^{(bb)}x_Q^{(bb)}}(0) = 0$. Referring to the block diagram in Figure 1.27b, as

$$\mathcal{P}_{x^{(h)}}(f) = \mathcal{P}_x(f) \quad \text{and} \quad \mathcal{P}_{x^{(h)}x}(f) = -j\,\mathrm{sgn}(f)\,\mathcal{P}_x(f) \tag{1.290}$$

one gets

$$\mathtt{r}_{x^{(h)}}(\tau) = \mathtt{r}_x(\tau) \quad \text{and} \quad \mathtt{r}_{x^{(h)}x}(\tau) = \mathtt{r}_x^{(h)}(\tau) \tag{1.291}$$

Then

$$\mathtt{r}_{x_I^{(bb)}}(\tau) = \mathtt{r}_{x_Q^{(bb)}}(\tau) = \mathtt{r}_x(\tau)\cos 2\pi f_0\tau + \mathtt{r}_x^{(h)}(\tau)\sin 2\pi f_0\tau \tag{1.292}$$

and

$$\mathtt{r}_{x_I^{(bb)} x_Q^{(bb)}}(\tau) = -\mathtt{r}_x^{(h)}(\tau)\cos 2\pi f_0\tau + \mathtt{r}_x(\tau)\sin 2\pi f_0\tau \tag{1.293}$$

In terms of statistical power the following relations hold:

$$\mathtt{r}_{x^{(+)}}(0) = \mathtt{r}_{x^{(-)}}(0) = \tfrac{1}{2}\mathtt{r}_x(0) \tag{1.294}$$

$$\mathtt{r}_{x^{(bb)}}(0) = \mathtt{r}_{x^{(a)}}(0) = 4\mathtt{r}_{x^{(+)}}(0) = 2\mathtt{r}_x(0) \tag{1.295}$$

$$\mathtt{r}_{x_I^{(bb)}}(0) = \mathtt{r}_{x_Q^{(bb)}}(0) = \mathtt{r}_x(0) \tag{1.296}$$

$$\mathtt{r}_{x^{(h)}}(0) = \mathtt{r}_x(0) \tag{1.297}$$

Example 1.7.2
Let x be a WSS process with power spectral density

$$\mathcal{P}_x(f) = \frac{N_0}{2}\left[\text{rect}\left(\frac{f-f_0}{B}\right) + \text{rect}\left(\frac{f+f_0}{B}\right)\right] \tag{1.298}$$

depicted in Figure 1.33. It is immediate to get

$$\mathcal{P}_{x^{(a)}}(f) = 2N_0\,\text{rect}\left(\frac{f-f_0}{B}\right) \tag{1.299}$$

and

$$\mathcal{P}_{x^{(bb)}}(f) = 2N_0\,\text{rect}\left(\frac{f}{B}\right) \tag{1.300}$$

Then

$$\mathcal{P}_{x_I^{(bb)}}(f) = \mathcal{P}_{x_Q^{(bb)}}(f) = \frac{1}{2}\mathcal{P}_{x^{(bb)}}(f) = N_0\,\text{rect}\left(\frac{f}{B}\right) \tag{1.301}$$

Moreover, being $\mathcal{P}_{x_I^{(bb)} x_Q^{(bb)}}(f) = 0$, we find that $x_I^{(bb)} \perp x_Q^{(bb)}$.

Cyclostationary processes

In short, we have seen that, if x is a real passband WSS process, then its complex envelope is WSS, and $x^{(bb)} \perp x^{(bb)*}$. The converse is also true: if $x^{(bb)}$ is a WSS process and $x^{(bb)} \perp x^{(bb)*}$, then

$$x(t) = Re[x^{(bb)}(t)\, e^{j2\pi f_0 t}] \tag{1.302}$$

is WSS with PSD given by (1.281). If $x^{(bb)}$ is WSS, however, with

$$\mathtt{r}_{x^{(bb)}x^{(bb)*}}(\tau) \neq 0 \tag{1.303}$$

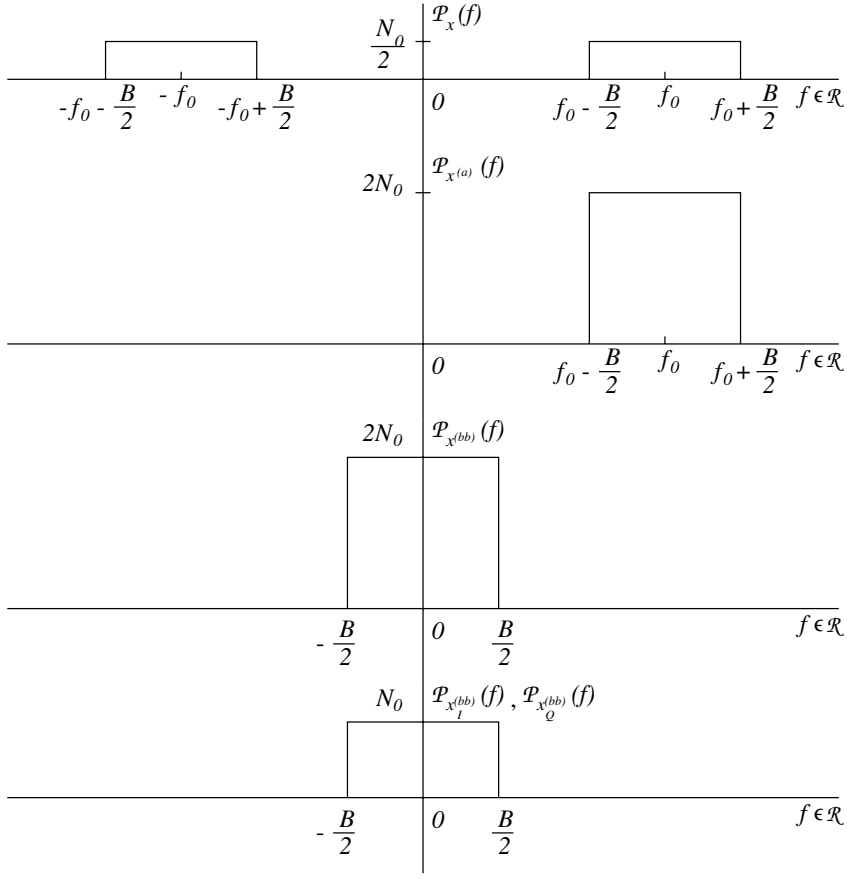


Figure 1.33. Spectral representation of a PB process and its BB components.

observing (1.302) we find that the autocorrelation of $x(t)$ is a periodic function in t of period $1/f_0$:

$$\begin{aligned} \mathrm{r}_x(t, t-\tau) = \tfrac{1}{4}[&\mathrm{r}_{x^{(bb)}}(\tau)e^{j2\pi f_0\tau} + \mathrm{r}^*_{x^{(bb)}}(\tau)e^{-j2\pi f_0\tau} \\ &+ \mathrm{r}_{x^{(bb)}x^{(bb)*}}(\tau)e^{-j2\pi f_0\tau}e^{j4\pi f_0 t} + \mathrm{r}^*_{x^{(bb)}x^{(bb)*}}(\tau)e^{j2\pi f_0\tau}e^{-j4\pi f_0 t}] \end{aligned} \tag{1.304}$$

In other words, x is a *cyclostationary* process of period $T_0 = 1/f_0$.[17]

In this case it is convenient to introduce the *average correlation*

$$\bar{\mathrm{r}}_x(\tau) = \frac{1}{T_0}\int_0^{T_0} \mathrm{r}_x(t, t-\tau)\, dt \tag{1.305}$$

[17] To be precise, x is cyclostationary in mean value with period $T_0 = 1/f_0$, while it is cyclostationary in correlation with period $T_0/2$.

whose Fourier transform is the *average power spectral density*

$$\bar{\mathcal{P}}_x(f) = \mathcal{F}[\bar{\mathtt{r}}_x(\tau)] = \frac{1}{T_0}\int_0^{T_0} \mathcal{P}_x(f,t)\,dt \tag{1.306}$$

where

$$\mathcal{P}_x(f,t) = \mathcal{F}_\tau[\mathtt{r}_x(t,t-\tau)] \tag{1.307}$$

In (1.307) $\mathcal{F}_\tau$ denotes the Fourier transform with respect to the variable τ. In our case, it is

$$\bar{\mathcal{P}}_x(f) = \tfrac{1}{4}[\mathcal{P}_{x^{(bb)}}(f-f_0) + \mathcal{P}_{x^{(bb)}}(-f-f_0)] \tag{1.308}$$

as in the stationary case (1.281).

Example 1.7.3
Let $x(t)$ be a modulated DSB signal (see (1.189))

$$x(t) = a(t)\cos(2\pi f_0 t + \varphi_0) \tag{1.309}$$

with $a(t)$ real random BB WSS process with bandwidth $B_a < f_0$ and autocorrelation $\mathtt{r}_a(\tau)$. From (1.192) it results in $x^{(bb)}(t) = a(t)\,e^{j\varphi_0}$. Hence we have

$$\mathtt{r}_{x^{(bb)}}(\tau) = \mathtt{r}_a(\tau) \qquad \mathtt{r}_{x^{(bb)}x^{(bb)*}}(\tau) = \mathtt{r}_a(\tau)\,e^{j2\varphi_0} \tag{1.310}$$

Because $\mathtt{r}_a(\tau)$ is not identical to zero, observing (1.303) we find that $x(t)$ is cyclostationary with period $1/f_0$. From (1.308) the average PSD of x is given by

$$\bar{\mathcal{P}}_x(f) = \tfrac{1}{4}[\mathcal{P}_a(f-f_0) + \mathcal{P}_a(f+f_0)] \tag{1.311}$$

Therefore x has a bandwidth equal to $2B_a$ and an average statistical power

$$\bar{\mathtt{M}}_x = \tfrac{1}{2}\,\mathtt{M}_a \tag{1.312}$$

We note that one finds the same result (1.311) assuming that φ_0 is a uniform r.v. in $[0, 2\pi)$; in this case x turns out to be WSS.

Example 1.7.4
Let $x(t)$ be a modulated *single sideband* (SSB) with an upper sideband,

$$\begin{aligned} x(t) &= Re[\tfrac{1}{2}(a(t) + j a^{(h)}(t))e^{j(2\pi f_0 t+\varphi_0)}] \\ &= \tfrac{1}{2}\,a(t)\cos(2\pi f_0 t + \varphi_0) - \tfrac{1}{2}\,a^{(h)}(t)\sin(2\pi f_0 t + \varphi_0) \end{aligned} \tag{1.313}$$

where $a^{(h)}(t)$ is the Hilbert transform of $a(t)$, a real WSS random process with autocorrelation $\mathtt{r}_a(\tau)$ and bandwidth B_a.

We note that the modulating signal $(a(t) + ja^{(h)}(t))$ coincides with the analytic signal $a^{(a)}(t)$ and it has spectral support only for positive frequencies, therefore it is one half of $a(t)$.

Being

$$x^{(bb)}(t) = \tfrac{1}{2}(a(t) + ja^{(h)}(t))e^{j\varphi_0}$$

it results that $x^{(bb)}$ and $x^{(bb)*}$ have non-overlapping passbands and

$$\mathtt{r}_{x^{(bb)}x^{(bb)*}}(\tau) = 0 \tag{1.314}$$

The process (1.313) is then stationary with

$$\mathcal{P}_x(f) = \tfrac{1}{4}[\mathcal{P}_{a^{(+)}}(f - f_0) + \mathcal{P}_{a^{(+)}}(-f - f_0)] \tag{1.315}$$

where $a^{(+)}(t)$ is defined in (1.271). In this case x has bandwidth equal to B_a and statistical power given by

$$\mathtt{M}_x = \tfrac{1}{4}\,\mathtt{M}_a \tag{1.316}$$

Example 1.7.5 (DSB and SSB demodulators)
Let the signal $r(t)$ be the sum of a desired part $x(t)$ and additive white noise $w(t)$ with PSD equal to $\mathcal{P}_w(f) = N_0/2$,

$$r(t) = x(t) + w(t) \tag{1.317}$$

where the signal x is modulated DSB (1.309). To obtain the signal $a(t)$ from $r(t)$, one can use the coherent demodulation scheme illustrated in Figure 1.34 (see Figure 1.30b) where h is an ideal lowpass filter, having a frequency response

$$\mathcal{H}(f) = \mathcal{H}_0 \text{ rect}\left(\frac{f}{2B_a}\right) \tag{1.318}$$

Let r_o be the output signal of the demodulator, given by the sum of the desired part x_o and noise w_o:

$$r_o(t) = x_o(t) + w_o(t) \tag{1.319}$$

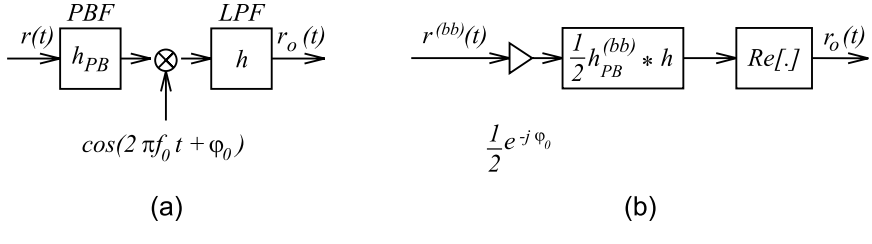


Figure 1.34. Coherent DSB demodulator and baseband-equivalent scheme.

We evaluate now the ratio between the powers of the signals in (1.319),

$$\Lambda_o = \frac{\mathtt{M}_{x_o}}{\mathtt{M}_{w_o}} \tag{1.320}$$

in terms of the reference ratio

$$\Gamma = \frac{\mathtt{M}_x}{(N_0/2)\,2B_a} \tag{1.321}$$

Using the equivalent block scheme of Figure 1.34 and (1.192), we have

$$r^{(bb)}(t) = a(t)\,e^{j\varphi_0} + w^{(bb)}(t) \tag{1.322}$$

with $\mathcal{P}_{w^{(bb)}}(f) = 2N_0\,1(f + f_0)$. Being

$$h * a(t) = \mathcal{H}_0\,a(t) \tag{1.323}$$

it results

$$\begin{aligned} x_o(t) &= Re[h(\eta) * \tfrac{1}{2}\,e^{-j\varphi_1}\,a(\eta)\,e^{j\varphi_0}](t) \\ &= \frac{\mathcal{H}_0}{2}\,a(t)\cos(\varphi_0 - \varphi_1) \end{aligned} \tag{1.324}$$

Hence we get

$$\mathtt{M}_{x_o} = \frac{\mathcal{H}_0^2}{4}\,\mathtt{M}_a \cos^2(\varphi_0 - \varphi_1) \tag{1.325}$$

In the same baseband equivalent scheme, we consider the noise w_{eq} at the output of filter h; we find

$$\begin{aligned} \mathcal{P}_{w_{eq}}(f) &= \tfrac{1}{4}\,|\mathcal{H}(f)|^2\,2N_0\,1(f + f_0) \\ &= \frac{\mathcal{H}_0^2}{2}\,N_0\,\mathrm{rect}\left(\frac{f}{2B_a}\right) \end{aligned} \tag{1.326}$$

Being now w WSS, $w^{(bb)}$ is uncorrelated with $w^{(bb)*}$ and thus w_{eq} with w_{eq}^*. Then, from

$$w_o(t) = w_{eq,I}(t) \tag{1.327}$$

and using (1.285) it follows

$$\mathcal{P}_{w_0}(f) = \frac{\mathcal{H}_0^2}{4}\,N_0\,\mathrm{rect}\left(\frac{f}{2B_a}\right) \tag{1.328}$$

and

$$\mathtt{M}_{w_0} = \frac{\mathcal{H}_0^2}{4}\,N_0\,2B_a \tag{1.329}$$

In conclusion, using (1.312), we have

$$\Lambda_o = \frac{(\mathcal{H}_0^2/4)\,\mathtt{M}_a \cos^2(\varphi_0 - \varphi_1)}{(\mathcal{H}_0^2/4)\,N_0\,2B_a} = \Gamma \cos^2(\varphi_0 - \varphi_1) \tag{1.330}$$

For $\varphi_1 = \varphi_0$ (1.330) becomes

$$\Lambda_o = \Gamma \tag{1.331}$$

It is interesting to observe that, at the demodulator input, the ratio between the power of the desired signal and the power of the noise in the *passband of* x is given by

$$\Lambda_i = \frac{\mathtt{M}_x}{(N_0/2)\,4B_a} = \frac{\Gamma}{2} \tag{1.332}$$

For $\varphi_1 = \varphi_0$ then

$$\Lambda_o = 2\Lambda_i \tag{1.333}$$

In other words, measuring the noise power in a passband equal to that of the desired signal, the DSB demodulator yields a gain of 2 in signal-to-noise ratio. We will now analyze the case of a SSB signal $x(t)$ (see (1.313)), coherently demodulated, following the scheme of Figure 1.35, where h_{PB} is a filter used to eliminate the noise that otherwise, after the *mixer*, would have fallen within the passband of the desired signal. The ideal frequency response of h_{PB} is given by

$$\mathcal{H}_{PB}(f) = \text{rect}\left(\frac{f - f_0 - B_a/2}{B_a}\right) + \text{rect}\left(\frac{-f - f_0 - B_a/2}{B_a}\right) \tag{1.334}$$

Note that in this scheme we have assumed the phase of the receiver carrier equal to that of the transmitter, to avoid distortion of the desired signal.

Being

$$\mathcal{H}_{PB}^{(bb)}(f) = 2\,\text{rect}\left(\frac{f - B_a/2}{B_a}\right) \tag{1.335}$$

the filter of the baseband-equivalent scheme is given by

$$h_{eq}(t) = \tfrac{1}{2}\, h_{PB}^{(bb)} * h(t) \tag{1.336}$$

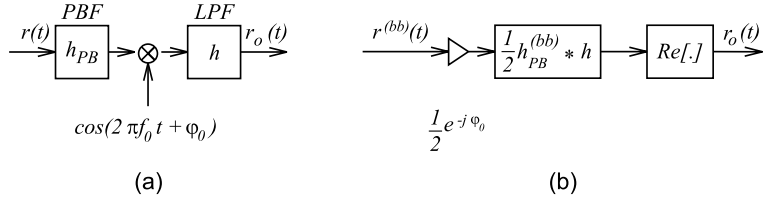


Figure 1.35. (a) Coherent SSB demodulator and (b) baseband-equivalent scheme.

with frequency response

$$\mathcal{H}_{eq}(f) = \mathcal{H}_0 \,\mathrm{rect}\left(\frac{f - B_a/2}{B_a}\right) \tag{1.337}$$

We now evaluate the desired component $x_o(t)$. Using the fact $x^{(bb)} * h_{eq}(t) = \mathcal{H}_0\, x^{(bb)}(t)$, it results in

$$\begin{aligned} x_o(t) &= Re[h_{eq}(\eta) * \tfrac{1}{2}\, e^{-j\varphi_0}\, \tfrac{1}{2}(a(\eta) + j\, a^{(h)}(\eta))\, e^{j\varphi_0}](t) \\ &= \frac{\mathcal{H}_0}{4}\, Re[a(t) + j\, a^{(h)}(t)] \\ &= \frac{\mathcal{H}_0}{4}\, a(t) \end{aligned} \tag{1.338}$$

In the baseband-equivalent scheme, the noise w_{eq} at the output of h_{eq} has a PSD given by

$$\begin{aligned} \mathcal{P}_{w_{eq}}(f) &= \tfrac{1}{4}\, |\mathcal{H}_{eq}(f)|^2\, 2N_0\, 1(f + f_0) \\ &= \frac{N_0}{2}\, \mathcal{H}_0^2\, \mathrm{rect}\left(\frac{f - B_a/2}{B_a}\right) \end{aligned} \tag{1.339}$$

From the relation $w_o = w_{eq,I}$ and using (1.285), which is valid because $w_{eq} \perp w_{eq}^*$, we have

$$\mathcal{P}_{w_o}(f) = \frac{1}{4}[\mathcal{P}_{w_{eq}}(f) + \mathcal{P}_{w_{eq}}(-f)] = \frac{\mathcal{H}_0^2}{8}\, N_0\, \mathrm{rect}\left(\frac{f}{2B_a}\right) \tag{1.340}$$

and

$$\mathtt{M}_{w_o} = \frac{\mathcal{H}_0^2}{8}\, N_0\, 2B_a \tag{1.341}$$

Then we obtain

$$\Lambda_o = \frac{(\mathcal{H}_0^2/16)\, \mathtt{M}_a}{(\mathcal{H}_0^2/8)\, N_0\, 2B_a} \tag{1.342}$$

which using (1.316) and (1.321) can be written as

$$\Lambda_o = \Gamma \tag{1.343}$$

We note that the SSB system yields the same performance (for $\varphi_1 = \varphi_0$) as a DSB system, even though half of the bandwidth is required. Finally, it results in

$$\Lambda_i = \frac{\mathtt{M}_x}{(N_0/2)\, 2B_a} = \Lambda_o \tag{1.344}$$

Observation 1.6

We note that also for the simple examples considered in this section the desired signal is analyzed via the various transformations, whereas the noise is analyzed via the PSD. As a matter of fact, we are typically interested only in the statistical power of the noise at the system output. The demodulated signal $x_o(t)$, on the other hand, must be expressed as the sum of a desired component proportional to $a(t)$ and an orthogonal component that represents the distortion, which is, typically, small and has the same effects as noise.

In the previous example, the considered systems do not introduce any distortion since $x_o(t)$ is proportional to $a(t)$.

1.8 The autocorrelation matrix

Definition

Given the discrete-time wide-sense stationary random process $\{x(k)\}$, we introduce the random vector with N components

$$\mathbf{x}^T(k) = [x(k), x(k-1), \ldots, x(k-N+1)] \tag{1.345}$$

The $N \times N$ autocorrelation matrix of $\mathbf{x}^*(k)$ is given by

$$\mathbf{R} = E[\mathbf{x}^*(k)\mathbf{x}^T(k)] = \begin{bmatrix} \mathtt{r}_x(0) & \mathtt{r}_x(-1) & \cdots & \mathtt{r}_x(-N+1) \\ \mathtt{r}_x(1) & \mathtt{r}_x(0) & \cdots & \mathtt{r}_x(-N+2) \\ \vdots & \vdots & \ddots & \cdots \\ \mathtt{r}_x(N-1) & \mathtt{r}_x(N-2) & \cdots & \mathtt{r}_x(0) \end{bmatrix} \tag{1.346}$$

Properties

1. $\mathbf{R}$ is *Hermitian*: $\mathbf{R}^H = \mathbf{R}$.
   For real random processes $\mathbf{R}$ is *symmetric*: $\mathbf{R}^T = \mathbf{R}$.

2. $\mathbf{R}$ is a *Toeplitz matrix*, i.e. all elements along any diagonal are equal.

3. $\mathbf{R}$ is *positive semi-definite* and almost always *positive definite*. Indeed, taking an arbitrary vector $\mathbf{v}^T = [v_0, \ldots, v_{N-1}]$, and letting $y = \mathbf{x}^T(k)\mathbf{v}$, we have

$$E[|y|^2] = E[\mathbf{v}^H\mathbf{x}^*(k)\mathbf{x}^T(k)\mathbf{v}] = \mathbf{v}^H\mathbf{R}\mathbf{v} = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} v_i^*\mathtt{r}_x(i-j)v_j \geq 0 \tag{1.347}$$

   If $\mathbf{v}^H\mathbf{R}\mathbf{v} > 0$, $\forall\mathbf{v}$, then $\mathbf{R}$ is said to be *positive definite* and all its *principal minor* determinants are positive; in particular $\mathbf{R}$ is *non-singular*.

Eigenvalues

We indicate by $\det[\mathbf{R}]$ the determinant of a matrix $\mathbf{R}$. The eigenvalues of $\mathbf{R}$ are the solutions λ_i, $i = 1, \ldots, N$, of the *characteristic equation* of order N

$$\det[\mathbf{R} - \lambda\mathbf{I}] = 0 \tag{1.348}$$

and the corresponding column eigenvectors $\mathbf{u}_i$ satisfy the equation

$$\mathbf{R}\mathbf{u}_i = \lambda_i\mathbf{u}_i \tag{1.349}$$

Example 1.8.1
Let $\{w(k)\}$ be a white noise process. Its autocorrelation matrix $\mathbf{R}$ assumes the form

$$\mathbf{R} = \begin{bmatrix} \sigma_w^2 & 0 & \cdots & 0 \\ 0 & \sigma_w^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_w^2 \end{bmatrix} \tag{1.350}$$

from which it follows that

$$\lambda_1 = \lambda_2 = \cdots = \lambda_N = \sigma_w^2 \tag{1.351}$$

and

$$\mathbf{u}_i \text{ can be any arbitrary vector} \qquad 1 \leq i \leq N \tag{1.352}$$

Example 1.8.2
We define a complex-valued sinusoid as

$$x(k) = e^{j(\omega k + \varphi)} \qquad \omega = 2\pi f T_c \tag{1.353}$$

with φ uniform r.v. in $[0, 2\pi)$. The matrix $\mathbf{R}$ is given by

$$\mathbf{R} = \begin{bmatrix} 1 & e^{-j\omega} & \cdots & e^{-j(N-1)\omega} \\ e^{j\omega} & 1 & \cdots & e^{-j(N-2)\omega} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j(N-1)\omega} & e^{j(N-2)\omega} & \cdots & 1 \end{bmatrix} \tag{1.354}$$

One can see that the rank of $\mathbf{R}$ is 1 and it will therefore have only one eigenvalue. A possible solution is given by

$$\lambda_1 = N \tag{1.355}$$

and the relative eigenvector is

$$\mathbf{u}_1^T = [1, e^{j\omega}, \ldots, e^{j(N-1)\omega}] \tag{1.356}$$

Other properties

1. From $\mathbf{R}^m \mathbf{u} = \lambda^m \mathbf{u}$ we obtain the relations of Table 1.7.
2. If the eigenvalues are distinct, then the eigenvectors are linearly independent:

$$\sum_{i=1}^{N} c_i \mathbf{u}_i \neq \mathbf{0} \tag{1.357}$$

for all combinations of $\{c_i\}$, $i = 1, 2, \ldots, N$, not all equal to zero. Therefore, in this case, the eigenvectors form a basis in $\Re^N$.

Table 1.7 Correspondence between eigenvalues and eigenvectors of four matrices.

	$\mathbf{R}$	$\mathbf{R}^m$	$\mathbf{R}^{-1}$	$\mathbf{I} - \mu\mathbf{R}$
Eigenvalue	λ_i	λ_i^m	λ_i^{-1}	$(1 - \mu\lambda_i)$
Eigenvector	$\mathbf{u}_i$	$\mathbf{u}_i$	$\mathbf{u}_i$	$\mathbf{u}_i$

3. The *trace* of a matrix $\mathbf{R}$ is defined as the sum of the elements of the main diagonal, and we indicate it with tr[$\mathbf{R}$]. It holds

$$\operatorname{tr}\mathbf{R} = \sum_{i=1}^{N} \lambda_i \tag{1.358}$$

Eigenvalue analysis for Hermitian matrices

As previously seen, the autocorrelation matrix $\mathbf{R}$ is Hermitian. Consequently, it enjoys the following properties, valid for Hermitian matrices:

1. The eigenvalues of a Hermitian matrix are real.
   By left multiplying both sides of (1.349) by $\mathbf{u}_i^H$, it follows

$$\mathbf{u}_i^H \mathbf{R} \mathbf{u}_i = \lambda_i \mathbf{u}_i^H \mathbf{u}_i \tag{1.359}$$

   from which, using (1.13), one gets

$$\lambda_i = \frac{\mathbf{u}_i^H \mathbf{R} \mathbf{u}_i}{\mathbf{u}_i^H \mathbf{u}_i} = \frac{\mathbf{u}_i^H \mathbf{R} \mathbf{u}_i}{\|\mathbf{u}_i\|^2} \tag{1.360}$$

   The ratio (1.360) is defined as *Rayleigh quotient*. As $\mathbf{R}$ is positive semi-definite, $\mathbf{u}_i^H \mathbf{R} \mathbf{u}_i \geq 0$, from which $\lambda_i \geq 0$.

2. If the eigenvalues of $\mathbf{R}$ are distinct, then the eigenvectors are orthogonal. In fact, from (1.349) one gets:

$$\mathbf{u}_i^H \mathbf{R} \mathbf{u}_j = \lambda_j \mathbf{u}_i^H \mathbf{u}_j \tag{1.361}$$

$$\mathbf{u}_i^H \mathbf{R} \mathbf{u}_j = \lambda_i \mathbf{u}_i^H \mathbf{u}_j \tag{1.362}$$

   Subtracting the second equation from the first:

$$0 = (\lambda_j - \lambda_i) \mathbf{u}_i^H \mathbf{u}_j \tag{1.363}$$

   and since $\lambda_j - \lambda_i \neq 0$ by hypothesis, it follows $\mathbf{u}_i^H \mathbf{u}_j = 0$.

3. If the eigenvalues of $\mathbf{R}$ are distinct and their corresponding eigenvectors are normalized, i.e.

$$\|\mathbf{u}_i\|^2 = \mathbf{u}_i^H \mathbf{u}_i = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{1.364}$$

   then the matrix $\mathbf{U}$, whose columns are the eigenvectors of $\mathbf{R}$,

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N] \tag{1.365}$$

is a unitary matrix, that is

$$\mathbf{U}^{-1} = \mathbf{U}^H \tag{1.366}$$

This property is an immediate consequence of the orthogonality of the eigenvectors $\{\mathbf{u}_i\}$. Moreover, if we define the matrix $\mathbf{\Lambda}$ as

$$\mathbf{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \tag{1.367}$$

we get

$$\mathbf{U}^H \mathbf{R} \mathbf{U} = \mathbf{\Lambda} \tag{1.368}$$

From (1.368) we obtain the following important relations:

$$\mathbf{R} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^H = \sum_{i=1}^{N} \lambda_i \mathbf{u}_i \mathbf{u}_i^H \tag{1.369}$$

and

$$\mathbf{I} - \mu\mathbf{R} = \mathbf{U}(\mathbf{I} - \mu\mathbf{\Lambda})\mathbf{U}^H = \sum_{i=1}^{N} (1 - \mu\lambda_i)\mathbf{u}_i \mathbf{u}_i^H \tag{1.370}$$

4. The eigenvalues of a positive semi-definite autocorrelation matrix $\mathbf{R}$ and the PSD of x are related by the inequalities,

$$\min_f \{\mathcal{P}_x(f)\} \leq \lambda_i \leq \max_f \{\mathcal{P}_x(f)\} \qquad i = 1, \ldots, N \tag{1.371}$$

In fact, let $U_i(f)$ be the Fourier transform of the sequence represented by the elements of $\mathbf{u}_i$:

$$U_i(f) = \sum_{n=1}^{N} u_{i,n} e^{-j2\pi f n T_c} \tag{1.372}$$

where $u_{i,n}$ is the n-th element of the eigenvector $\mathbf{u}_i$. Observing that

$$\mathbf{u}_i^H \mathbf{R} \mathbf{u}_i = \sum_{n=1}^{N} \sum_{m=1}^{N} u_{i,n}^* \mathrm{r}_x(n-m) u_{i,m} \tag{1.373}$$

and using (1.248) and (1.372), the preceding equation can be written as

$$\begin{aligned} \mathbf{u}_i^H \mathbf{R} \mathbf{u}_i &= \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \mathcal{P}_x(f) \sum_{n=1}^{N} u_{i,n}^* e^{j2\pi f n T_c} \sum_{m=1}^{N} u_{i,m} e^{-j2\pi f m T_c} \, df \\ &= \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \mathcal{P}_x(f) \, |U_i(f)|^2 \, df \end{aligned} \tag{1.374}$$

Substituting the latter result in (1.360) one finds

$$\lambda_i = \frac{\int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \mathcal{P}_x(f)\,|U_i(f)|^2\,df}{\int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} |U_i(f)|^2\,df} \tag{1.375}$$

from which (1.371) follows.

If we indicate with $\lambda_{\min}$ and $\lambda_{\max}$, respectively, the minimum and maximum eigenvalue of $\mathbf{R}$, in view of the latter point we can define the *eigenvalue spread* as:

$$\chi(\mathbf{R}) = \frac{\lambda_{\max}}{\lambda_{\min}} \leq \frac{\max_f\{\mathcal{P}_x(f)\}}{\min_f\{\mathcal{P}_x(f)\}} \tag{1.376}$$

From (1.376) we observe that $\chi(\mathbf{R})$ may assume large values in the case $\mathcal{P}_x(f)$ exhibits large variations. Moreover, $\chi(\mathbf{R})$ assumes the minimum value of 1 for a white process.

1.9 Examples of random processes

Before reviewing some important random processes, we recall the definition of Gaussian complex-valued random vector.

Example 1.9.1
A r.v. with a Gaussian distribution can be generated from two r.v.s with uniform distribution (see Appendix 1.B for an illustration of the method).

Example 1.9.2
Let $\mathbf{x}^T = [x_1, \ldots, x_N]$ be a real Gaussian random vector, $x_i \in \mathcal{N}\left(\mathtt{m}_i, \sigma_i^2\right)$. The joint probability density function is

$$p_{\mathbf{x}}(\boldsymbol{\xi}) = [(2\pi)^N \det \mathbf{C}_N]^{-\frac{1}{2}} e^{-\frac{1}{2}(\boldsymbol{\xi}-\mathbf{m}_x)^T \mathbf{C}_N^{-1}(\boldsymbol{\xi}-\mathbf{m}_x)} \tag{1.377}$$

where $\boldsymbol{\xi}^T = [\xi_1, \ldots, \xi_N]$, $\mathbf{m}_x = E[\mathbf{x}]$ is the vector of mean values and $\mathbf{C}_N = E[(\mathbf{x} - \mathbf{m}_x)(\mathbf{x} - \mathbf{m}_x)^T]$ is the covariance matrix.

Example 1.9.3
Let $\mathbf{x}^T = [x_{1,I} + jx_{1,Q}, \ldots, x_{N,I} + jx_{N,Q}]$ be a complex-valued Gaussian random vector. If the in-phase component $x_{i,I}$ and the quadrature component $x_{i,Q}$ are uncorrelated,

$$E[(x_{i,I} - \mathtt{m}_{x_{i,I}})(x_{i,Q} - \mathtt{m}_{x_{i,Q}})] = 0 \qquad i = 1, 2, \ldots, N \tag{1.378}$$

and, moreover,

$$\sigma^2_{x_{i,I}} = \sigma^2_{x_{i,Q}} = \tfrac{1}{2}\sigma^2_{x_i} \tag{1.379}$$

then the joint probability density function is

$$p_{\mathbf{x}}(\boldsymbol{\xi}) = [\pi^N \det \mathbf{C}_N]^{-1} e^{-(\boldsymbol{\xi}-\mathbf{m}_x)^H \mathbf{C}_N^{-1}(\boldsymbol{\xi}-\mathbf{m}_x)} \tag{1.380}$$

with the vector of mean values and the covariance matrix given by

$$\mathbf{m}_x = E[\mathbf{x}] = E[\mathbf{x}_I] + jE[\mathbf{x}_Q] \tag{1.381}$$

$$\mathbf{C}_N = E[(\mathbf{x}-\mathbf{m}_x)(\mathbf{x}-\mathbf{m}_x)^H] \tag{1.382}$$

The vector $\mathbf{x}$ is called a *circularly symmetric Gaussian random vector*.

Example 1.9.4
Let $\mathbf{x}^T = [x_1(t_1), \ldots, x_N(t_N)]$ be a complex-valued Gaussian (vector) process, with each element $x_i(t_i)$ having real and imaginary components that are uncorrelated Gaussian r.v.s with zero mean and equal variance for all values of t_i. The vector $\mathbf{x}$ is called a *circularly symmetric Gaussian random process*. The joint probability density function in this case results in

$$p_{\mathbf{x}}(\boldsymbol{\xi}) = [\pi^N \det \mathbf{C}]^{-1} e^{-\boldsymbol{\xi}^H \mathbf{C}^{-1} \boldsymbol{\xi}} \tag{1.383}$$

where $\mathbf{C}$ is the covariance matrix of $[x_1(t_1), x_2(t_2), \ldots, x_N(t_N)]$.

Example 1.9.5
Let $x(t) = A\sin(2\pi f t + \varphi)$ be a real-valued sinusoidal signal with φ r.v. uniform in $[0, 2\pi)$, for which we will use the notation $\varphi \in \mathcal{U}[0, 2\pi)$. The mean of x is

$$\begin{aligned} \mathfrak{m}_x(t) &= E[x(t)] \\ &= \int_0^{2\pi} \frac{1}{2\pi} A \sin(2\pi f t + a)\, da \\ &= 0 \end{aligned} \tag{1.384}$$

and the autocorrelation function is given by

$$\begin{aligned} \mathfrak{r}_x(\tau) &= \int_0^{2\pi} \frac{1}{2\pi} A \sin(2\pi f t + a) A \sin[2\pi f(t-\tau) + a]\, da \\ &= \frac{A^2}{2} \cos(2\pi f \tau) \end{aligned} \tag{1.385}$$

Example 1.9.6
Given N real-valued sinusoidal signals

$$x(t) = \sum_{i=1}^{N} A_i \sin(2\pi f_i t + \varphi_i) \tag{1.386}$$

with $\{\varphi_i\}$ statistically independent uniform r.v.s in $[0, 2\pi)$, from Example 1.9.5 it is able to obtain the mean value

$$\mathtt{m}_x(t) = \sum_{i=1}^{N} \mathtt{m}_{x_i}(t) = 0 \tag{1.387}$$

and the autocorrelation function

$$\mathtt{r}_x(\tau) = \sum_{i=1}^{N} \frac{A_i^2}{2} \cos(2\pi f_i \tau) \tag{1.388}$$

We note that, according to the Definition 1.12, page 48, the process (1.386) is not asymptotically uncorrelated.

Example 1.9.7
Given N complex-valued sinusoidal signals

$$x(t) = \sum_{i=1}^{N} A_i \, e^{j(2\pi f_i t + \varphi_i)} \tag{1.389}$$

with $\{\varphi_i\}$ statistically independent uniform r.v.s in $[0, 2\pi)$, following a similar procedure to that used in Examples 1.9.5 and 1.9.6, we find

$$\mathtt{r}_x(\tau) = \sum_{i=1}^{N} |A_i|^2 \, e^{j2\pi f_i \tau} \tag{1.390}$$

We note that the process (1.390) is not asymptotically uncorrelated.

Example 1.9.8
Let the discrete-time random process $y(k) = x(k) + w(k)$ be given by the sum of the random process $x(k)$ of Example 1.9.7 and white noise $w(k)$ with variance σ_w^2. Moreover, we assume $\{x(k)\}$ and $\{w(k)\}$ are uncorrelated. In this case

$$\mathtt{r}_y(n) = \sum_{i=1}^{N} |A_i|^2 \, e^{j2\pi f_i n T_c} + \sigma_w^2 \delta_n \tag{1.391}$$

Example 1.9.9
We consider a signal obtained by pulse-amplitude modulation (PAM), expressed as

$$y(t) = \sum_{k=-\infty}^{+\infty} x(k) \, h_{Tx}(t - kT) \tag{1.392}$$

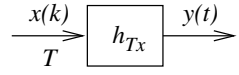


Figure 1.36. Modulator of a PAM system as interpolator filter.

The signal $y(t)$ is the output of the system shown in Figure 1.36, where h_{Tx} is a finite-energy pulse, and $\{x(k)\}$ is a discrete-time (with T-spaced samples) WSS sequence, having power spectral density $\mathcal{P}_x$. We note that $\mathcal{P}_x(f)$ is a periodic function of period $1/T$.

Let $\mathtt{r}_{h_{Tx}}(\tau)$ be the deterministic autocorrelation of the signal h_{Tx}:

$$\mathtt{r}_{h_{Tx}}(\tau) = \int_{-\infty}^{+\infty} h_{Tx}(t) h_{Tx}^*(t-\tau)\, dt = [h_{Tx}(t) * h_{Tx}^*(-t)](\tau) \tag{1.393}$$

with Fourier transform equal to $|\mathcal{H}_{Tx}(f)|^2$. In general y is a cyclostationary process of period T. In fact we have

1. *Mean*

$$\mathtt{m}_y(t) = \mathtt{m}_x \sum_{k=-\infty}^{+\infty} h_{Tx}(t - kT) \tag{1.394}$$

2. *Correlation*

$$\mathtt{r}_y(t, t-\tau) = \sum_{i=-\infty}^{+\infty} \mathtt{r}_x(i) \sum_{m=-\infty}^{+\infty} h_{Tx}(t - (i+m)\,T) h_{Tx}^*(t - \tau - mT) \tag{1.395}$$

If we introduce the average spectral analysis

$$\bar{\mathtt{m}}_y = \frac{1}{T} \int_0^T \mathtt{m}_y(t)\, dt = \mathtt{m}_x \mathcal{H}_{Tx}(0) \tag{1.396}$$

$$\bar{\mathtt{r}}_y(\tau) = \frac{1}{T} \int_0^T \mathtt{r}_y(t, t-\tau)\, dt = \frac{1}{T} \sum_{i=-\infty}^{+\infty} \mathtt{r}_x(i) \mathtt{r}_{h_{Tx}}(\tau - iT) \tag{1.397}$$

and

$$\bar{\mathcal{P}}_y(f) = \mathcal{F}[\bar{\mathtt{r}}_y(\tau)] = \left| \frac{1}{T} \mathcal{H}_{Tx}(f) \right|^2 \mathcal{P}_x(f) \tag{1.398}$$

we observe that the modulator of a PAM system may be regarded as an interpolator filter with frequency response $\mathcal{H}_{Tx}/T$.

3. *Average power for a white noise input*
For a white noise input with power $\mathtt{M}_x$, from (1.397) the average statistical power of the output signal is given by

$$\bar{\mathtt{M}}_y = \mathtt{M}_x \frac{E_h}{T} \tag{1.399}$$

where $E_h = \int_{-\infty}^{+\infty} |h_{Tx}(t)|^2\, dt$ is the energy of h_{Tx}.

4. *Moments of y for a circularly symmetric i.i.d. input*
Let $x(k)$ be a complex-valued random circularly symmetric sequence with zero mean (see (1.378) and (1.379)), i.e. letting

$$x_I(k) = Re[x(k)] \qquad x_Q(k) = Im[x(k)] \tag{1.400}$$

we have

$$E[x_I^2(k)] = E[x_Q^2(k)] = \frac{E[|x(k)|^2]}{2} \tag{1.401}$$

and

$$E[x_I(k)\, x_Q(k)] = 0 \tag{1.402}$$

These two relations can be merged into one,

$$E[x^2(k)] = E[x_I^2(k)] - E[x_Q^2(k)] + 2j\, E[x_I(k)\, x_Q(k)] = 0 \tag{1.403}$$

Filtering the i.i.d. input signal $\{x(k)\}$ by using the scheme depicted in Figure 1.36, and observing the relation

$$\mathtt{r}_{yy^*}(t, t-\tau) = \sum_{i=-\infty}^{+\infty} \mathtt{r}_{xx^*}(i) \sum_{m=-\infty}^{+\infty} h_{Tx}(t-(i+m)T) h_{Tx}(t-\tau-mT) \tag{1.404}$$

then

$$\mathtt{r}_{xx^*}(i) = E[x^2(k)]\delta(i) = 0 \tag{1.405}$$

and

$$\mathtt{r}_{yy^*}(t, t-\tau) = 0 \tag{1.406}$$

that is $y(t) \perp y^*(t)$. In particular we find that $y(t)$ is circularly symmetric, i.e.

$$E[y^2(t)] = 0 \tag{1.407}$$

We note that the condition (1.406) can be obtained assuming the less stringent condition that $x \perp x^*$; on the other hand, this requires that the following two conditions are verified

$$\mathtt{r}_{x_I}(i) = \mathtt{r}_{x_Q}(i) \tag{1.408}$$

and

$$\mathtt{r}_{x_I x_Q}(i) = -\mathtt{r}_{x_I x_Q}(-i) \tag{1.409}$$

Observation 1.7
It can be shown that if the filter h_{Tx} has a bandwidth smaller than $1/(2T)$ and x is a WSS sequence, then y is WSS with spectral density given by (1.398).

Example 1.9.10
Let us consider a PAM signal sampled with period $T_Q = T/Q_0$, where Q_0 is a positive integer number. Let

$$y_q = y(q\,T_Q) \qquad h_p = h_{Tx}(p\,T_Q) \tag{1.410}$$

from (1.392) it follows

$$y_q = \sum_{k=-\infty}^{+\infty} x(k)\,h_{q-kQ_0} \tag{1.411}$$

If $Q_0 \neq 1$, (1.411) describes the input–output relation of an interpolator filter (see (1.609)). We recall the statistical analysis given in Table 1.6, page 52. We denote with $\mathcal{H}(f)$ the Fourier transform (see (1.84)) and with $\mathtt{r}_h(n)$ the deterministic autocorrelation (see (1.260)) of the sequence $\{h_p\}$. We also assume that $\{x(k)\}$ is a WSS random sequence with mean $\mathtt{m}_x$ and autocorrelation $\mathtt{r}_x(n)$. In general, $\{y_q\}$ is a cyclostationary random sequence of period Q_0 with

1. *Mean*

$$\mathtt{m}_y(q) = \mathtt{m}_x \sum_{k=-\infty}^{+\infty} h_{q-kQ_0} \tag{1.412}$$

2. *Correlation*

$$\mathtt{r}_y(q, q-n) = \sum_{i=-\infty}^{+\infty} \mathtt{r}_x(i) \sum_{m=-\infty}^{+\infty} h_{q-(i+m)Q_0}\,h^*_{q-n-m\,Q_0} \tag{1.413}$$

By the average spectral analysis we obtain

$$\bar{\mathtt{m}}_y = \frac{1}{Q_0} \sum_{q=0}^{Q_0-1} \mathtt{m}_y(q) = \mathtt{m}_x\,\frac{\mathcal{H}(0)}{Q_0} \tag{1.414}$$

where

$$\mathcal{H}(0) = \sum_{p=-\infty}^{+\infty} h_p \tag{1.415}$$

and

$$\bar{\mathtt{r}}_y(n) = \frac{1}{Q_0} \sum_{q=0}^{Q_0-1} \mathtt{r}_y(q, q-n) = \frac{1}{Q_0} \sum_{i=-\infty}^{+\infty} \mathtt{r}_x(i)\,\mathtt{r}_h(n - i\,Q_0) \tag{1.416}$$

Consequently, the average PSD is given by

$$\bar{\mathcal{P}}_y(f) = T_Q\,\mathcal{F}[\bar{\mathtt{r}}_y(n)] = \left|\frac{1}{Q_0}\,\mathcal{H}(f)\right|^2 \mathcal{P}_x(f) \tag{1.417}$$

If $\{x(k)\}$ is white noise with power $\mathtt{M}_x$, from (1.416) it results in

$$\bar{\mathtt{r}}_y(n) = \mathtt{M}_x\,\frac{\mathtt{r}_h(n)}{Q_0} \tag{1.418}$$

In particular the average power of the filter output signal is given by

$$\bar{\mathtt{M}}_y = \mathtt{M}_x\,\frac{E_h}{Q_0} \tag{1.419}$$

where $E_h = \sum_{p=-\infty}^{+\infty} |h_p|^2$ is the energy of $\{h_p\}$. We point out that the condition $\bar{\mathtt{M}}_y = \mathtt{M}_x$ is satisfied if the energy of the filter impulse response is equal to the interpolation factor Q_0.

1.10 Matched filter

Referring to Figure 1.37, we consider a finite-energy signal pulse g in the presence of additive noise w having zero mean and power spectral density $\mathcal{P}_w$. The signal

$$x(t) = g(t) + w(t) \tag{1.420}$$

is filtered with a filter having impulse response g_M. We indicate with g_u and w_u respectively the desired signal and the noise component at the output:

$$g_u(t) = g_M * g(t) \tag{1.421}$$

$$w_u(t) = g_M * w(t) \tag{1.422}$$

The output is expressed as

$$y(t) = g_u(t) + w_u(t) \tag{1.423}$$

We now suppose that y is observed at a given instant t_0. The problem is to determine g_M so that the ratio between the squared amplitude $g_u(t_0)$ and the power of the noise component $w_u(t_0)$ is maximum, i.e.

$$g_M : \max_{g_M} \frac{|g_u(t_0)|^2}{E[|w_u(t_0)|^2]} \tag{1.424}$$

The optimum filter has frequency response

$$\mathcal{G}_M(f) = K \frac{\mathcal{G}^*(f)}{\mathcal{P}_w(f)} e^{-j2\pi f t_0} \tag{1.425}$$

where K is a constant. In other words, the best filter selects the frequency components of the desired input signal and weights them with weights that are inversely proportional to the noise level.

Proof. $g_u(t_0)$ coincides with the inverse Fourier transform of $\mathcal{G}_M(f)\mathcal{G}(f)$ evaluated in $t = t_0$, while the power of $w_u(t_0)$ is equal to

$$\mathtt{r}_{w_u}(0) = \int_{-\infty}^{+\infty} \mathcal{P}_w(f) |\mathcal{G}_M(f)|^2 \, df \tag{1.426}$$

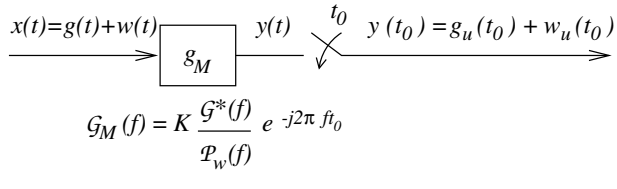


Figure 1.37. Reference scheme for the matched filter.

Then

$$\begin{aligned}\frac{|g_u(t_0)|^2}{\mathtt{r}_{w_u}(0)} &= \frac{\left|\int_{-\infty}^{+\infty} \mathcal{G}_M(f)\mathcal{G}(f)e^{j2\pi f t_0}\,df\right|^2}{\int_{-\infty}^{+\infty} \mathcal{P}_w(f)|\mathcal{G}_M(f)|^2\,df} \\ &= \frac{\left|\int_{-\infty}^{+\infty} \mathcal{G}_M(f)\sqrt{\mathcal{P}_w(f)}\,\frac{\mathcal{G}(f)}{\sqrt{\mathcal{P}_w(f)}}e^{j2\pi f t_0}\,df\right|^2}{\int_{-\infty}^{+\infty} \mathcal{P}_w(f)|\mathcal{G}_M(f)|^2\,df}\end{aligned} \tag{1.427}$$

where the integrand at the numerator was divided and multiplied by $\sqrt{\mathcal{P}_w(f)}$. Implicitly it is assumed that $\mathcal{P}_w(f) \neq 0$. Applying the Schwarz inequality (see Section 1.1) to the functions

$$\mathcal{G}_M(f)\sqrt{\mathcal{P}_w(f)} \tag{1.428}$$

and

$$\frac{\mathcal{G}^*(f)}{\sqrt{\mathcal{P}_w(f)}}\,e^{-j2\pi f t_0} \tag{1.429}$$

it turns out

$$\frac{|g_u(t_0)|^2}{\mathtt{r}_{w_u}(0)} \leq \int_{-\infty}^{+\infty}\left|\frac{\mathcal{G}(f)}{\sqrt{\mathcal{P}_w(f)}}e^{j2\pi f t_0}\right|^2 df = \int_{-\infty}^{+\infty}\left|\frac{\mathcal{G}(f)}{\sqrt{\mathcal{P}_w(f)}}\right|^2 df \tag{1.430}$$

Therefore the maximum value is equal to the right-hand side of (1.430) and is achieved for

$$\mathcal{G}_M(f)\sqrt{\mathcal{P}_w(f)} = K\frac{\mathcal{G}^*(f)}{\sqrt{\mathcal{P}_w(f)}}\,e^{-j2\pi f t_0} \tag{1.431}$$

where K is a constant. From (1.431) the solution (1.425) follows immediately.

Matched filter in the presence of white noise

If w is white, then $\mathcal{P}_w(f) = \mathcal{P}_w$ is a constant and the optimum solution (1.425) becomes

$$\mathcal{G}_M(f) = K\mathcal{G}^*(f)e^{-j2\pi f t_0} \tag{1.432}$$

Correspondingly, the filter has impulse response

$$g_M(t) = Kg^*(t_0 - t) \tag{1.433}$$

from which comes the name of *matched filter* (MF), i.e. matched to the input signal pulse. The desired signal pulse at the filter output has the frequency response

$$\mathcal{G}_u(f) = K|\mathcal{G}(f)|^2 e^{-j2\pi f t_0} \tag{1.434}$$

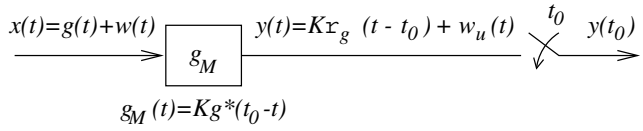


Figure 1.38. Matched filter for an input pulse in the presence of white noise.

From the definition of the autocorrelation function of g,

$$\mathtt{r}_g(\tau) = \int_{-\infty}^{+\infty} g(a)g^*(a-\tau)\,da \tag{1.435}$$

then, as depicted in Figure 1.38,

$$g_u(t) = K\,\mathtt{r}_g(t-t_0) \tag{1.436}$$

If E_g is the energy of g, using the relation $E_g = \mathtt{r}_g(0)$ the maximum of the functional (1.424) becomes

$$\frac{|g_u(t_0)|^2}{\mathtt{r}_{w_u}(0)} = \frac{|K|^2\mathtt{r}_g^2(0)}{\mathcal{P}_w|K|^2\mathtt{r}_g(0)} = \frac{E_g}{\mathcal{P}_w} \tag{1.437}$$

In Figure 1.39 the different pulse shapes are illustrated for a signal pulse g with limited duration t_g. Note that in this case the matched filter has also limited duration and it is causal if $t_0 \geq t_g$.

Example 1.10.1 (MF for a rectangular pulse)
Let

$$g(t) = \mathtt{w}_T(t) = \mathrm{rect}\left(\frac{t-T/2}{T}\right) \tag{1.438}$$

with

$$\mathtt{r}_g(\tau) = T\left(1-\frac{|\tau|}{T}\right)\mathrm{rect}\left(\frac{\tau}{2T}\right) \tag{1.439}$$

For $t_0 = T$, the matched filter is proportional to g

$$g_M(t) = K\,\mathtt{w}_T(t) \tag{1.440}$$

and the output pulse in the absence of noise is equal to

$$g_u(t) = \begin{cases} KT\left(1-\left|\dfrac{t-T}{T}\right|\right) & 0 < t < 2T \\ 0 & \text{elsewhere} \end{cases} \tag{1.441}$$

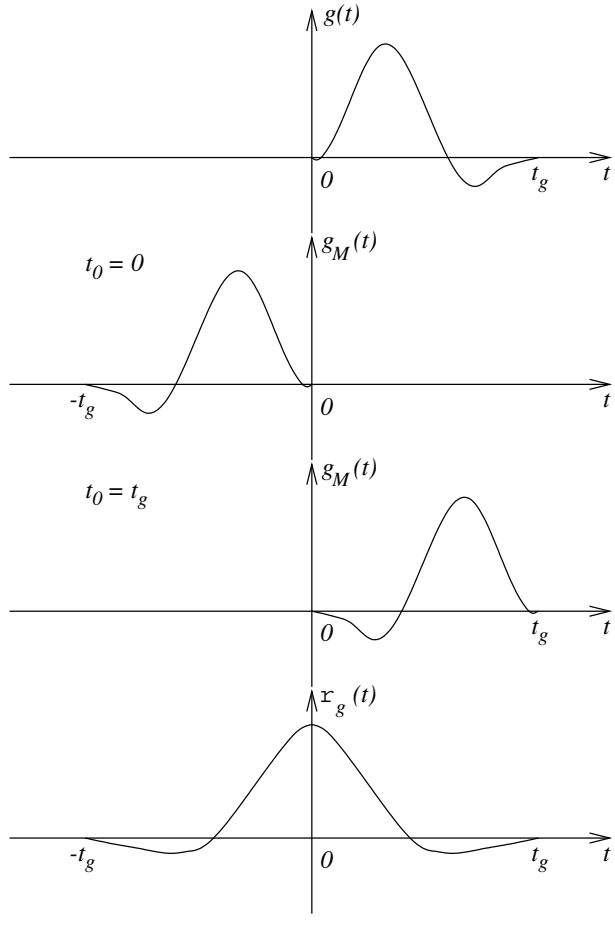


Figure 1.39. Various pulse shapes related to a matched filter.

1.11 Ergodic random processes

The functions that have been introduced in the previous sections for the analysis of random processes give a valid statistical description of an ensemble of realizations of a random process. We investigate now the possibility of moving from ensemble averaging to time averaging, that is we consider the problem of estimating a statistical descriptor of a random process from the observation of a single realization. Let x be a discrete-time WSS random process having mean m_x. If in the limit it holds[18]

$$\lim_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} x(k) = E[x(k)] = \mathrm{m}_x \tag{1.442}$$

[18] The *limit* is meant in the mean-square sense, that is the variance of the r.v. $\left(\frac{1}{K}\sum_{k=0}^{K-1} x(k) - \mathrm{m}_x\right)$ vanishes for $K \to \infty$.

then x is said to be *ergodic in the mean*. In other words, for a process for which the above limit is true, the time-average of samples tends to the statistical mean as the number of samples increases. We note that the existence of the limit (1.442) implies the condition

$$\lim_{K\to\infty} E\left[\left|\frac{1}{K}\sum_{k=0}^{K-1} x(k) - \mathrm{m}_x\right|^2\right] = 0 \tag{1.443}$$

or equivalently

$$\lim_{K\to\infty} \frac{1}{K}\sum_{n=-(K-1)}^{K-1}\left[1 - \frac{|n|}{K}\right] \mathrm{c}_x(n) = 0 \tag{1.444}$$

From (1.444) we see that for a random process to be ergodic in the mean, some conditions on the second-order statistics must be verified. Analogously to definition (1.442), we say that x is *ergodic in correlation* if in the limits it holds:

$$\lim_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K-1} x(k)x^*(k-n) = E[x(k)x^*(k-n)] = \mathrm{r}_x(n) \tag{1.445}$$

Also for processes that are ergodic in correlation one could get a condition of ergodicity similar to that expressed by the limit (1.444). Let $y(k) = x(k)x^*(k-n)$. Observing (1.445) and (1.442), we find that the ergodicity in correlation of the process x is equivalent to the ergodicity in the mean of the process y. Therefore it is easy to deduce that the condition (1.444) for y translates into a condition on the statistical moments of the fourth order for x.

In practice, we *will assume all stationary processes to be ergodic*; ergodicity is, however, difficult to prove for non-Gaussian random processes. We will not consider particular processes that are not ergodic such as $x(k) = A$, where A is a random variable, or $x(k)$ equal to the sum of sinusoidal signals (see (1.386)).

The property of ergodicity assumes a fundamental importance if we observe that from a single realization it is possible to obtain an estimate of the autocorrelation function and from this, the power spectral density. Alternatively, one could prove that under the hypothesis[19]

$$\sum_{n=-\infty}^{+\infty} |n|\, \mathrm{r}_x(n) < \infty \tag{1.446}$$

the following limit holds:

$$\lim_{K\to\infty} E\left[\frac{1}{KT_c}\left|T_c\sum_{k=0}^{K-1} x(k)\, e^{-j2\pi f k T_c}\right|^2\right] = \mathcal{P}_x(f) \tag{1.447}$$

Then, exploiting the ergodicity of a WSS random process, one obtains the relations among the process itself, its autocorrelation function and power spectral density shown

[19] We note that for random processes with non-zero mean and/or sinusoidal components this property is not verified. Therefore it is usually recommended that the deterministic components of the process be removed before the spectral estimation is performed.

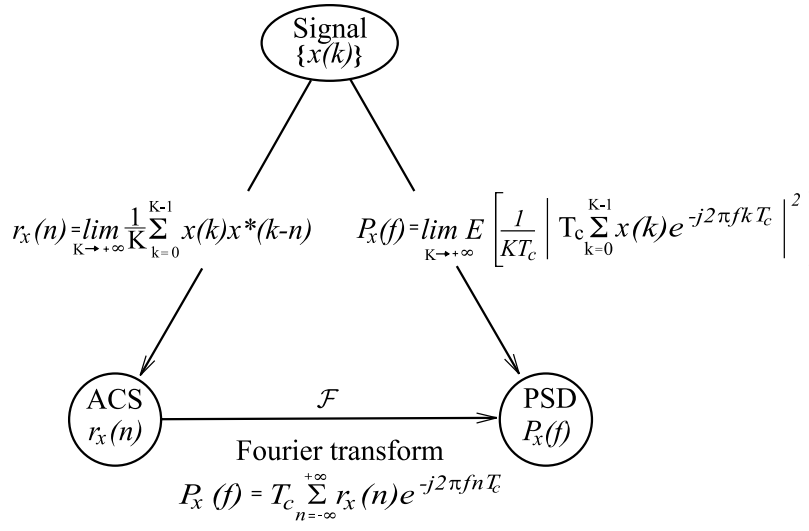


Figure 1.40. Relation between ergodic processes and their statistical description.

in Figure 1.40. We note how the direct computation of the PSD, given by (1.447), makes use of a statistical ensemble of the Fourier transform of process x, while the indirect method via ACS makes use of a single realization.

If we let

$$\tilde{\mathcal{X}}_{KT_c}(f) = T_c\, \mathcal{F}[x(k)\, \mathrm{w}_K(k)] \tag{1.448}$$

where w_K is the rectangular window of length K (see (1.474)) and $T_d = KT_c$, (1.447) becomes

$$\mathcal{P}_x(f) = \lim_{T_d\to\infty} \frac{E[|\tilde{\mathcal{X}}_{T_d}(f)|^2]}{T_d} \tag{1.449}$$

The relation (1.449) holds also for continuous-time ergodic random processes, where $\tilde{\mathcal{X}}_{T_d}(f)$ denotes the Fourier transform of the windowed realization of the process, with a rectangular window of duration T_d.

1.11.1 Mean value estimators

Given the random process $\{x(k)\}$, we wish to estimate the mean value of a related process $\{y(k)\}$: for example, to estimate the statistical power of x we set $y(k) = |x(k)|^2$, while for the estimation of the correlation of x with lag n we set $y(k) = x(k)x^*(k-n)$. Based on a realization of $\{y(k)\}$, from (1.442) an estimate of the mean value of y is given by the expression

$$\hat{\mathrm{m}}_y = \frac{1}{K} \sum_{k=0}^{K-1} y(k) \tag{1.450}$$

In fact, (1.450) attempts to determine the average component of the signal $\{y(k)\}$. As illustrated in Figure 1.41a, in general we can think of extracting the average component of $\{y(k)\}$ using an LPF filter h having unit gain, i.e. $\mathcal{H}(0) = 1$, and suitable bandwidth B. Let K be the length of the impulse response with support from $k = 0$ to $k = K - 1$. Note that for a unit step input signal the transient part of the output signal will last $K - 1$ time instants. Therefore we assume

$$\hat{\mathbb{m}}_y = z(k) = h * y(k) \qquad \text{for } k \geq K - 1 \tag{1.451}$$

We now compute mean and variance of the estimate. From (1.451), the mean value is given by

$$E[\hat{\mathbb{m}}_y] = \mathbb{m}_y \mathcal{H}(0) = \mathbb{m}_y \tag{1.452}$$

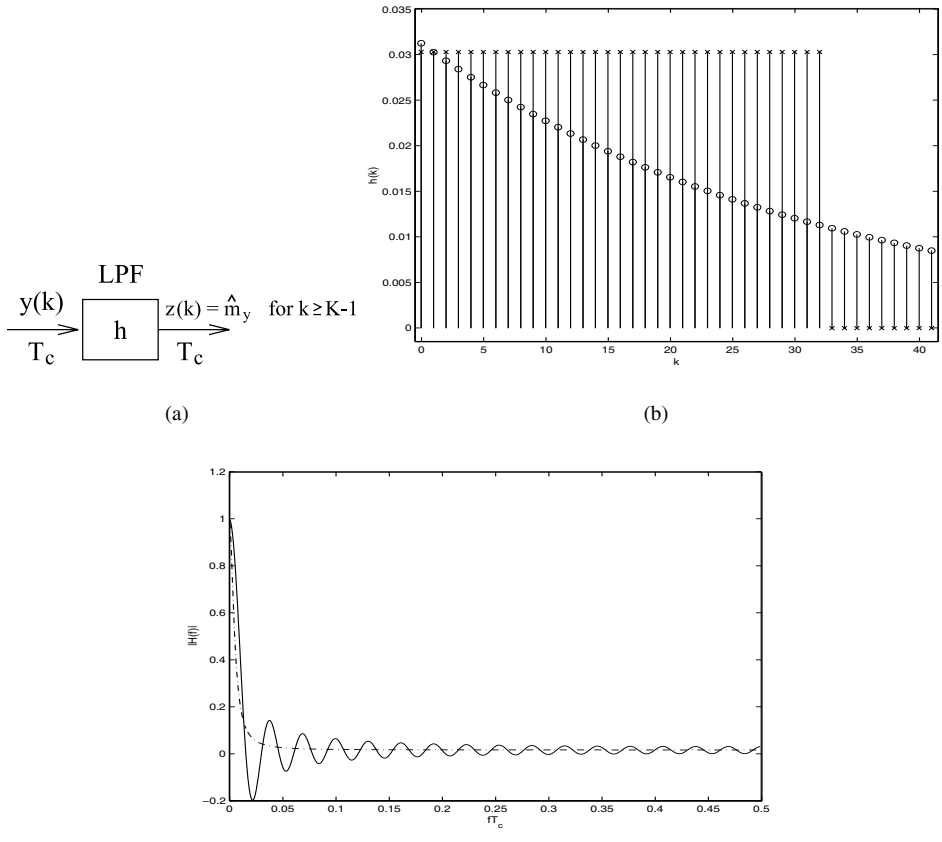


Figure 1.41. (a) Time average as output of a narrow band lowpass filter. (b) Typical impulse responses: exponential filter with parameter $a = 1 - 2^{-5}$ and rectangular window with $K = 33$. (c) Corresponding frequency responses.

as $\mathcal{H}(0) = 1$. Using the expression in Table 1.6 of the correlation of a filter output signal given the input, the variance of the estimate is given by

$$\text{var}[\hat{\mathtt{m}}_y] = \sigma_y^2 = \sum_{n=-\infty}^{+\infty} \mathtt{r}_h(-n)\mathtt{c}_y(n) \tag{1.453}$$

Assuming

$$S = \sum_{n=-\infty}^{+\infty} |\mathtt{c}_y(n)| = \sigma_y^2 \sum_{n=-\infty}^{+\infty} \frac{|\mathtt{c}_y(n)|}{\sigma_y^2} < \infty \tag{1.454}$$

and being $|\mathtt{r}_h(n)| \leq \mathtt{r}_h(0)$, the variance in (1.453) is *bounded* by

$$\text{var}[\hat{\mathtt{m}}_y] \leq E_h S \tag{1.455}$$

where $E_h = \mathtt{r}_h(0)$.

For an ideal lowpass filter

$$\mathcal{H}(f) = \text{rect}\left(\frac{f}{2B}\right) \qquad |f| < \frac{1}{2T_c} \tag{1.456}$$

assuming as filter length K that of the principal lobe of $\{h(k)\}$, and neglecting a delay factor, it results in $E_h = 2B$ and $K \simeq 1/B$. Introducing the criterion that for a good estimate it must be

$$\text{var}[\hat{\mathtt{m}}_y] \leq \varepsilon \tag{1.457}$$

with $\varepsilon \ll |\mathtt{m}_y|^2$, from (1.455) it follows

$$B \leq \frac{\varepsilon}{2S} \tag{1.458}$$

and

$$K \geq \frac{2S}{\varepsilon} \tag{1.459}$$

In other words, from (1.454) and (1.459), for a fixed ε, the length K of the filter impulse response must be larger, or equivalently the bandwidth B must be smaller, to obtain estimates for those processes $\{y(k)\}$ that exhibit larger variance and/or larger correlation among samples. Because of their simple implementation, two commonly used filters are the rectangular window and the exponential filter, whose impulse responses are shown in Figure 1.41.

Rectangular window

$$h(k) = \begin{cases} \dfrac{1}{K} & k = 0, 1, \ldots, K-1 \\ 0 & \text{elsewhere} \end{cases} \tag{1.460}$$

The frequency response is given by

$$\mathcal{H}(f) = \frac{1}{K} e^{-j2\pi f\left(\frac{K-1}{2}\right)T_c} \frac{\sin(\pi f K T_c)}{\sin(\pi f T_c)} \tag{1.461}$$

For the rectangular window we have $E_h = 1/K$ and, adopting as bandwidth the frequency of the first zero of $|\mathcal{H}(f)|$, $B = 1/(KT_c)$. The filter output is given by

$$z(k) = \sum_{n=0}^{K-1} \frac{1}{K} y(k-n) \tag{1.462}$$

that can be expressed as

$$z(k) = z(k-1) + \frac{y(k) - y(k-K)}{K} \tag{1.463}$$

Exponential filter

$$h(k) = \begin{cases} (1-a)a^k & k \geq 0 \\ 0 & \text{elsewhere} \end{cases} \tag{1.464}$$

with $|a| < 1$. The frequency response is given by

$$\mathcal{H}(f) = \frac{1-a}{1-ae^{-j2\pi f T_c}} \tag{1.465}$$

Moreover, $E_h = (1-a)/(1+a)$ and, adopting as length of h the time constant of the filter, i.e. the interval it takes for the amplitude of the impulse response to decrease of a factor e,

$$K - 1 = \frac{1}{\ln 1/a} \simeq \frac{1}{1-a} \tag{1.466}$$

where the approximation holds for $a \simeq 1$. The 3 dB filter bandwidth is equal to

$$B = \frac{1-a}{2\pi} \frac{1}{T_c} \qquad \text{for } a > 0.9 \tag{1.467}$$

The filter output has a simple expression given by the recursive equation

$$z(k) = az(k-1) + (1-a)\, y(k) \tag{1.468}$$

We note that choosing a as

$$a = 1 - 2^{-l} \tag{1.469}$$

the expression (1.468) becomes

$$z(k) = z(k-1) + 2^{-l}(y(k) - z(k-1)) \tag{1.470}$$

whose computation requires only two additions and one *shift* of l bits. Moreover, from (1.466), the filter time constant is given by

$$K - 1 = 2^l \tag{1.471}$$

General window

In addition to the two filters described above, a general window can be defined as

$$h(k) = A\mathrm{w}(k) \tag{1.472}$$

with $\{\mathrm{w}(k)\}$ window[20] of length K. The factor A in (1.472) is introduced to normalize the area of h to 1. We note that, for random processes with slowly time-varying statistics, the equations (1.463) and (1.470) give an expression to update the estimates.

1.11.2 Correlation estimators

Let $\{x(k)\}$, $k = 0, 1, \ldots, K-1$, be a realization of a random process with K samples. We examine two estimates.

Unbiased estimate

$$\hat{\mathrm{r}}_x(n) = \frac{1}{K-n} \sum_{k=n}^{K-1} x(k)x^*(k-n) \qquad n = 0, 1, \ldots, K-1 \tag{1.478}$$

---

[20] We define the *continuous-time rectangular window* with duration T_d as

$$\mathrm{w}_{T_d}(t) = \mathrm{rect}\left(\frac{t - T_d/2}{T_d}\right) = \begin{cases} 1 & 0 < t < T_d \\ 0 & \text{elsewhere} \end{cases} \tag{1.473}$$

Commonly used *discrete-time windows* are:

1. *Rectangular window*

$$\mathrm{w}(k) = \mathrm{w}_D(k) = \begin{cases} 1 & k = 0, 1, \ldots, D-1 \\ 0 & \text{elsewhere} \end{cases} \tag{1.474}$$

where D denotes the length of the rectangular window expressed in number of samples.

2. *Raised cosine or Hamming window*

$$\mathrm{w}(k) = \begin{cases} 0.54 + 0.46\cos\left(2\pi \dfrac{k - \dfrac{D-1}{2}}{D-1}\right) & k = 0, 1, \ldots, D-1 \\ 0 & \text{elsewhere} \end{cases} \tag{1.475}$$

3. *Hann window*

$$\mathrm{w}(k) = \begin{cases} 0.50 + 0.50\cos\left(2\pi \dfrac{k - \dfrac{D-1}{2}}{D-1}\right) & k = 0, 1, \ldots, D-1 \\ 0 & \text{elsewhere} \end{cases} \tag{1.476}$$

4. *Triangular or Bartlett window*

$$\mathrm{w}(k) = \begin{cases} 1 - 2\left|\dfrac{k - \dfrac{D-1}{2}}{D-1}\right| & k = 0, 1, \ldots, D-1 \\ 0 & \text{elsewhere} \end{cases} \tag{1.477}$$

The unbiased estimate has mean value equal to

$$E[\hat{\mathtt{r}}_x(n)] = \frac{1}{K-n}\sum_{k=n}^{K-1} E[x(k)x^*(k-n)] = \mathtt{r}_x(n) \tag{1.479}$$

If the process is Gaussian, one can show that the variance of the estimate is approximately given by

$$\mathrm{var}[\hat{\mathtt{r}}_x(n)] \simeq \frac{K}{(K-n)^2}\sum_{m=-\infty}^{+\infty} [\mathtt{r}_x^2(m) + \mathtt{r}_x(m+n)\mathtt{r}_x(m-n)] \tag{1.480}$$

from which it follows

$$\mathrm{var}[\hat{\mathtt{r}}_x(n)] \xrightarrow[K\to\infty]{} 0 \tag{1.481}$$

The above limit holds for $n \ll K$. Note that the variance of the estimate increases with the correlation lag n.

Biased estimate

$$\check{\mathtt{r}}_x(n) = \frac{1}{K}\sum_{k=n}^{K-1} x(k)x^*(k-n) = \left(1 - \frac{|n|}{K}\right)\hat{\mathtt{r}}_x(n) \tag{1.482}$$

The mean value of the biased estimate satisfies the following relations:

$$E[\check{\mathtt{r}}_x(n)] = \left(1 - \frac{|n|}{K}\right)\mathtt{r}_x(n) \xrightarrow[K\to\infty]{} \mathtt{r}_x(n) \tag{1.483}$$

Unlike the unbiased estimate, the mean of the biased estimate is not equal to the autocorrelation function, but approaches it as K increases. Note that the biased estimate differs from the autocorrelation function by one additive constant, denoted as *BIAS*:

$$BIAS = E[\check{\mathtt{r}}_x(n)] - \mathtt{r}_x(n) \tag{1.484}$$

For a Gaussian process, the variance of the biased estimate is expressed as

$$\mathrm{var}[\check{\mathtt{r}}_x(n)] = \left(\frac{K-|n|}{K}\right)^2 \mathrm{var}[\hat{\mathtt{r}}_x(n)] \simeq \frac{1}{K}\sum_{m=-\infty}^{+\infty} [\mathtt{r}_x^2(m) + \mathtt{r}_x(m+n)\mathtt{r}_x(m-n)] \tag{1.485}$$

In general, the biased estimate of the ACS exhibits a mean-square error[21] larger than the unbiased, especially for large values of n. It should also be noted that the estimate does not necessarily yield sequences that satisfy the properties of autocorrelation functions: for example, the following property may not be verified:

$$\hat{\mathtt{r}}_x(0) \geq |\hat{\mathtt{r}}_x(n)| \qquad n \neq 0 \tag{1.487}$$

[21] For example, for the estimator (1.478) the mean-square error is defined as

$$E[|\hat{\mathtt{r}}_x(n) - \mathtt{r}_x(n)|^2] = \mathrm{var}[\hat{\mathtt{r}}_x(n)] + |\mathrm{BIAS}|^2 \tag{1.486}$$

1.11.3 Power spectral density estimators

After examining ACS estimators, we review some spectral density estimation methods.

Periodogram or instantaneous spectrum

Let $\tilde{\mathcal{X}}(f) = T_c\mathcal{X}(f)$, where $\mathcal{X}(f)$ is the Fourier transform of $\{x(k)\}$, $k = 0, \ldots, K-1$; an estimate of the statistical power of $\{x(k)\}$ is given by

$$\begin{aligned}\hat{\mathtt{M}}_x &= \frac{1}{K}\sum_{k=0}^{K-1}|x(k)|^2 \\ &= \frac{1}{KT_c}\int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}}|\tilde{\mathcal{X}}(f)|^2\,df\end{aligned} \tag{1.488}$$

using the properties of the Fourier transform (Parseval theorem). Based on (1.488), a PSD estimator called a periodogram is given by

$$\mathcal{P}_{\mathrm{PER}}(f) = \frac{1}{KT_c}|\tilde{\mathcal{X}}(f)|^2 \tag{1.489}$$

We can write (1.489) as

$$\mathcal{P}_{\mathrm{PER}}(f) = T_c\sum_{n=-(K-1)}^{K-1}\check{\mathtt{r}}_x(n)\,e^{-j2\pi fnT_c} \tag{1.490}$$

and, consequently,

$$\begin{aligned}E[\mathcal{P}_{\mathrm{PER}}(f)] &= T_c\sum_{n=-(K-1)}^{K-1}E[\check{\mathtt{r}}_x(n)]e^{-j2\pi fnT_c} \\ &= T_c\sum_{n=-(K-1)}^{K-1}\left(1-\frac{|n|}{K}\right)\mathtt{r}_x(n)e^{-j2\pi fnT_c} \\ &= T_c\mathcal{W}_B * \mathcal{P}_x(f)\end{aligned} \tag{1.491}$$

where $\mathcal{W}_B(f)$ is the Fourier transform of the *Bartlett window*

$$\mathtt{w}_B(n) = \begin{cases}1-\dfrac{|n|}{K} & |n| \le K-1 \\ 0 & |n| > K-1\end{cases} \tag{1.492}$$

and

$$\mathcal{W}_B(f) = \frac{1}{K}\left[\frac{\sin(\pi fKT_c)}{\sin(\pi fT_c)}\right]^2 \tag{1.493}$$

We note the periodogram estimate is affected by *BIAS* for finite K. Moreover, it also exhibits a large variance, as $\mathcal{P}_{\mathrm{PER}}(f)$ is computed using the samples of $\check{\mathtt{r}}_x(n)$ even for lags up to $K-1$, whose variance is very large.

Welch periodogram

This method is based on applying (1.447) for finite K. Given a sequence of K samples, different subsequences of consecutive D samples are extracted. Subsequences may partially overlap. Let $x^{(s)}$ be the s-th subsequence, characterized by S samples in common with the preceding subsequence $x^{(s-1)}$ and with the following one $x^{(s+1)}$. In general, $0 \leq S \leq D/2$, with the choice $S = 0$ yielding subsequences with no overlap and therefore with less correlation. The number of subsequences N_s is[22]

$$N_s = \left\lfloor \frac{K - D}{D - S} + 1 \right\rfloor \tag{1.494}$$

Let $\mathtt{w}$ be a window (see footnote 20 on page 82) of D samples: then

$$x^{(s)}(k) = \mathtt{w}(k)\, x(k + s(D - S)) \qquad k = 0, 1, \ldots, D - 1 \qquad s = 0, 1, \ldots, N_s - 1 \tag{1.495}$$

For each s, compute the Fourier transform

$$\tilde{\mathcal{X}}^{(s)}(f) = T_c \sum_{k=0}^{D-1} x^{(s)}(k) e^{-j2\pi f k T_c} \tag{1.496}$$

and obtain

$$\mathcal{P}^{(s)}_{\mathrm{PER}}(f) = \frac{1}{D T_c \mathtt{M}_{\mathtt{w}}} \left| \tilde{\mathcal{X}}^{(s)}(f) \right|^2 \tag{1.497}$$

where

$$\mathtt{M}_{\mathtt{w}} = \frac{1}{D} \sum_{k=0}^{D-1} \mathtt{w}^2(k) \tag{1.498}$$

is the normalized energy of the window. As a last step, for each frequency, average the periodograms:

$$\mathcal{P}_{\mathrm{WE}}(f) = \frac{1}{N_s} \sum_{s=0}^{N_s-1} \mathcal{P}^{(s)}_{\mathrm{PER}}(f) \tag{1.499}$$

Mean and variance of the estimate are given by

$$E[\mathcal{P}_{\mathrm{WE}}(f)] = T_c[|\mathcal{W}(\lambda)|^2 * \mathcal{P}_x(\lambda)](f) \tag{1.500}$$

where

$$\mathcal{W}(f) = \sum_{k=0}^{D-1} \mathtt{w}(k) e^{-j2\pi f k T_c} \tag{1.501}$$

[22] The symbol $\lfloor a \rfloor$ denotes the function *floor*, that is the largest integer smaller than or equal to a. The symbol $\lceil a \rceil$ denotes the function *ceiling*, that is the smallest integer larger than or equal to a.

Assuming the process is Gaussian and the different subsequences are statistically independent, we get[23]

$$\mathrm{var}[\mathcal{P}_{\mathrm{WE}}(f)] \propto \frac{1}{N_s}\mathcal{P}_x^2(f) \tag{1.502}$$

Note that the partial overlap introduces correlation between subsequences. From (1.502), we see that the variance of the estimate is reduced by increasing the number of subsequences. In general, D must be large enough so that the generic subsequence represents the process and also N_s must be large to obtain a reliable estimate (see (1.502)); therefore the application of the Welch method requires many samples.

Blackman and Tukey correlogram

For an *unbiased* estimate of the ACS, $\{\hat{\mathfrak{r}}_x(n)\}$, $n = -L, \ldots, L$, consider the Fourier transform

$$\mathcal{P}_{\mathrm{BT}}(f) = T_c \sum_{n=-L}^{L} \mathtt{w}(n)\hat{\mathfrak{r}}_x(n)\, e^{-j2\pi f n T_c} \tag{1.503}$$

where $\mathtt{w}$ is a window[24] of length $2L+1$, with $\mathtt{w}(0) = 1$. If K is the number of samples of the realization sequence, we require that $L \leq K/5$ to reduce the variance of the estimate. Then if the Bartlett window (1.493) is chosen, one finds that $\mathcal{P}_{\mathrm{BT}}(f) \geq 0$.

In terms of the mean value of the estimate, we find

$$E[\mathcal{P}_{\mathrm{BT}}(f)] = T_c \mathcal{W}(f) * \mathcal{P}_x(f) \tag{1.504}$$

For a Gaussian process, if the Bartlett window is chosen, the variance of the estimate is given by

$$\mathrm{var}[\mathcal{P}_{\mathrm{BT}}(f)] = \frac{1}{K}\mathcal{P}_x^2(f) E_{\mathtt{w}} = \frac{2}{3}\frac{L}{K}\mathcal{P}_x^2(f) \tag{1.505}$$

Windowing and window closing

The operation of windowing time samples in the periodogram, and the autocorrelation sequence in the correlogram, has a strong effect on the performance of the estimate. In fact, any truncation of a sequence is equivalent to a windowing operation, carried out via the function "rect". The choice of the window type in the frequency domain depends on the compromise between a narrow central lobe (to reduce *smearing*) and a fast decay of secondary lobes (to reduce *leakage*). *Smearing* yields a lower spectral resolution, that is the capability to distinguish two spectral lines that are close. On the other hand, *leakage* can mask spectral components that are further apart and have different amplitudes.

---

[23] The symbol '$\propto$' indicates proportional.

[24] The windows used in (1.503) are the same as those introduced in note 20: the only difference is that they are now centered around zero instead of $(D-1)/2$. To simplify the notation, we will use the same symbol in both cases.

The choice of the window length is based on the compromise between spectral resolution and the variance of the estimate. An example has already been seen in the correlogram, where the condition $L \leq K/5$ must be satisfied. Another example is the Welch periodogram. For a given observation of K samples, it is initially better to choose a small number of samples over which to perform the DFT, and therefore a large number of windows (subsequences) over which to average the estimate. The estimate is then repeated by increasing the number of samples per window, thus decreasing the number of windows. In this way we get estimates with a higher resolution, but also characterized by an increasing variance. The procedure is terminated once it is found that the increase in variance is no longer compensated by an increase in the spectral resolution. The aforementioned method is called *window closing*.

Example 1.11.1
Consider a realization of $K = 10000$ samples of the signal:

$$y(kT_c) = \frac{1}{A_h} \sum_{n=-16}^{16} h(nT_c) w((k-n)T_c) + A_1 \cos(2\pi f_1 k T_c + \varphi_1) + A_2 \cos(2\pi f_2 k T_c + \varphi_2), \tag{1.506}$$

where $\varphi_1, \varphi_2 \in \mathcal{U}[0, 2\pi)$, $w(nT_c)$ is a white random process with zero mean and variance $\sigma_w^2 = 5$, $T_c = 0.2$, $A_1 = 1/20$, $f_1 = 1.5$, $A_2 = 1/40$, $f_2 = 1.75$, and

$$A_h = \sum_{-16}^{16} h(kT_c) \tag{1.507}$$

Moreover

$$h(kT_c) = \frac{\sin\left(\pi(1-\rho)\frac{kT_c}{T}\right) + 4\rho\frac{kT_c}{T}\cos\left(\pi(1+\rho)\frac{kT_c}{T}\right)}{\pi\left[1 - \left(4\rho\frac{kT_c}{T}\right)^2\right]\frac{kT_c}{T}} \operatorname{rect}\left(\frac{kT_c}{8T + T_c}\right) \tag{1.508}$$

with $T = 4T_c$ and $\rho = 0.32$.

Actually y is the sum of two sinusoidal signals and filtered white noise through h. Consequently, observing (1.264) and (1.388),

$$\mathcal{P}_y(f) = \sigma_w^2 T_c \frac{|\mathcal{H}(f)|^2}{A_h^2} + \frac{A_1^2}{4}(\delta(f - f_1) + \delta(f + f_1)) + \frac{A_2^2}{4}(\delta(f - f_2) + \delta(f + f_2)) \tag{1.509}$$

where $\mathcal{H}(f)$ is the Fourier transform of $\{h(kT_c)\}$.

The shape of the PSD in (1.509) is shown in Figures 1.42 to 1.44 as a solid line. A Dirac impulse is represented by an isosceles triangle having a base equal to twice the desired

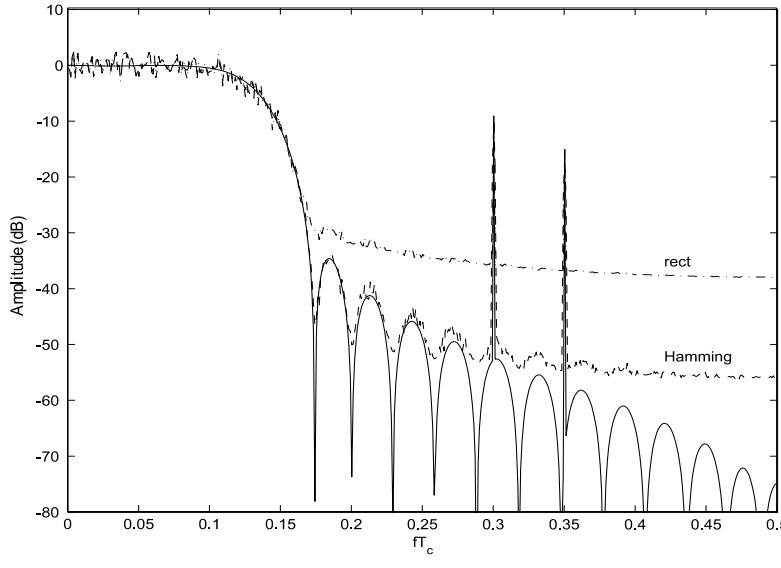


Figure 1.42. Comparison between spectral estimates obtained with Welch periodogram method, using the Hamming or the rectangular window, and the analytical PSD given by (1.509).

frequency resolution F_q. Consequently, a Dirac impulse, for example, of area $A_1^2/4$ will have a height equal to $A_1^2/(4F_q)$, thus maintaining the equivalence in statistical power between different representations.

We now compare several spectral estimates, obtained using the previously described methods; in particular we will emphasize the effect on the resolution of the type of window used and the number of samples for each window.

We state beforehand the following result. Windowing a complex sinusoidal signal $\{e^{j2\pi f_1 kT_c}\}$ with $\{\mathtt{w}(k)\}$ produces a signal having Fourier transform equal to $\mathcal{W}(f - f_1)$, where $\mathcal{W}(f)$ is the Fourier transform of $\mathtt{w}$. Therefore, in the frequency domain the spectral line of a sinusoidal signal becomes a signal with shape $\mathcal{W}(f)$ centered around f_1.

In general, from (1.497), the periodogram of a real sinusoidal signal with amplitude A_1 and frequency f_1 is

$$\mathcal{P}_{\mathrm{PER}}(f) = \frac{T_c}{D\mathtt{M}_\mathtt{w}} \left(\frac{A_1}{2}\right)^2 |\mathcal{W}(f - f_1) + \mathcal{W}(f + f_1)|^2 \tag{1.510}$$

Figure 1.42 shows, in addition to the analytical PSD (1.509), the estimate obtained by the Welch periodogram method using the Hamming or the rectangular windows. Parameters used in (1.496) and (1.499) are: $D = 1000$, $N_s = 19$ and 50% overlap between windows. We observe that the use of the Hamming window yields an improvement of the estimate due to less *leakage*. Likewise Figure 1.43 shows how the Hamming window also improves the estimate carried out with the correlogram; in particular, the estimates of Figure 1.43 were obtained using in (1.503) $L = 500$. Finally, Figure 1.44 shows how the resolution and

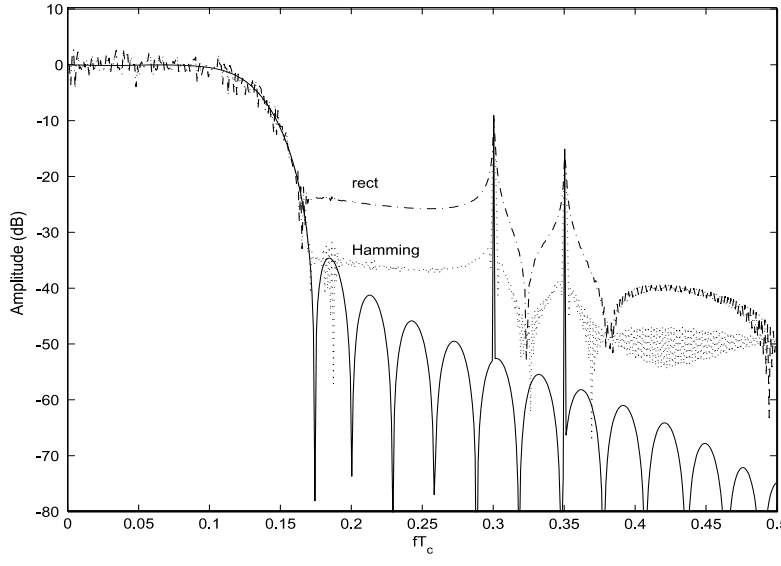


Figure 1.43. Comparison between spectral estimates obtained with the correlogram using the Hamming or the rectangular window, and the analytical PSD given by (1.509).

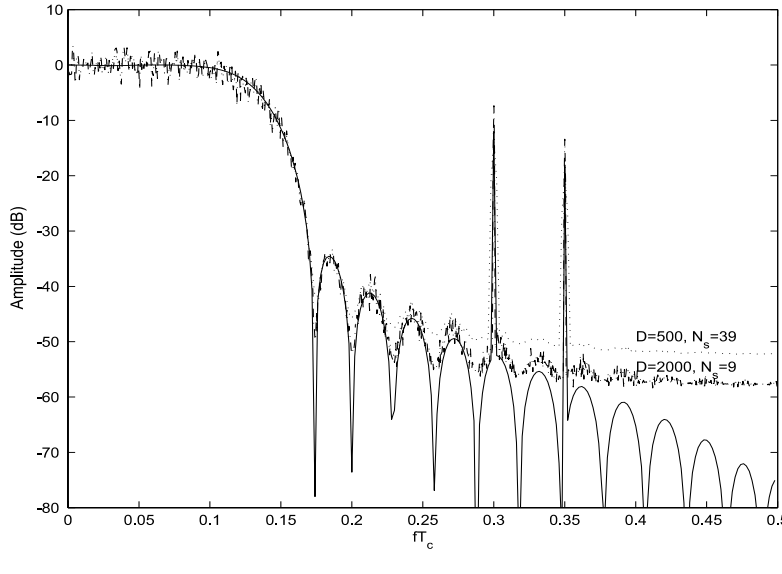


Figure 1.44. Comparison of spectral estimates obtained with the Welch periodogram method, using the Hamming window, by varying parameters D ed N_S.

the variance of the estimate obtained by the Welch periodogram vary with the parameters D and N_s, using the Hamming window. Note that by increasing D, and hence decreasing N_s, both resolution and variance of the estimate increase.

1.12 Parametric models of random processes

ARMA(p,q) model

Let us consider the realization of a random process x according to the *auto-regressive moving average* model illustrated in Figure 1.45. In other words, the process x, also called observed sequence, is the output of an IIR filter having as input white noise with variance σ_w^2, and is given by the recursive equation[25]

$$x(k) = -\sum_{n=1}^{p} \mathtt{a}_n x(k-n) + \sum_{n=0}^{q} \mathtt{b}_n w(k-n) \tag{1.511}$$

Rewriting (1.511) in terms of the input–output relation of the linear system, from (1.129) we find in general

$$x(k) = \sum_{n=0}^{+\infty} h_{\mathrm{ARMA}}(n) w(k-n) \tag{1.512}$$

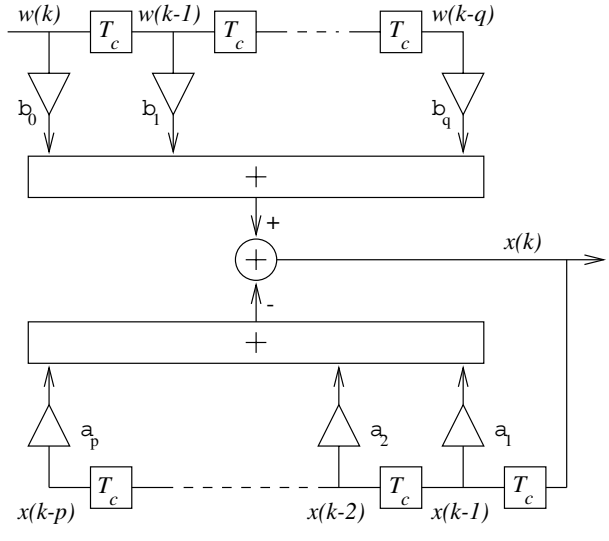


Figure 1.45. ARMA model of a process $x(k)$.

[25] In a simulation of the process, the first samples $x(k)$ generated by (1.511) should be ignored because they depend on the initial conditions.

which indicates that the filter used to realize the ARMA model is causal. From (1.129) one finds that the filter transfer function is given by

$$H_{\mathrm{ARMA}}(z) = \frac{B(z)}{A(z)} \qquad \text{where} \quad \begin{cases} B(z) = \sum_{n=0}^{q} \mathtt{b}_n z^{-n} \\ A(z) = \sum_{n=0}^{p} \mathtt{a}_n z^{-n} \qquad \text{assuming } \mathtt{a}_0 = 1 \end{cases} \tag{1.513}$$

Using (1.264), the power spectral density of the process x is given by:

$$\mathcal{P}_x(f) = T_c \sigma_w^2 \left| \frac{\mathcal{B}(f)}{\mathcal{A}(f)} \right|^2 \qquad \text{where} \quad \begin{cases} \mathcal{B}(f) = B(e^{j2\pi f T_c}) \\ \mathcal{A}(f) = A(e^{j2\pi f T_c}) \end{cases} \tag{1.514}$$

MA(q) model

If we particularize the ARMA model, assuming

$$\mathtt{a}_i = 0 \qquad i = 1, 2, \ldots, p \tag{1.515}$$

or $A(z) = 1$, we get the *moving average* model of order q. The equations of the ARMA model therefore are reduced to

$$H_{\mathrm{MA}}(z) = B(z) \tag{1.516}$$

and

$$\mathcal{P}_x(f) = T_c \, \sigma_w^2 |\mathcal{B}(f)|^2 \tag{1.517}$$

If we represent the function $\mathcal{P}_x(f)$ of a process obtained by the MA model, we see that its behavior is generally characterized by wide "peaks" and narrow "valleys", as illustrated in Figure 1.46.

AR(N) model

The *auto-regressive* model of order N is shown in Figure 1.47. The output process is described in this case by the recursive equation

$$x(k) = -\sum_{n=1}^{N} \mathtt{a}_n \, x(k-n) + w(k) \tag{1.518}$$

where w is white noise with variance σ_w^2. The transfer function is given by

$$H_{\mathrm{AR}}(z) = \frac{1}{A(z)} \tag{1.519}$$

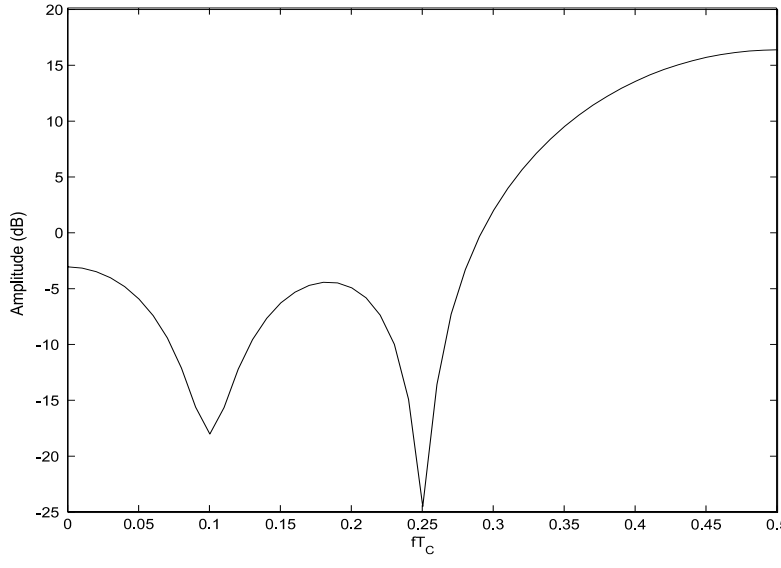


Figure 1.46. Power spectral density of a MA process with $q = 4$.

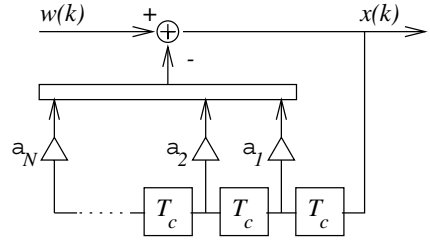


Figure 1.47. AR model of a process x.

with

$$A(z) = 1 + \sum_{n=1}^{N} \mathsf{a}_n z^{-n} \tag{1.520}$$

We observe that (1.519) describes a filter having N poles. Therefore $H_{\mathrm{AR}}(z)$ can be expressed as

$$H_{\mathrm{AR}}(z) = \frac{1}{(1 - \mathsf{p}_1 z^{-1})(1 - \mathsf{p}_2 z^{-1}) \cdots (1 - \mathsf{p}_N z^{-1})} \tag{1.521}$$

For a causal filter, the stability condition is $|\mathsf{p}_i| < 1$, $i = 1, 2, \ldots, N$, i.e. all poles must be inside the unit circle of the z plane.

In the case of the AR model, from Table 1.6 the z-transform of the ACS of x is given by

$$P_x(z) = P_w(z)\frac{1}{A(z)A^*\left(\frac{1}{z^*}\right)} = \frac{\sigma_w^2}{A(z)A^*\left(\frac{1}{z^*}\right)} \tag{1.522}$$

Hence the function $P_x(z)$ has poles of the type

$$|\mathsf{p}_i|e^{j\varphi_i} \quad \text{and} \quad \frac{1}{|\mathsf{p}_i|}e^{j\varphi_i} \tag{1.523}$$

Letting

$$\mathcal{A}(f) = A(e^{j2\pi f T_c}) \tag{1.524}$$

one obtains the power spectral density of x, given by

$$\mathcal{P}_x(f) = \frac{T_c\sigma_w^2}{|\mathcal{A}(f)|^2} \tag{1.525}$$

Typically, the function $\mathcal{P}_x(f)$ of an AR process will have narrow "peaks" and wide "valleys" (see Figure 1.48), reciprocal to the MA model.

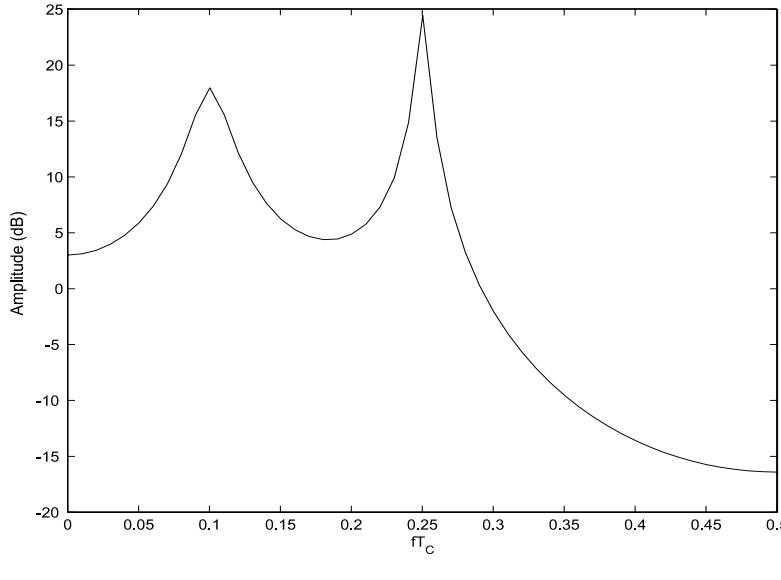


Figure 1.48. Power spectral density of an AR process with $N = 4$.

Spectral factorization of an AR(N) model

Consider the AR process described by (1.522). Observing (1.523), we have the following decomposition:

$$P_x(z) = \frac{\sigma_w^2}{(1 - |\mathsf{p}_1| e^{j\varphi_1} z^{-1}) \cdots (1 - |\mathsf{p}_N| e^{j\varphi_N} z^{-1})} \left(1 - \frac{1}{|\mathsf{p}_1|} e^{j\varphi_1} z^{-1}\right) \cdots \left(1 - \frac{1}{|\mathsf{p}_N|} e^{j\varphi_N} z^{-1}\right) \tag{1.526}$$

For a given $P_x(z)$, it is clear that the N zeros of $A(z)$ in (1.522) can be chosen in 2^N different ways. The selection of the zeros of $A(z)$ is called *spectral factorization*. Two examples are illustrated in Figure 1.49.

As stated by the spectral factorization theorem (see page 53) there exists a unique spectral factorization that yields a minimum-phase $A(z)$, which is obtained by associating with $A(z)$ only the poles of $P_x(z)$ that lie inside the unit circle of the z-plane.

Whitening filter

We observe an important property illustrated in Figure 1.50. Suppose x is modeled as an AR process of order N and has PSD given by (1.522). If x is input to a filter having transfer function $A(z)$, the output of this latter filter would be white noise. In this case the filter $A(z)$ is called *whitening filter* (WF).

If $A(z)$ is minimum phase, the white process w is also called *innovation* of the process x, in the sense that the new information associated with the sample $x(k)$ is carried only by $w(k)$.

Relation between ARMA, MA and AR models

The relations between the three parametric models are expressed through the following propositions.

Wold decomposition. Every WSS random process y can be decomposed into:

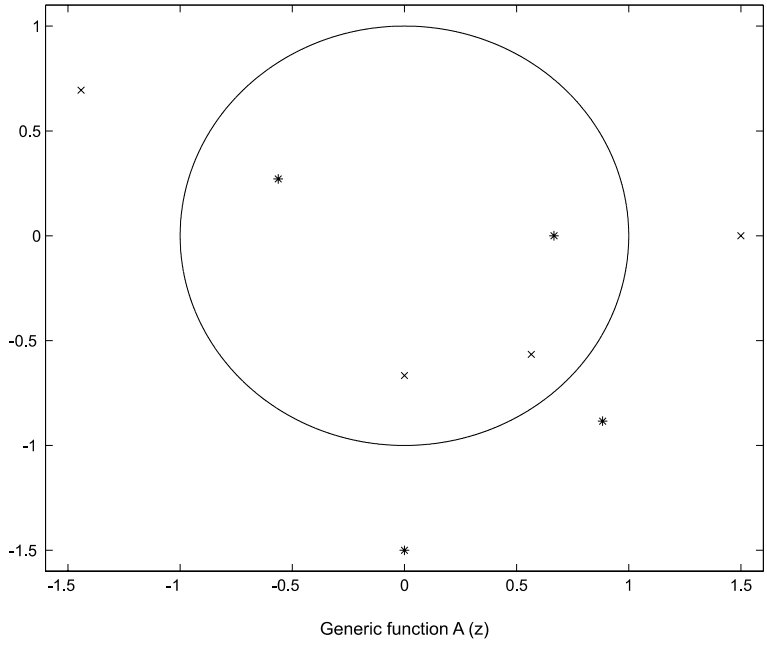
$$y(k) = s(k) + x(k) \tag{1.527}$$

where s and x are uncorrelated processes. The process s, called *predictable process*, is described by the recursive equation

$$s(k) = -\sum_{n=1}^{+\infty} \alpha_n s(k-n) \tag{1.528}$$

while x is obtained as filtered white noise:

$$x(k) = \sum_{n=0}^{+\infty} h(n) w(k-n) \tag{1.529}$$



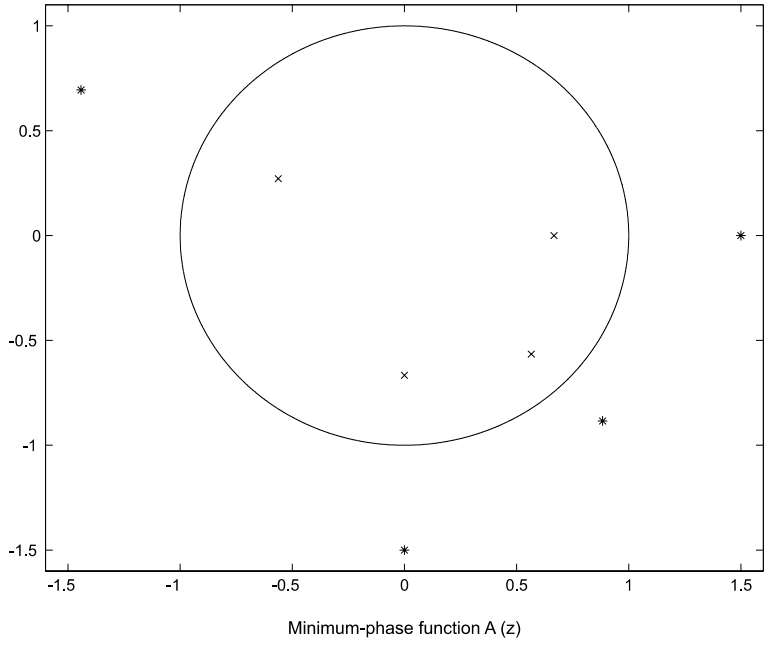


Figure 1.49. Two examples of possible choices of the zeros ($\times$) of $A(z)$, among the poles of $P_X(z)$.

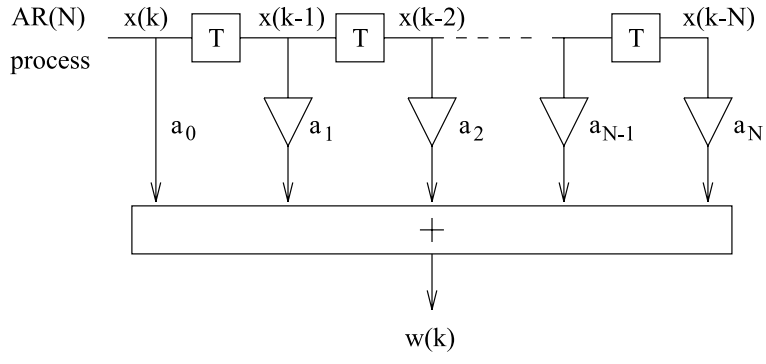


Figure 1.50. Whitening filter for an AR process of order *N*.

Theorem 1.4 (Kolmogorov theorem)
Any ARMA or MA process can be represented by an AR process of infinite order.

Therefore any one of the three descriptions (ARMA, MA, or AR) can be adopted to approximate the spectrum of a process, provided that the order is sufficiently high.

1.12.1 Autocorrelation of AR processes

It is interesting to evaluate the autocorrelation function of a process x obtained by the AR model. Multiplying both members of (1.518) by $x^*(k-n)$, we find

$$x(k)x^*(k-n) = -\sum_{m=1}^{N} \mathtt{a}_m x(k-m)x^*(k-n) + w(k)x^*(k-n) \tag{1.530}$$

Taking expectations, and observing that $w(k)$ is uncorrelated with all past values of x, for $n \geq 0$ one gets

$$E[x(k)x^*(k-n)] = -\sum_{m=1}^{N} \mathtt{a}_m \, E[x(k-m)\, x^*(k-n)] + \sigma_w^2 \delta_n \tag{1.531}$$

From (1.531), it follows

$$\mathtt{r}_x(n) = -\sum_{m=1}^{N} \mathtt{a}_m \mathtt{r}_x(n-m) + \sigma_w^2 \delta_n \tag{1.532}$$

In particular we have

$$\mathtt{r}_x(n) = \begin{cases} -\displaystyle\sum_{m=1}^{N} \mathtt{a}_m \mathtt{r}_x(n-m) & n > 0 \\ -\displaystyle\sum_{m=1}^{N} \mathtt{a}_m \mathtt{r}_x(-m) + \sigma_w^2 & n = 0 \\ \mathtt{r}_x^*(-n) & n < 0 \end{cases} \tag{1.533}$$

We observe that, for $n > 0$, $\mathtt{r}_x(n)$ satisfies an equation analogous to the (1.518), with the exception of the component $w(k)$. This implies that, if $\{p_i\}$ are zeros of $A(z)$, $\mathtt{r}_x$ can be written as

$$\mathtt{r}_x(n) = \sum_{i=1}^{N} \mathtt{c}_i \, p_i^n \qquad n > 0 \tag{1.534}$$

Assuming an AR process with $|p_i| < 1$, for $i = 1, 2, \ldots, N$, we get:

$$\mathtt{r}_x(n) \xrightarrow[n\to\infty]{} 0 \tag{1.535}$$

Simplifying notation $\mathtt{r}_x(n)$ with $\mathtt{r}(n)$, and observing (1.533), for $n = 1, 2, \ldots, N$, one gets a set of equations that in matrix notation are expressed as

$$\begin{bmatrix} \mathtt{r}(0) & \mathtt{r}(-1) & \cdots & \mathtt{r}(-N+1) \\ \mathtt{r}(1) & \mathtt{r}(0) & \cdots & \mathtt{r}(-N+2) \\ \vdots & \vdots & \ddots & \cdots \\ \mathtt{r}(N-1) & \mathtt{r}(N-2) & \cdots & \mathtt{r}(0) \end{bmatrix} \begin{bmatrix} \mathtt{a}_1 \\ \mathtt{a}_2 \\ \vdots \\ \mathtt{a}_N \end{bmatrix} = - \begin{bmatrix} \mathtt{r}(1) \\ \mathtt{r}(2) \\ \vdots \\ \mathtt{r}(N) \end{bmatrix} \tag{1.536}$$

that is

$$\mathbf{R}\mathbf{a} = -\mathbf{r} \tag{1.537}$$

with obvious definition of the vectors. In the hypothesis that the matrix $\mathbf{R}$ has an inverse, the solution for the coefficients $\{\mathtt{a}_i\}$ is given by

$$\mathbf{a} = -\mathbf{R}^{-1}\mathbf{r} \tag{1.538}$$

Equations (1.537) and (1.538), called *Yule–Walker equations*, allow us to obtain the coefficients of an AR model for a process having autocorrelation function $\mathtt{r}_x$. The variance σ_w^2 of white noise at the input can be obtained from (1.533) for $n = 0$, which yields

$$\begin{aligned} \sigma_w^2 &= \mathtt{r}_x(0) + \sum_{m=1}^{N} \mathtt{a}_m \mathtt{r}_x(-m) \\ &= \mathtt{r}_x(0) + \mathbf{r}^H \mathbf{a} \end{aligned} \tag{1.539}$$

Observation 1.8

- From (1.537) one finds that $\mathbf{a}$ does not depend on $\mathtt{r}_x(0)$, but only on the correlation coefficients

$$\rho_x(n) = \frac{\mathtt{r}_x(n)}{\mathtt{r}_x(0)} \qquad n = 1, \ldots, N \tag{1.540}$$

- Exploiting the fact that $\mathbf{R}$ is Toeplitz and Hermitian, the set of equations (1.538) and (1.539) can be numerically solved by the Levinson–Durbin or by Delsarte–Genin algorithms, with a computational complexity proportional to N^2 (see Sections 2.2.1 and 2.2.2).

- We note that the knowledge of $\mathtt{r}_x(0), \mathtt{r}_x(1), \ldots, \mathtt{r}_x(N)$ univocally determines the ACS of an AR(N) process; for $n > N$, from (1.533), we get

$$\mathtt{r}_x(n) = -\sum_{m=1}^{N} \mathtt{a}_m \mathtt{r}_x(n-m) \tag{1.541}$$

1.12.2 Spectral estimation of an AR(N) process

Assuming an AR(N) model for a process x, (1.538) yields the coefficient vector $\mathbf{a}$, which implies an estimate of the ACS up to lag N is available. From (1.525), we define as spectral estimate

$$\mathcal{P}_{\mathrm{AR}}(f) = \frac{T_c \sigma_w^2}{|\mathcal{A}(f)|^2} \tag{1.542}$$

Usually the estimate (1.542) allows a better resolution than estimates obtained by other methods, such as $\mathcal{P}_{\mathrm{BT}}(f)$, because it does not show the effects of ACS truncation. In fact the AR model yields

$$\mathcal{P}_{\mathrm{AR}}(f) = T_c \sum_{n=-\infty}^{+\infty} \hat{\mathfrak{r}}_x(n)\, e^{-j2\pi f n T_c} \tag{1.543}$$

where $\hat{\mathfrak{r}}_x(n)$ is estimated for $|n| \leq N$ with one of the two methods of Section 1.11.2, while for $|n| > N$ the recursive equation (1.541) is used. The AR model accurately estimates processes with a spectrum similar to that given in Figure 1.48. For example, a spectral estimate for the process of Example 1.11.1 on page 87 obtained by an AR(12) model is depicted in Figure 1.51. The correlation coefficients were obtained by a *biased* estimate on 10000 samples. Note that the continuous part of the spectrum is estimated only approximately; on the other hand, the choice of a larger order N would result in an estimate with larger variance.

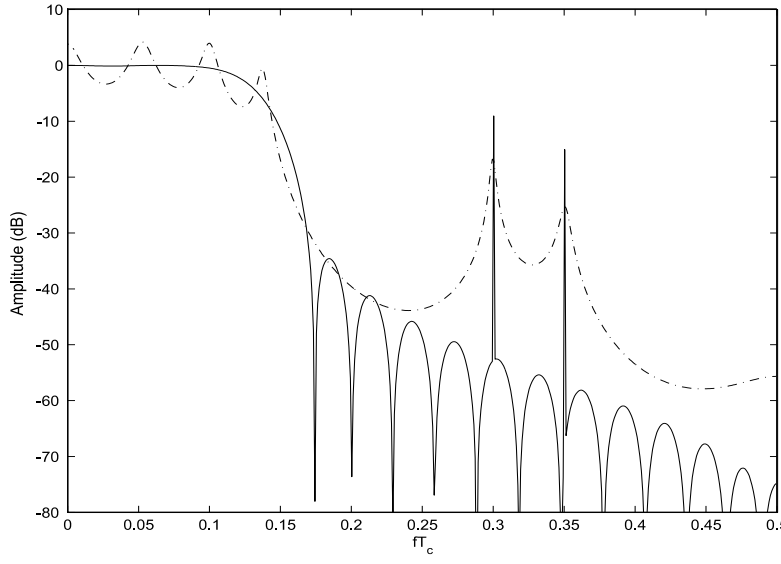


Figure 1.51. Comparison between the spectral estimate obtained by an AR(12) process model and the analytical PSD given by (1.509).

Also note that the presence of spectral lines in the original process leads to zeros of the polynomial $A(z)$ near the unit circle (see page 101). In practice, the correlation estimation method and a choice of a large N may result in an ill-conditioned matrix $\mathbf{R}$. In this case the solution may have poles outside the unit circle, and hence the system would be unstable.

Some useful relations

We will illustrate some examples of AR models. In particular we will focus on the Yule–Walker equations and the relation (1.539) for $N = 1$ and $N = 2$.

AR(1). From

$$\begin{cases} \mathtt{r}_x(n) = -\mathtt{a}_1 \mathtt{r}_x(n-1) & n > 0 \\ \sigma_w^2 = \mathtt{r}_x(0) + \mathtt{a}_1 \mathtt{r}_x(-1) \end{cases} \tag{1.544}$$

we obtain

$$\mathtt{r}_{\mathrm{AR}(1)}(n) = \frac{\sigma_w^2}{1 - |\mathtt{a}_1|^2}(-\mathtt{a}_1)^{|n|} \tag{1.545}$$

from which the spectral density is

$$\mathcal{P}_{\mathrm{AR}(1)}(f) = \frac{T_c \sigma_w^2}{|1 + \mathtt{a}_1 e^{-j2\pi f T_c}|^2} \tag{1.546}$$

The behavior of the spectral density of an AR(1) process is illustrated in Figure 1.52.

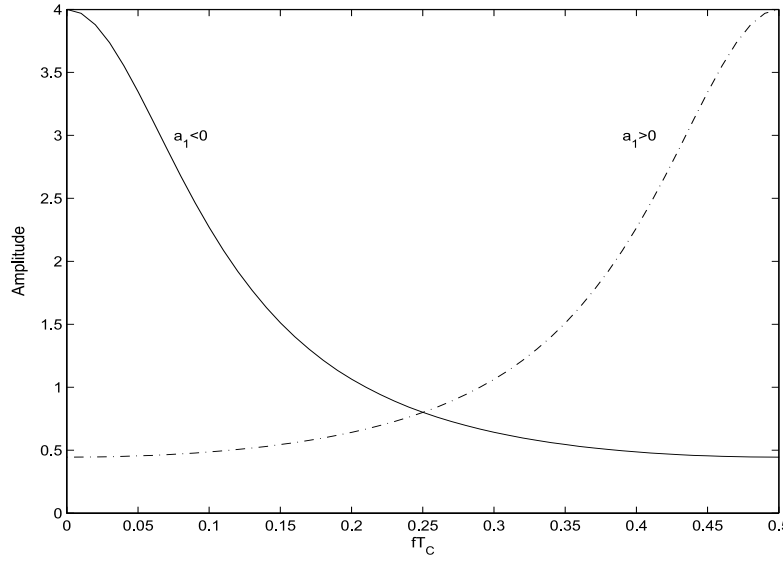


Figure 1.52. Spectral density of an AR(1) process.

AR(2). Let $p_{1,2} = \varrho e^{\pm j\varphi_0}$, where $\varphi_0 = 2\pi f_0 T_c$, be the two complex roots of $A(z) = 1 + \mathtt{a}_1 z^{-1} + \mathtt{a}_2 z^{-2}$. We consider a *real process*:

$$\begin{cases} \mathtt{a}_1 = -2\varrho\cos(2\pi f_0 T_c) \\ \mathtt{a}_2 = \varrho^2 \end{cases} \tag{1.547}$$

Letting

$$\vartheta = \tan^{-1}\left[\frac{1-\varrho^2}{1+\varrho^2}\tan^{-1}(2\pi f_0 T_c)\right] \tag{1.548}$$

we find

$$\mathtt{r}_{\mathrm{AR}(2)}(n) = \sigma_w^2 \frac{\dfrac{1+\varrho^2}{1-\varrho^2}\sqrt{1+\left(\dfrac{1-\varrho^2}{1+\varrho^2}\right)^2\left[\tan^{-1}(2\pi f_0 T_c)\right]^2}}{1-\varrho^2\cos^2(4\pi f_0 T_c)+\varrho^4}\varrho^{|n|}\cos(2\pi f_0|n|T_c - \vartheta) \tag{1.549}$$

The spectral density is thus given by

$$\mathcal{P}_{\mathrm{AR}(2)}(f) = \frac{T_c\sigma_w^2}{\left|1-\varrho e^{-j2\pi(f-f_0)T_c}\right|^2\left|1-\varrho e^{-j2\pi(f+f_0)T_c}\right|^2} \tag{1.550}$$

We observe that, as $\varrho \to 1$, $\mathcal{P}_{\mathrm{AR}(2)}(f)$ has a peak that becomes more pronounced, as illustrated in Figure 1.53, and $\mathtt{r}_x(k)$ tends to exhibit a sinusoidal behavior.

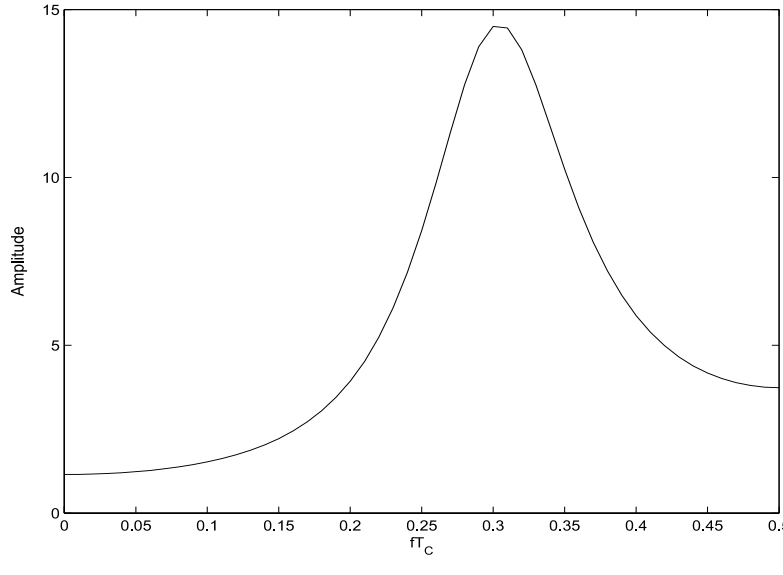


Figure 1.53. Spectral density of an AR(2) process.

Solutions of the Yule–Walker equations are

$$\begin{cases} \mathtt{a}_1 &= -\dfrac{\mathtt{r}_x(1)\mathtt{r}_x(0) - \mathtt{r}_x(1)\mathtt{r}_x(2)}{\mathtt{r}_x^2(0) - \mathtt{r}_x^2(1)} \\ \mathtt{a}_2 &= -\dfrac{\mathtt{r}_x(0)\mathtt{r}_x(2) - \mathtt{r}_x^2(1)}{\mathtt{r}_x^2(0) - \mathtt{r}_x^2(1)} \\ \sigma_w^2 &= \mathtt{r}_x(0) + \mathtt{a}_1\mathtt{r}_x(1) + \mathtt{a}_2\mathtt{r}_x(2) \end{cases} \tag{1.551}$$

Solving the previous set of equations with respect to $\mathtt{r}_x(0)$, $\mathtt{r}_x(1)$ and $\mathtt{r}_x(2)$, one obtains

$$\begin{cases} \mathtt{r}_x(0) &= \dfrac{1+\mathtt{a}_2}{1-\mathtt{a}_2}\dfrac{\sigma_w^2}{(1+\mathtt{a}_2)^2 - \mathtt{a}_1^2} \\ \mathtt{r}_x(1) &= -\dfrac{\mathtt{a}_1}{1+\mathtt{a}_2}\mathtt{r}_x(0) \\ \mathtt{r}_x(2) &= \left(-\mathtt{a}_2 + \dfrac{\mathtt{a}_1^2}{1+\mathtt{a}_2}\right)\mathtt{r}_x(0) \end{cases} \tag{1.552}$$

In general, for $n > 0$, we have

$$\mathtt{r}_x(n) = \mathtt{r}_x(0)\left[\frac{p_1(p_2^2-1)}{(p_2-p_1)(p_1p_2+1)}p_1^n - \frac{p_2(p_1^2-1)}{(p_2-p_1)(p_1p_2+1)}p_2^n\right] \tag{1.553}$$

AR model of sinusoidal processes

The general formulation of a sinusoidal process is:

$$x(k) = A\cos(2\pi f_0 k T_c + \varphi) \tag{1.554}$$

with $\varphi \in \mathcal{U}[0, 2\pi)$. We observe that the process described by (1.554) satisfies the following difference equation for $k \geq 0$:

$$x(k) = 2\cos(2\pi f_0 T_c)x(k-1) - x(k-2) + \delta_k A\cos\varphi - \delta_{k-1}A\cos(2\pi f_0 T_c - \varphi) \tag{1.555}$$

with $x(-2) = x(-1) = 0$. We note that the Kronecker impulses determine only the amplitude and phase of x.

In the z-domain, we get the homogeneous equation

$$A(z) = 1 - 2\cos(2\pi f_0 T_c)z^{-1} + z^{-2} \tag{1.556}$$

The zeros of $A(z)$ are

$$p_{1,2} = e^{\pm j2\pi f_0 T_c} \tag{1.557}$$

It is important to verify that these zeros belong to the unit circle of the z plane. Consequently the representation of a sinusoidal process via the AR model is not possible, as the stability

condition, $|p_i| < 1$, is not satisfied. Moreover the input (1.555) is not white noise. In any case, we can try to find an approximation. In the hypothesis of uniform φ, from Example 1.9.5,

$$\mathtt{r}_x(n) = \frac{A^2}{2}\cos(2\pi f_0 n T_c) \tag{1.558}$$

This autocorrelation function can be approximated by the autocorrelation of an AR(2) process for $\varrho \to 1$ and $\sigma_w^2 \to 0$. Using the formula (1.549), for $\varrho \simeq 1$ we find

$$\mathtt{r}_{\mathrm{AR}(2)}(n) \simeq \left[\frac{\sigma_w^2 \dfrac{2}{1-\varrho^2}}{2-\varrho^2\cos^2(4\pi f_0 T_c)}\right]\cos(2\pi f_0 n T_c) \tag{1.559}$$

and impose the condition

$$\lim_{\varrho\to 1,\sigma_w^2\to 0} \frac{\sigma_w^2 \dfrac{2}{1-\varrho^2}}{2-\varrho^2\cos^2(4\pi f_0 T_c)} = \frac{A^2}{2} \tag{1.560}$$

Observation 1.9
We can observe the following facts about the order of an AR model approximating a sinusoidal process.

- From (1.390) one finds that an AR process of order N is required to model N complex sinusoids; on the other hand, from (1.388), one sees that an AR process of order $2N$ is required to model N real sinusoids.

- An ARMA process of order $(2N, 2N)$ is required to model N real sinusoids plus white noise having variance σ_b^2. Observing (1.513), it results $\sigma_w^2 \to \sigma_b^2$ and $B(z) \to A(z)$.

A better estimate is obtained by separating the continuous part from the spectral lines, for example by the scheme illustrated in Figure 3.38. The two components are then estimated by different methods.

1.13 Guide to the bibliography

Many of the topics surveyed in this chapter are treated in general in several texts on digital communications, in particular [4, 5, 6].

In-depth studies on deterministic systems and signals are found in [3, 7, 8, 9, 10, 11]. For a statistical analysis of random processes we refer to [1, 12, 13]. Finally, the subject of spectral estimation is discussed in detail in [2, 14, 15, 16].

Bibliography

[1] A. Papoulis, *Probability, random variables and stochastic processes*. New York: McGraw-Hill, 3rd ed., 1991.

[2] M. B. Priestley, *Spectral analysis and time series*. New York, NY: Academic Press, 1981.

[3] A. Papoulis, *Signal analysis*. New York: McGraw-Hill, 1984.

[4] S. Benedetto and E. Biglieri, *Principles of digital transmission with wireless applications*. New York: Kluwer Academic Publishers, 1999.

[5] D. G. Messerschmitt and E. A. Lee, *Digital communication*. Boston, MA: Kluwer Academic Publishers, 2nd ed., 1994.

[6] J. G. Proakis, *Digital communications*. New York: McGraw-Hill, 3rd ed., 1995.

[7] G. Cariolaro, *La teoria unificata dei segnali*. Turin: UTET, 1996.

[8] A. Papoulis, *The Fourier integral and its applications*. New York: McGraw-Hill, 1962.

[9] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[10] P. P. Vaidyanathan, *Multirate systems and filter banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[11] R. E. Crochiere and L. R. Rabiner, *Multirate digital signal processing*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[12] J. W. B. Davenport and W. L. Root, *An introduction to the theory of random signals and noise*. New York: IEEE Press, 1987.

[13] A. N. Shiryayev, *Probability*. New York: Springer–Verlang, 1984.

[14] S. M. Kay, *Modern spectral estimation-theory and applications*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[15] L. S. Marple Jr., *Digital spectral analysis with applications*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[16] P. Stoica and R. Moses, *Introduction to spectral analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1997.

[17] L. Erup, F. M. Gardner, and R. A. Harris, "Interpolation in digital modems—Part II: implementation and performance", *IEEE Trans. on Communications*, vol. 41, pp. 998–1008, June 1993.

[18] H. Meyr, M. Moeneclaey, and S. A. Fechtel, *Digital communication receivers*. New York: John Wiley & Sons, 1998.

Appendix 1.A *Multirate systems*

The first part of this appendix is a synthesis from [10, 11].

1.A.1 Fundamentals

We consider the discrete-time linear transformation of Figure 1.54, with impulse response $h(t)$, $t \in \Re$; the sampling period of the input signal is T_c, whereas that of the output signal is T_c'.

The input–output relation is given by the equation

$$y(kT_c') = \sum_{n=-\infty}^{+\infty} h(kT_c' - nT_c)x(nT_c) \tag{1.561}$$

We will use the following simplified notation:

$$x_n = x(nT_c) \tag{1.562}$$

$$y_k = y(kT_c') \tag{1.563}$$

If we assume that h has a finite support, say, between t_1 and t_2, that is $h(kT_c' - nT_c) \neq 0$ for

$$kT_c' - nT_c < t_2 \qquad kT_c' - nT_c > t_1 \tag{1.564}$$

or equivalently for

$$n > \frac{kT_c' - t_2}{T_c} \qquad n < \frac{kT_c' - t_1}{T_c} \tag{1.565}$$

then, letting

$$n_1 = \left\lceil \frac{kT_c' - t_2}{T_c} \right\rceil \tag{1.566}$$

$$n_2 = \left\lfloor \frac{kT_c' - t_1}{T_c} \right\rfloor \tag{1.567}$$

(1.561) can be written as

$$y_k = \sum_{n=n_1}^{n_2} h(kT_c' - nT_c)x_n = x_{n_1}h(kT_c' - n_1T_c) + \cdots + x_{n_2}h(kT_c' - n_2T_c) \tag{1.568}$$

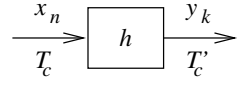


Figure 1.54. Discrete-time linear transformation.

One observes from (1.561) that:

- the values of h that contribute to y_k are equally spaced by T_c;
- the limits of the summation (1.568) are a complicated function of T_c, T_c', t_1, and t_2.

Introducing the change of variable

$$i = \left\lfloor \frac{kT_c'}{T_c} \right\rfloor - n \tag{1.569}$$

and setting

$$\Delta_k = \frac{kT_c'}{T_c} - \left\lfloor \frac{kT_c'}{T_c} \right\rfloor \tag{1.570}$$

$$I_1 = \left\lceil \frac{t_1}{T_c} - \Delta_k \right\rceil = \left\lceil \frac{t_1}{T_c} - \frac{kT_c'}{T_c} \right\rceil + \left\lfloor \frac{kT_c'}{T_c} \right\rfloor \tag{1.571}$$

$$I_2 = \left\lceil \frac{t_2}{T_c} - \Delta_k \right\rceil = \left\lceil \frac{t_2}{T_c} - \frac{kT_c'}{T_c} \right\rceil + \left\lfloor \frac{kT_c'}{T_c} \right\rfloor \tag{1.572}$$

(1.568) becomes

$$y_k = \sum_{i=I_1}^{I_2} h((i + \Delta_k)T_c) x_{\lfloor kT_c'/T_c \rfloor - i} \tag{1.573}$$

From the definition (1.570) it is clear that Δ_k represents the truncation error of kT_c'/T_c and that $0 \leq \Delta_k < 1$. In the special case

$$\frac{T_c'}{T_c} = \frac{M}{L} \tag{1.574}$$

with M and L integers, we get

$$\begin{aligned} \Delta_k &= k\frac{M}{L} - \left\lfloor k\frac{M}{L} \right\rfloor \\ &= \frac{1}{L}\left(kM - \left\lfloor k\frac{M}{L} \right\rfloor L\right) \\ &= \frac{1}{L}(kM)_{mod\, L} \end{aligned} \tag{1.575}$$

We observe that Δ_k can assume L values $\{0, 1/L, 2/L, \ldots, (L-1)/L\}$ for any value of k. Hence there are only L univocally determined sets of values of h that are used in the computation of $\{y_k\}$; in particular, if $L = 1$ only one set of coefficients exists, while

if $M = 1$ the sets are L. Summarizing, the output of a filter with impulse response h and with different input and output time domains can be expressed as

$$y_k = \sum_{i=-\infty}^{+\infty} g_{k,i} x_{\left\lfloor \frac{kM}{L} \right\rfloor - i} \tag{1.576}$$

where

$$g_{k,i} = h((i + \Delta_k)T_c) \tag{1.577}$$

We note that the system is *linear* and *periodically time-varying*. For $T_c' = T_c$, that is for $L = M = 1$, we get $\Delta_k = 0$, and the input–output relation is the usual convolution

$$y_k = \sum_{i=-\infty}^{+\infty} g_{0,i} x_{k-i} \tag{1.578}$$

We will now analyze a few elementary multirate transformations.

1.A.2 Decimation

Figure 1.55 represents a decimator or downsampler, with the output sequence related to the input sequence $\{x_n\}$ by

$$y_k = x_{kM} \tag{1.579}$$

where M, the *decimation factor*, is an integer number.

We now obtain an expression for the z-transform of the output $Y(z)$ in terms of $X(z)$. We will show that

$$Y(z) = \frac{1}{M} \sum_{m=0}^{M-1} X(z^{\frac{1}{M}} W_M^m) \tag{1.580}$$

where $W_M = e^{-j\frac{2\pi}{M}}$ is defined in (1.92). Equivalently, in terms of the radian frequency normalized by the sampling frequency, $\omega' = 2\pi f / F_c'$, (1.580) can be written as

$$Y(e^{j\omega'}) = \frac{1}{M} \sum_{m=0}^{M-1} X\left(e^{j\frac{\omega' - 2\pi m}{M}}\right) \tag{1.581}$$

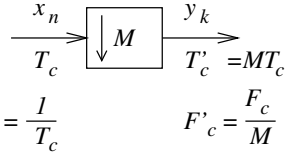
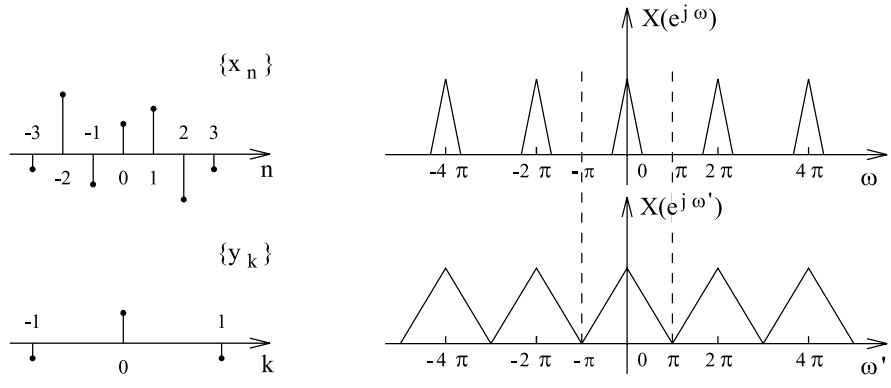


Figure 1.55. Decimation or downsampling transformation by a factor M.



(a) Time domain (b) Frequency domain

Figure 1.56. Decimation by a factor $M = 3$: a) in the time domain, and b) in the normalized radian frequency domain.

A graphical interpretation of (1.581) is shown in Figure 1.56:

- expand $X(e^{j\omega})$ by a factor M, obtaining $X(e^{j\omega'/M})$;
- create $M - 1$ replicas of the expanded version, and frequency-shift them uniformly with increments of 2π for each replica;
- sum all the replicas and divide the result by M.

We observe that, after summation, the result is periodic in ω' with period 2π, as we would expect from a discrete Fourier transform.

It is also useful to give the expression of the output sequence in the frequency domain; we get

$$\mathcal{Y}(f) = \frac{1}{M} \sum_{m=0}^{M-1} \mathcal{X}\left(f - \frac{m}{MT_c}\right) \tag{1.582}$$

where

$$\mathcal{X}(f) = X(e^{j2\pi f T_c}) \tag{1.583}$$

$$\mathcal{Y}(f) = Y(e^{j2\pi f M T_c}) \tag{1.584}$$

The relation (1.582) for the signal of Figure 1.56 is represented in Figure 1.57. Note that the only difference with respect to the previous representation is that all frequency responses are now functions of the frequency f.

Proof of (1.580). The z-transform of $\{y_k\}$ can be written as

$$Y(z) = \sum_{k=-\infty}^{+\infty} y_k z^{-k} = \sum_{k=-\infty}^{+\infty} x_{Mk} z^{-k} \tag{1.585}$$

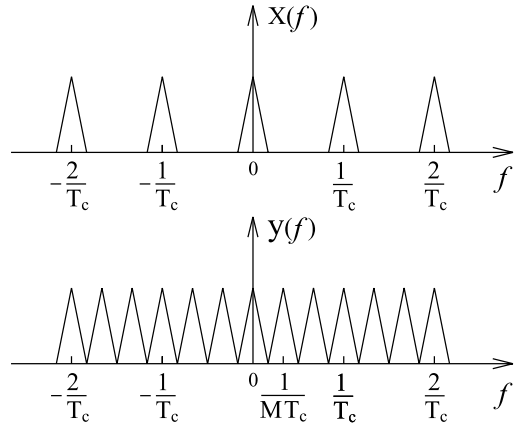


Figure 1.57. Effect of decimation in the frequency domain.

We define the intermediate sequence

$$x'_k = \begin{cases} x_k & k = 0, \pm M, \pm 2M, \dots \\ 0 & \text{otherwise} \end{cases} \tag{1.586}$$

so that $y_k = x_{Mk} = x'_{Mk}$. With this position we get

$$Y(z) = \sum_{k'=-\infty}^{+\infty} x'_{k'M} z^{-k'} = \sum_{k=-\infty}^{+\infty} x'_k z^{-k/M} = X'(z^{1/M}) \tag{1.587}$$

This relation is valid, because x' is non-zero only at multiples of M. It only remains to express $X'(z)$ in terms of $X(z)$; to do this, we note that (1.586) can be expressed as

$$x'_k = c_k x_k \tag{1.588}$$

where c_k is defined as:

$$c_k = \begin{cases} 1 & k = 0, \pm M, \pm 2M, \dots \\ 0 & \text{otherwise} \end{cases} \tag{1.589}$$

Note that the (1.589) can be written as

$$c_k = \frac{1}{M} \sum_{m=0}^{M-1} W_M^{-km} \tag{1.590}$$

Hence we obtain

$$X'(z) = \frac{1}{M} \sum_{m=0}^{M-1} \sum_{k=-\infty}^{+\infty} x_k W_M^{-km} z^{-k} = \frac{1}{M} \sum_{m=0}^{M-1} \sum_{k=-\infty}^{+\infty} x_k \left(z W_M^m\right)^{-k} \tag{1.591}$$

The inner summation yields $X(zW_M^m)$: hence, observing (1.587) we get (1.580).

1.A.3 Interpolation

Figure 1.58 represents an interpolator or upsampler, with the input sequence $\{x_n\}$ related to the output sequence by

$$y_k = \begin{cases} x\left(\dfrac{k}{L}\right) & k = 0, \pm L, \pm 2L, \ldots \\ 0 & \text{otherwise} \end{cases} \tag{1.592}$$

where L, the *interpolation factor*, is an integer number.

We will show that the input–output relation in terms of the z-transforms $Y(z)$ and $X(z)$ is given by

$$Y(z) = X(z^L) \tag{1.593}$$

Equivalently, in terms of radian frequency normalized by the sampling frequency, $\omega' = 2\pi f / F_c'$, then (1.593) can be expressed as

$$Y(e^{j\omega'}) = X(e^{j\omega' L}) \tag{1.594}$$

The graphical interpretation of (1.594) is illustrated in Figure 1.59: $Y(e^{j\omega})$ is the compressed version by a factor L of $X(e^{j\omega})$; moreover, there are $L - 1$ replicas of the compressed spectrum, called *images*. The creation of images implies that a lowpass signal does not remain lowpass after interpolation.

It is also useful to give the expression of the output sequence in the frequency domain; we get

$$\mathcal{Y}(f) = \mathcal{X}(f) \tag{1.595}$$

where

$$\mathcal{X}(f) = X(e^{j2\pi f T_c}) \tag{1.596}$$

$$\mathcal{Y}(f) = Y\left(e^{j2\pi f \frac{T_c}{L}}\right) \tag{1.597}$$

The (1.595) for the signal of Figure 1.59 is illustrated in Figure 1.60. We note that the only effect of the interpolation is that the signal $\mathcal{X}$ must be regarded as periodic with period F_c' rather than F_c.

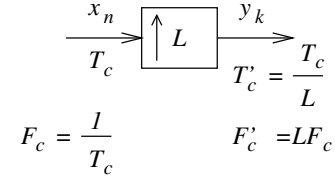


Figure 1.58. Interpolation or upsampling transformation by a factor L.

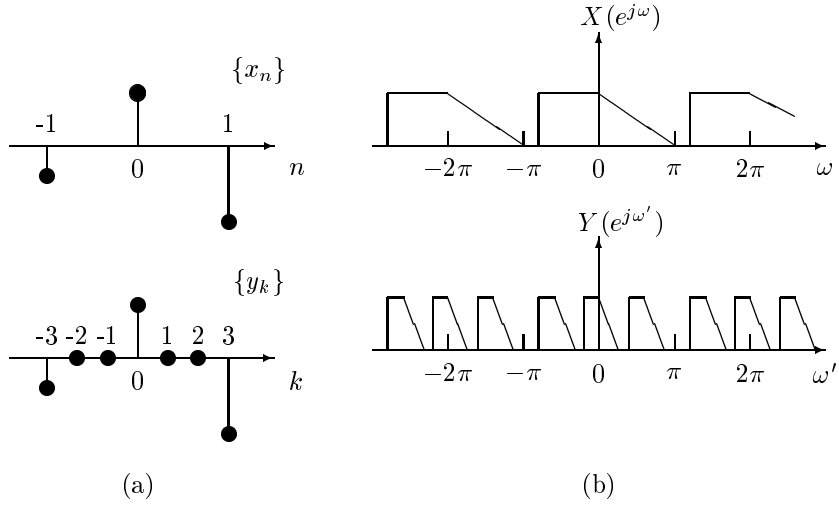


Figure 1.59. Interpolation by a factor $L = 3$: (a) in the time domain, (b) in the normalized radian frequency domain.

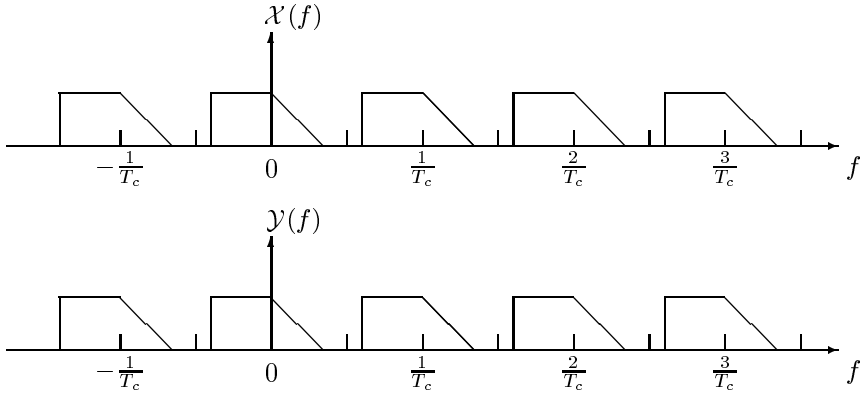


Figure 1.60. Effect of interpolation in the frequency domain.

Proof of (1.593). Observing (1.592) we get

$$Y(z) = \sum_{k=-\infty}^{+\infty} y_k z^{-k} = \sum_{n=-\infty}^{+\infty} y_{nL} z^{-nL} = \sum_{n=-\infty}^{+\infty} x_n z^{-nL} = X(z^L) \tag{1.598}$$

1.A.4 Decimator filter

In most applications, a downsampler is preceded by a lowpass digital filter, to form a *decimator filter* as illustrated in Figure 1.61. The filter ensures that the signal v_n is bandlimited, to avoid *aliasing* in the downsampling process.

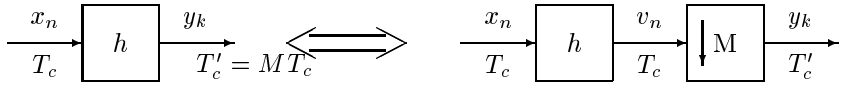


Figure 1.61. Decimator filter.

Let $h_n = h(nT_c)$. Then we have

$$y_k = v_{kM} \tag{1.599}$$

and

$$v_n = \sum_{i=-\infty}^{+\infty} h_i x_{n-i} \tag{1.600}$$

The output can be expressed as

$$y_k = \sum_{i=-\infty}^{+\infty} h_i x_{kM-i} = \sum_{n=-\infty}^{+\infty} h_{kM-n} x_n \tag{1.601}$$

Using definition (1.577) we get

$$g_{k,i} = h_i \qquad \forall k, i \tag{1.602}$$

Note that the overall system is not time invariant, unless the delay applied to the input is constrained to be a multiple of M.

From $V(z) = X(z)H(z)$ it follows that

$$Y(z) = \frac{1}{M} \sum_{m=0}^{M-1} H(z^{1/M} W_M^m) X(z^{1/M} W_M^m) \tag{1.603}$$

or, equivalently, recalling that $\omega' = 2\pi f M T_c$,

$$Y(e^{j\omega'}) = \frac{1}{M} \sum_{m=0}^{M-1} H(e^{j\frac{\omega' - 2\pi m}{M}}) X(e^{j\frac{\omega' - 2\pi m}{M}}) \tag{1.604}$$

If

$$H(e^{j\omega}) = \begin{cases} 1 & |\omega| \le \dfrac{\pi}{M} \\ 0 & \text{otherwise} \end{cases} \tag{1.605}$$

we obtain

$$Y(e^{j\omega'}) = \frac{1}{M} X\left(e^{\frac{j\omega'}{M}}\right) \qquad |\omega'| \le \pi \tag{1.606}$$

In this case h is a lowpass filter that avoids *aliasing* caused by sampling; if x is bandlimited, the specifications of h can be made less stringent.

The decimator filter transformations are illustrated in Figure 1.62 for $M = 4$.

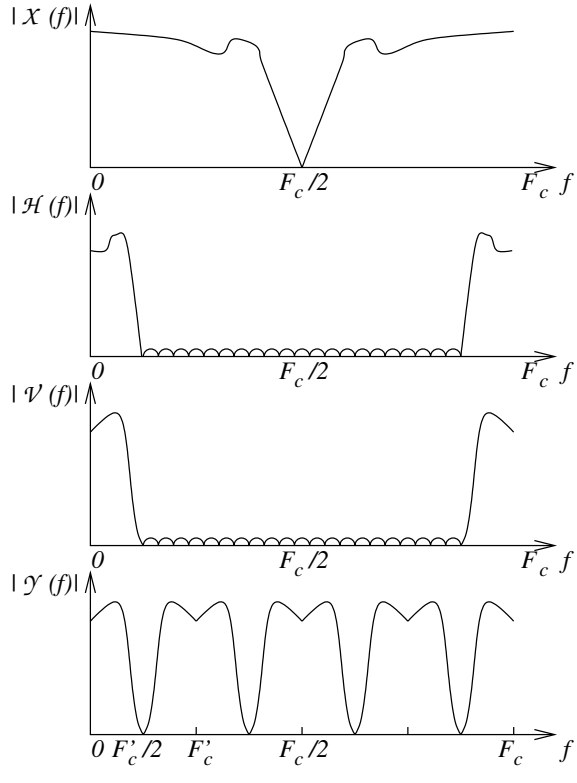


Figure 1.62. Frequency responses related to the transformations in a decimator filter for $M = 4$.

1.A.5 Interpolator filter

An interpolator filter is given by the cascade of an upsampler and a digital filter, as illustrated in Figure 1.63; the task of the digital filter is to suppress images created by upsampling [17].

Let $h_n = h(nT_c')$. Then we have the following input–output relations:

$$y_k = \sum_{j=-\infty}^{+\infty} h_{k-j} w_j \tag{1.607}$$

$$w_k = \begin{cases} x\left(\dfrac{k}{L}\right) & k = 0, \pm L, \ldots \\ 0 & \text{otherwise} \end{cases} \tag{1.608}$$

Therefore

$$y_k = \sum_{r=-\infty}^{+\infty} h_{k-rL} x_r \tag{1.609}$$

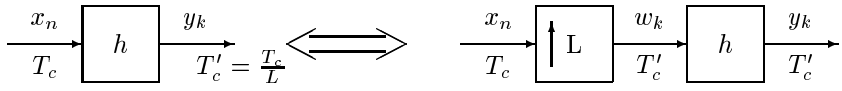


Figure 1.63. Interpolator filter.

Let $i = \lfloor k/L \rfloor - r$ and $g_{k,i} = h_{iL+(k)_{mod\,L}}$. From (1.609) we get

$$y_k = \sum_{i=-\infty}^{+\infty} g_{k,i} x_{\lfloor \frac{k}{L} \rfloor - i} \tag{1.610}$$

We note that $g_{k,i}$ is periodic in k of period L.

In the z-transform domain we find

$$W(z) = X(z^L) \tag{1.611}$$

$$Y(z) = H(z)W(z) = H(z)X(z^L) \tag{1.612}$$

or, equivalently,

$$Y(e^{j\omega'}) = H(e^{j\omega'})X(e^{j\omega' L}) \tag{1.613}$$

where $\omega' = 2\pi f T/L = \omega/L$.

The interpolator filter transformations in the time and frequency domains are illustrated in Figure 1.64 for $L = 3$.

If

$$H(e^{j\omega'}) = \begin{cases} 1 & |\omega'| \le \dfrac{\pi}{L} \\ 0 & \text{elsewhere} \end{cases} \tag{1.614}$$

we find

$$Y(e^{j\omega'}) = \begin{cases} X(e^{j\omega'}) & |\omega'| \le \dfrac{\pi}{L} \\ 0 & \text{elsewhere} \end{cases} \tag{1.615}$$

The relation between the input and output signal power for an interpolator filter is expressed by (1.419).

1.A.6 Rate conversion

Decimator and interpolator filters can be employed to vary the sampling frequency of a signal by an integer factor; in some applications, however, it is necessary to change the sampling frequency by a rational factor L/M. A possible procedure consists of first converting a discrete-time signal into a continuous-time signal by a *digital-to-analog converter* (DAC), then re-sampling it at the new frequency. It is, however, easier and more convenient to change the sampling frequency by discrete-time transformations, for example, using the structure of Figure 1.65.

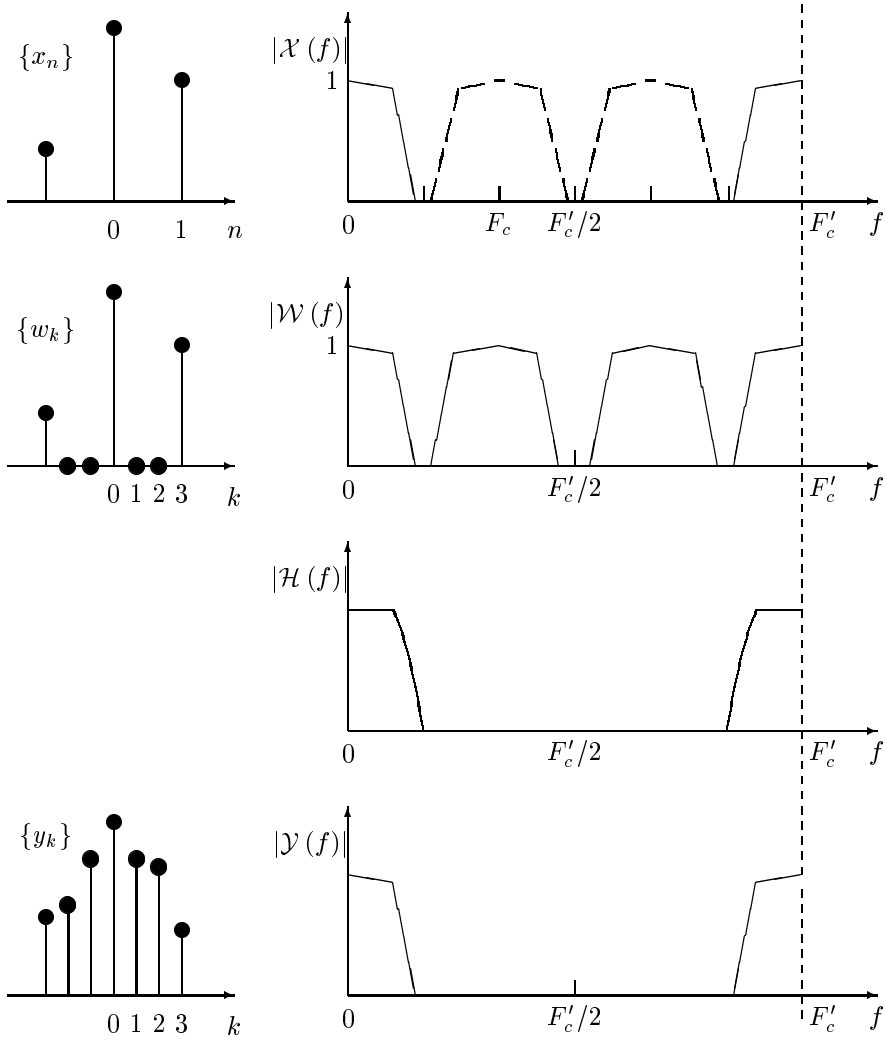


Figure 1.64. Time and frequency responses related to the transformations in an interpolator filter for $L = 3$.

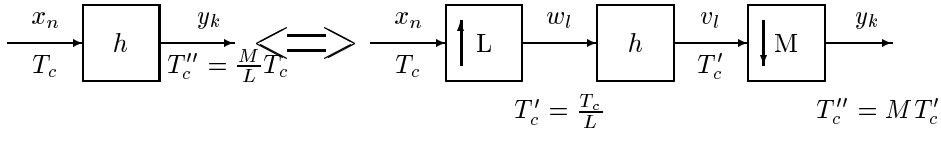


Figure 1.65. Sampling frequency conversion by a rational factor.

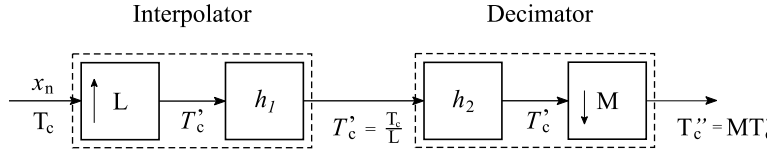


Figure 1.66. Decomposition of the system of Figure 1.65.

This system can be thought of as the cascade of an interpolator and decimator filter, as illustrated in Figure 1.66, where $h = h_1 * h_2$. We obtain that

$$H(e^{j\omega'}) = \begin{cases} 1 & |\omega'| \le \min\left(\frac{\pi}{L}, \frac{\pi}{M}\right) \\ 0 & \text{elsewhere} \end{cases} \tag{1.616}$$

In the time domain the following relation holds:

$$y_k = \sum_{i=-\infty}^{+\infty} g_{k,i} x_{\left\lfloor \frac{kM}{L} \right\rfloor - i} \tag{1.617}$$

where $g_{k,i} = h((iL + (kM)_{mod\,L})T_c')$ is the time-varying impulse response.

In the frequency domain we get

$$Y(e^{j\omega''}) = \frac{1}{M} \sum_{l=0}^{M-1} V(e^{j\frac{\omega''-2\pi l}{M}}) \tag{1.618}$$

As

$$V(e^{j\omega'}) = H(e^{j\omega'}) X(e^{j\omega' L}) \tag{1.619}$$

we obtain

$$Y(e^{j\omega''}) = \frac{1}{M} \sum_{l=0}^{M-1} H(e^{j\frac{\omega''-2\pi l}{M}}) X(e^{j\frac{\omega'' L-2\pi l}{M}}) \tag{1.620}$$

From (1.616) we have

$$Y(e^{j\omega''}) = \begin{cases} \frac{1}{M} X\left(e^{\frac{j\omega'' L}{M}}\right) & |\omega''| \le \min\left(\pi, \frac{\pi M}{L}\right) \\ 0 & \text{elsewhere} \end{cases} \tag{1.621}$$

or

$$\mathcal{Y}(f) = \frac{1}{M} \mathcal{X}(f) \quad \text{for} \quad |f| \le \min\left(\frac{1}{2T_c}, \frac{L}{2MT_c}\right) \tag{1.622}$$

Example 1.A.1 ($M > L$: $M = 5$, $L = 4$)
Transformations for $M = 5$ and $L = 4$ are illustrated in Figure 1.67. Observing the fact that $W(e^{j\omega'})$ is zero for $\frac{\pi}{M} \le \omega' \le 2\frac{\pi}{L} - \frac{\pi}{M}$, the desired result is obtained by a response $H(e^{j\omega'})$ that has the stopband cut-off frequency within this interval.

Example 1.A.2 ($M < L$: $M = 4$, $L = 5$)
The inverse transformation of the above example is obtained by a transformation with $M = 4$ and $L = 5$, as depicted in Figure 1.68.

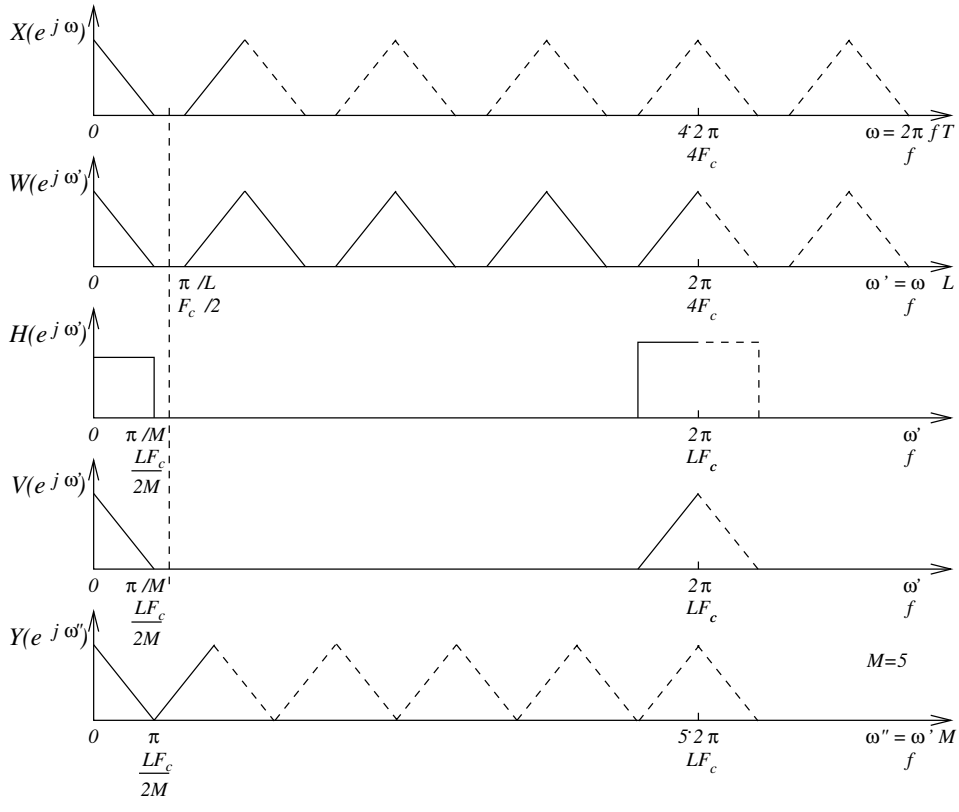


Figure 1.67. Rate conversion by a rational factor L/M where $M > L$.

1.A.7 Time interpolation

Referring to the interpolator filter h of Figure 1.63, one finds that if L is large the filter implementation may require non-negligible complexity; in fact, the number of coefficients required for an FIR filter implementation can be very large. Consequently, in the case of a very large interpolation factor L, after a first interpolator filter with a moderate value of the interpolation factor, the samples $\{y_k = y(kT_c')\}$ may be further time interpolated until the desired sampling accuracy is reached [17].

As shown in Figure 1.69, let $\{y_k\}$ be the sequence that we need to interpolate to produce the signal $z(t)$, $t \in \Re$; we describe below two time interpolation methods, linear and quadratic.

Linear interpolation

Given two samples y_{k-1} and y_k, the signal $z(t)$, limited to interval $[(k-1)\ T_c', kT_c']$, is obtained by the linear interpolation

$$z(t) = y_{k-1} + \frac{t - (k-1)T_c'}{T_c'}\,(y_k - y_{k-1}) \tag{1.623}$$

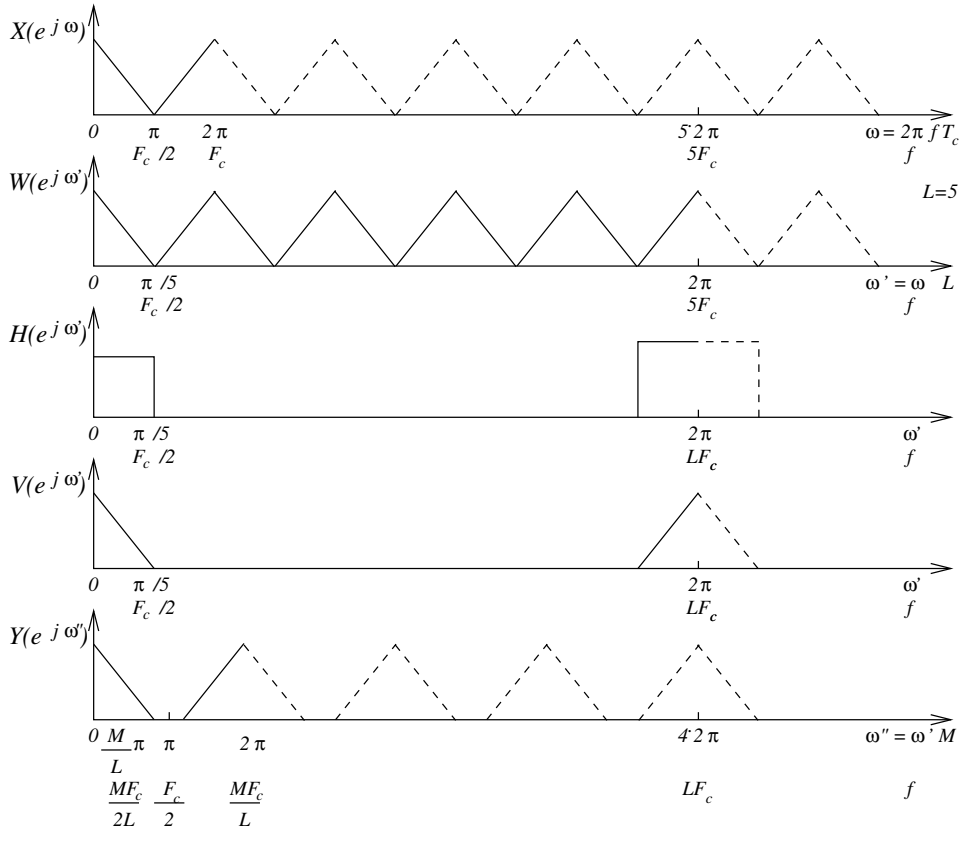


Figure 1.68. Rate conversion by a rational factor L/M where $M < L$.

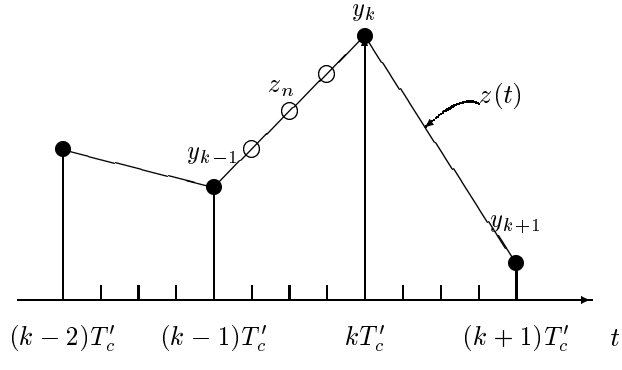


Figure 1.69. Linear interpolation in time by a factor $P = 4$.

For an interpolation factor P of y_k, we need to consider the sampling instants

$$nT_c'' = n\frac{T_c'}{P} \tag{1.624}$$

and the values of $z_n = z(nT_c'')$ are given by

$$z_n = y_{k-1} + \frac{n-(k-1)P}{P}(y_k - y_{k-1}) \tag{1.625}$$

where $n = (k-1)P, (k-1)P+1, \ldots, kP-1$. The case $k=1$ is of particular interest:

$$z_n = y_0 + \frac{n}{P}(y_1 - y_0) \qquad n = 0, 1, \ldots, P-1 \tag{1.626}$$

In fact, regarding y_0 and y_1 as the two most recent input samples, their linear interpolation originates the sequence of P values given by (1.626).

Quadratic interpolation

In many applications linear interpolation does not always yield satisfactory results. Therefore, instead of connecting two points with a straight line, one resorts to a polynomial of degree $Q-1$ passing through Q points that are determined by the samples of the input sequence. For this purpose the Lagrange interpolation is widely used. As an example we report here the case of quadratic interpolation. In this case we consider a polynomial of degree 2 that passes through 3 points that are determined by the input samples. Let y_{k-1}, y_k and y_{k+1} be the samples to interpolate by a factor P in the interval $[(k-1)T_c', (k+1)T_c']$. The quadratic interpolation yields the values

$$z_n = \frac{n'}{2P}\left(\frac{n'}{P} - 1\right) y_{k-1} + \left(1 - \frac{n'}{P}\right)\left(1 + \frac{n'}{P}\right) y_k + \frac{n'}{2P}\left(\frac{n'}{P} + 1\right) y_{k+1} \tag{1.627}$$

with $n' = 0, 1, \ldots, P-1$ and $n = (k-1)\,P + n'$.

1.A.8 The noble identities

We recall some important properties of decimator and interpolator filters, known as *noble identities*; they will be used extensively in the next section on polyphase decomposition.

Let $G(z)$ be a rational transfer function, i.e., a function expressed as the ratio of two polynomials in z or in z^{-1}; it is possible to exchange the order of downsampling and filtering, or the order of upsampling and filtering as illustrated in Figure 1.70; in other words, the system of Figure 1.70a is equivalent to that of Figure 1.70b, and the system of Figure 1.70c is equivalent to that of Figure 1.70d.

The proof of the noble identities is simple. For the first identity, it is sufficient to note that $W_M^{-mM} = 1$, hence

$$\begin{aligned} Y_2(z) &= \frac{1}{M}\sum_{m=0}^{M-1} X(z^{1/M} W_M^{-m}) G((z^{1/M} W_M^{-m})^M) \\ &= \frac{1}{M}\sum_{m=0}^{M-1} X(z^{1/M} W_M^{-m}) G(z) = Y_1(z) \end{aligned} \tag{1.628}$$

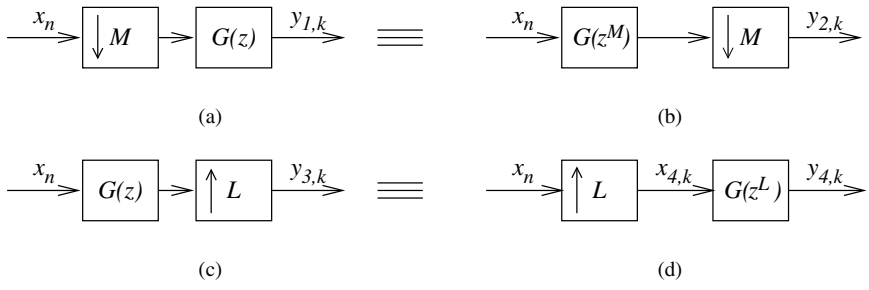


Figure 1.70. Noble identities.

For the second identity it is sufficient to observe that

$$Y_4(z) = G(z^L)X_4(z) = G(z^L)X(z^L) = Y_3(z) \tag{1.629}$$

1.A.9 The polyphase representation

The polyphase representation allows considerable simplifications in the analysis of transformations via interpolator and decimator filters, as well as the efficient implementation of such filters. To explain the basic concept, let us consider a filter having transfer function $H(z) = \sum_{n=0}^{\infty} h_n z^{-n}$. Separating the coefficients with even and odd time indices, we get

$$H(z) = \sum_{m=0}^{\infty} h_{2m} z^{-2m} + z^{-1} \sum_{m=0}^{\infty} h_{2m+1} z^{-2m} \tag{1.630}$$

Defining

$$E^{(0)}(z) = \sum_{m=0}^{\infty} h_{2m} z^{-m} \qquad E^{(1)}(z) = \sum_{m=0}^{\infty} h_{2m+1} z^{-m} \tag{1.631}$$

we can write $H(z)$ as

$$H(z) = E^{(0)}(z^2) + z^{-1} E^{(1)}(z^2) \tag{1.632}$$

To expand this idea, let M be an integer; we can always decompose $H(z)$ as

$$\begin{aligned} H(z) = &\sum_{m=0}^{\infty} h_{mM} z^{-mM} \\ &+ z^{-1} \sum_{m=0}^{\infty} h_{mM+1} z^{-mM} + \cdots + z^{-(M-1)} \sum_{m=0}^{\infty} h_{mM+M-1} z^{-mM} \end{aligned} \tag{1.633}$$

Letting

$$e_m^{(\ell)} = h_{mM+\ell} \qquad 0 \le \ell \le M-1 \tag{1.634}$$

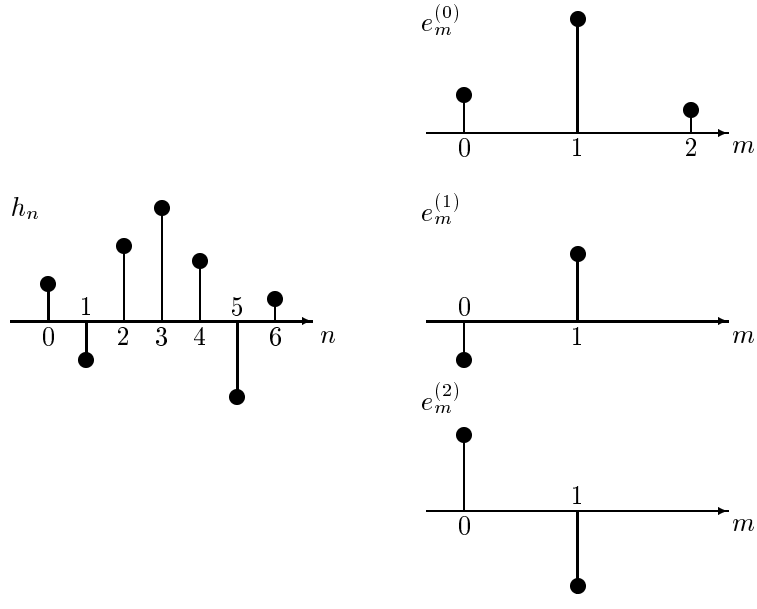


Figure 1.71. Polyphase representation of the impulse response $\{h_n\}$, $n = 0, \ldots, 6$, for $M = 3$.

we can express compactly the previous equation as

$$H(z) = \sum_{\ell=0}^{M-1} z^{-\ell} E^{(\ell)}(z^M) \tag{1.635}$$

where

$$E^{(\ell)}(z) = \sum_{i=0}^{\infty} e_i^{(\ell)} z^{-i} \tag{1.636}$$

The expression (1.635) is called the *type 1 polyphase representation* (with respect to M), and $E^{(\ell)}(z)$, where $\ell = 0, 1, \ldots, M-1$, the *polyphase components* of $H(z)$.

The polyphase representation of an impulse response $\{h_n\}$ with 7 coefficients is illustrated in Figure 1.71 for $M = 3$.

A variation of (1.635), called *type 2 polyphase representation*, is given by

$$H(z) = \sum_{\ell=0}^{M-1} z^{-(M-1-\ell)} R^{(\ell)}(z^M) \tag{1.637}$$

where the components $R^{(\ell)}(z)$ are permutations of $E^{(\ell)}(z)$, that is $R^{(\ell)}(z) = E^{(M-1-\ell)}(z)$.

Efficient implementations

The polyphase representation is the key to obtaining efficient implementation of decimator and interpolator filters. In the following, we will first consider the efficient implementations for $M = 2$ and $L = 2$, then we will extend the results to the general case.

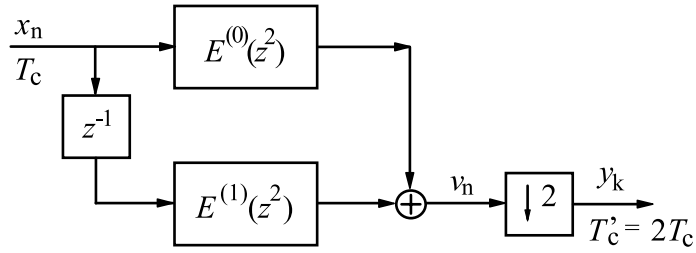


Figure 1.72. Implementation of a decimator filter using the type 1 polyphase representation for $M = 2$.

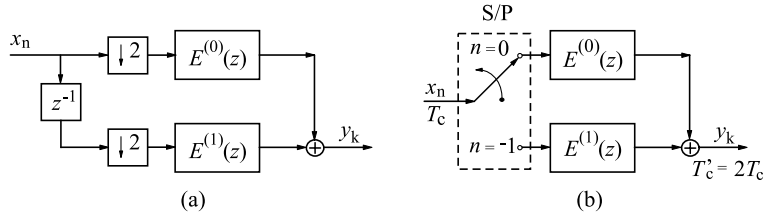


Figure 1.73. Optimized implementation of a decimator filter using the type 1 polyphase representation for $M = 2$.

Decimator filter. Referring to Figure 1.61, we consider a decimator filter with $M = 2$. By (1.635), we can represent $H(z)$ as illustrated in Figure 1.72; by the noble identities, the filter representation can be drawn as in Figure 1.73a. The structure can be also drawn as in Figure 1.73b, where input samples $\{x_n\}$ are alternately presented at the input to the two filters $e^{(0)}$ and $e^{(1)}$; this latter operation is generally called *serial–to–parallel* (S/P) conversion. Note that the system output is now given by the sum of the outputs of two filters, each operating at half the input frequency and having half the number of coefficients as the original filter.

To formalize the above ideas, let N be the number of coefficients of h, and $N^{(0)}$ and $N^{(1)}$ be the number of coefficients of $e^{(0)}$ and $e^{(1)}$, respectively, so that $N = N^{(0)} + N^{(1)}$. In this implementation $e^{(\ell)}$ requires $N^{(\ell)}$ multiplications and $N^{(\ell)} - 1$ additions; the total cost is still N multiplications and $N - 1$ additions, but, as $e^{(\ell)}$ operates at half the input rate, the computational complexity in terms of multiplications per second (MPS) is

$$\text{MPS} = \frac{NF_c}{2} \tag{1.638}$$

while the number of additions per second (APS) is given by

$$\text{APS} = \frac{(N-1)F_c}{2} \tag{1.639}$$

Therefore the complexity is about one half the complexity of the original filter. The efficient implementation for the general case is obtained as an extension of the case for $M = 2$ and is shown in Figure 1.74.

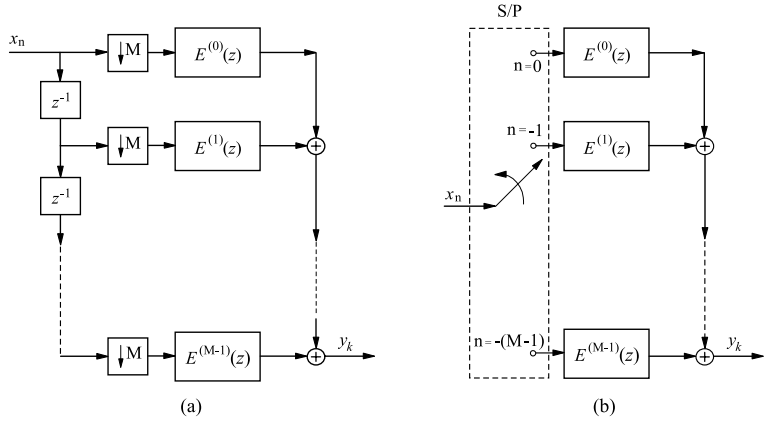


Figure 1.74. Implementation of a decimator filter using the type 1 polyphase representation.

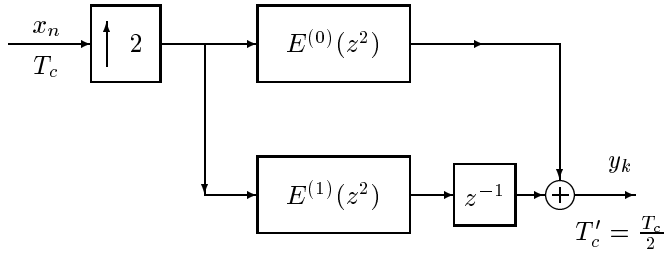


Figure 1.75. Implementation of an interpolator filter using the type 1 polyphase representation for $L = 2$.

Interpolator filter. With reference to Figure 1.63, we consider an interpolator filter with $L = 2$. By (1.635), we can represent $H(z)$ as illustrated in Figure 1.75; by the noble identities, the filter representation can be drawn as in Figure 1.76a. The structure can be also drawn as in Figure 1.76b, where output samples are alternately taken from the output of the two filters $e^{(0)}$ and $e^{(1)}$; this latter operation is generally called *parallel–to–serial* (P/S) conversion. Remarks on the computational complexity are analogous to those of the decimator filter case.

In the general case, efficient implementations are easily obtainable as extensions of the case for $L = 2$ and are shown in Figure 1.77. The type 2 polyphase implementations of interpolator filters are depicted in Figure 1.78.

Interpolator-decimator filter. As illustrated in Figure 1.79, at the receiver of a transmission system it is often useful to interpolate the signal $\{r(nT_Q)\}$ from T_Q to T'_Q to get the signal $\{x(qT'_Q)\}$.

Let

$$r_n = r(nT_Q) \qquad x_q = x(qT'_Q) \tag{1.640}$$

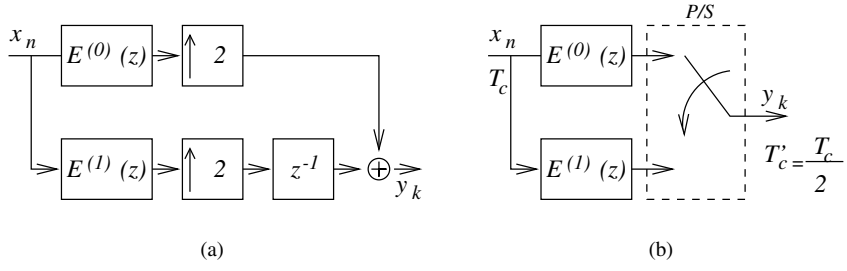


Figure 1.76. Optimized implementation of an interpolator filter using the type 1 polyphase representation for $L = 2$.

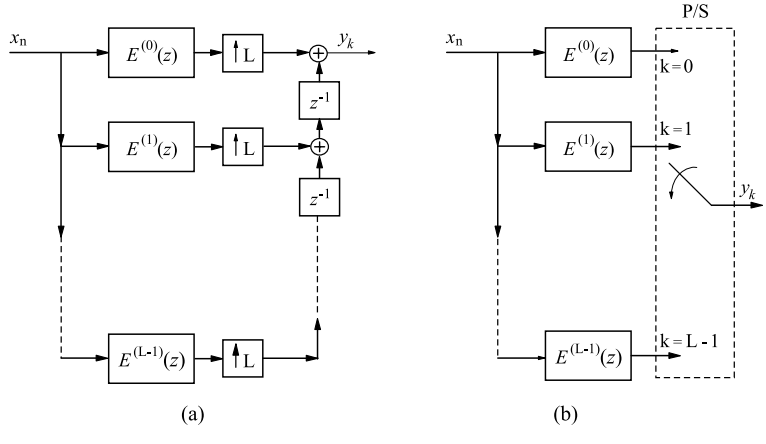


Figure 1.77. Implementation of an interpolator filter using the type 1 polyphase representation.

The sequence $\{x(qT'_Q)\}$ is then downsampled with *timing phase* t_0. Let y_k be the output with sampling period T_c,

$$y_k = x(kT_c + t_0) \tag{1.641}$$

To simplify the notation, we assume the following relations:

$$L = \frac{T_Q}{T'_Q} \qquad M = \frac{T_c}{T_Q} \tag{1.642}$$

with L and M positive integer numbers. Moreover, we assume that t_0 is a multiple of T'_Q,

$$\frac{t_0}{T'_Q} = \ell_0 + \mathcal{L}_0 L \tag{1.643}$$

where $\ell_0 \in \{0, 1, \ldots, L-1\}$, and $\mathcal{L}_0$ is a non-negative integer number. For the general case of an interpolator-decimator filter where t_0 and the ratio T_c/T'_Q are not constrained, we refer to [18] (see also Chapter 14).

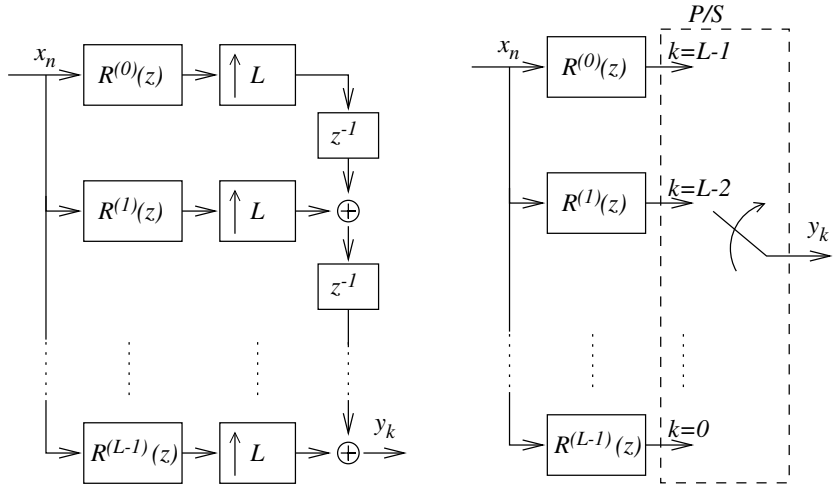


Figure 1.78. Implementation of an interpolator filter using the type 2 polyphase representation.

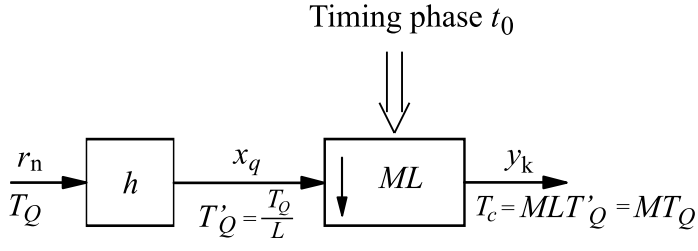


Figure 1.79. Interpolator-decimator filter.

Based on the above equations we have

$$y_k = x_{kML+\ell_0+\mathcal{L}_0 L} \tag{1.644}$$

We now recall the polyphase representation of $\{h(nT'_Q)\}$ with L phases

$$\{E^{(\ell)}(z)\} \quad \ell = 0, 1, \ldots, L-1 \tag{1.645}$$

The interpolator filter structure from T_Q to T'_Q is illustrated in Figure 1.77. For the special case $M = 1$, that is for $T_c = T_Q$, the implementation of the interpolator-decimator filter is given in Figure 1.80, where

$$y_k = v_{k+\mathcal{L}_0} \tag{1.646}$$

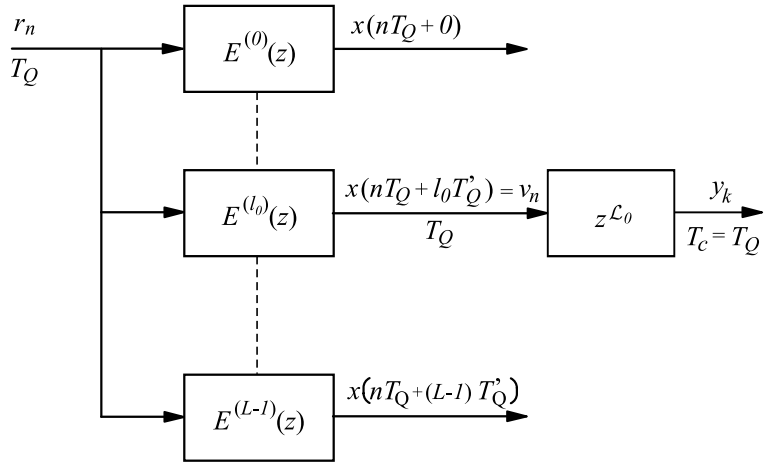
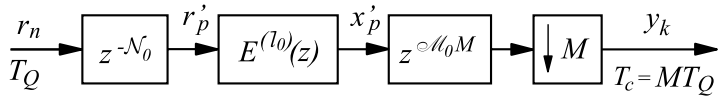
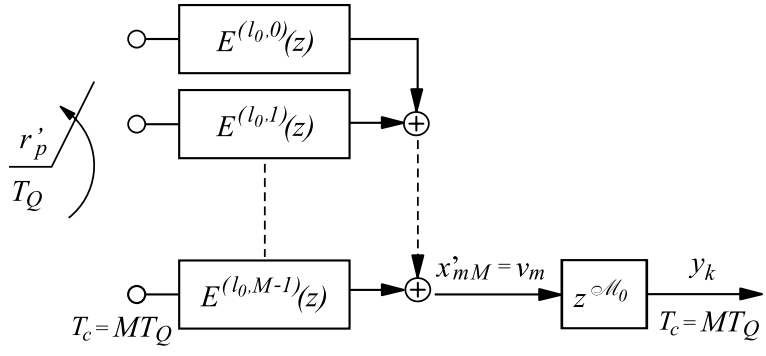


Figure 1.80. Polyphase implementation of an interpolator-decimator filter with timing phase $t_0 = (\ell_0 + \mathcal{L}_0 L)T'_Q$.



(a) Basic scheme



(b) Efficient implementation

Figure 1.81. Implementation of an interpolator-decimator filter with timing phase $t_0 = (\ell_0 + \mathcal{L}_0 L)T'_Q$.

In other words, $\{y_k\}$ coincides with the signal $\{v_n\}$ at the output of branch ℓ_0 of the polyphase structure. In practice we need to ignore the first $\mathcal{L}_0$ samples of $\{v_n\}$, as the relation between $\{v_n\}$ and $\{y_k\}$ must take into account a lead, $z^{\mathcal{L}_0}$, of $\mathcal{L}_0$ samples. With reference to Figure 1.80, the output $\{x_q\}$ at instants that are multiples of T'_Q is given by the outputs of the various polyphase branches in sequence. In fact, let $q = \ell + nL$, $\ell = 0, 1, \ldots, L-1$, and n integer, we have

$$x_{\ell+nL} = x(nLT'_Q + \ell T'_Q) = x(nT_Q + \ell T'_Q) \tag{1.647}$$

We now consider the general case $M \neq 1$. First, to downsample the signal interpolated at T'_Q one can still use the polyphase structure of Figure 1.80. In any case, once t_0 is chosen, the branch is identified (say ℓ_0) and its output must be downsampled by a factor ML. Notice that there is the timing lead $\mathcal{L}_0 L$ in (1.643) to be considered. Given $\mathcal{L}_0$, we determine a positive integer $\mathcal{N}_0$ so that $\mathcal{L}_0 + \mathcal{N}_0$ is a multiple of M, that is

$$\mathcal{L}_0 + \mathcal{N}_0 = \mathcal{M}_0 M \tag{1.648}$$

The structure of Figure 1.80, considering only branch ℓ_0, is equivalent to that given in Figure 1.81a, in which we have introduced a lag of $\mathcal{N}_0$ samples on the sequence $\{r_n\}$ and a further lead of $\mathcal{N}_0$ samples before the downsampler. In particular we have

$$r'_p = r_{p-\mathcal{N}_0} \quad \text{and} \quad x'_p = x_{p-\mathcal{N}_0} \tag{1.649}$$

As a result, the signal is not modified before the downsampler.

Using now the representation of $E^{(\ell_0)}(z^L)$ in M phases:

$$E^{(\ell_0,m)}(z^{LM}) \qquad m = 0, 1, \ldots, M-1 \tag{1.650}$$

an efficient implementation of the interpolator-decimator filter is given in Figure 1.81b.

Appendix 1.B *Generation of Gaussian noise*

Let $\bar{w} = \bar{w}_I + j\bar{w}_Q$ be a complex Gaussian r.v. with zero mean and unit variance; note that $\bar{w}_I = Re\,[\bar{w}]$ and $\bar{w}_Q = Im\,[\bar{w}]$. In polar notation,

$$\bar{w} = A\, e^{j\varphi} \tag{1.651}$$

It can be shown that φ is a uniform r.v. in $[0, 2\pi)$, and A is a Rayleigh r.v. with probability distribution

$$P[A \leq a] = \begin{cases} 1 - e^{-a^2} & a > 0 \\ 0 & a < 0 \end{cases} \tag{1.652}$$

Observing (1.652) and (1.651), if u_1 and u_2 are two uniform r.v.s in the interval $[0, 1)$, then

$$A = \sqrt{-\ln(1 - u_1)} \tag{1.653}$$

and

$$\varphi = 2\pi\, u_2 \tag{1.654}$$

In terms of real components, it results that

$$\bar{w}_I = A\cos\varphi \quad \text{and} \quad \bar{w}_Q = A\sin\varphi \tag{1.655}$$

are two statistically independent Gaussian r.v.s, each with zero mean and variance equal to 0.5.

The r.v. $\bar{w}$ is also called circularly symmetric Gaussian r.v., as the real and imaginary components, being statistically independent with equal variance, have a circularly symmetric Gaussian joint probability density function.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 2

The Wiener filter and linear prediction

The theory of the Wiener filter [1, 2] that will be presented in this chapter is fundamental to the comprehension of several important applications. The development of this theory assumes the knowledge of the correlation functions of the relevant processes. An approximation of the Wiener filter can be obtained by least squares methods, through realizations of the processes involved.

2.1 The Wiener filter

With reference to Figure 2.1, let x and d be two individually and jointly wide sense stationary random processes with zero mean; the problem is to determine the FIR filter so that, if the filter input is $x(k)$, the output $y(k)$ replicates as closely as possible $d(k)$. The Wiener theory provides the means to design the required filter.

The FIR filter in Figure 2.1 is called *transversal filter*, as the output is formed by summing the products of the delayed input samples by suitable coefficients. If we indicate with $\{c_n\}$, $n = 0, 1, \ldots, N-1$, the N coefficients of the filter, we have

$$y(k) = \sum_{n=0}^{N-1} c_n x(k-n) \tag{2.1}$$

If $d(k)$ is the *desired sample* at the filter output at instant k, we define the *estimation error* as

$$e(k) = d(k) - y(k) \tag{2.2}$$

In the Wiener theory, to cause the filter output $y(k)$ to replicate as closely as possible $d(k)$, the coefficients of the filter are determined using the *minimum mean-square error* (MMSE) criterion. Therefore the cost function is defined as

$$J = E[|e(k)|^2] \tag{2.3}$$

and the coefficients of the optimum filter are those that minimize J:

$$\min_{\{c_n\}, n=0,1,\ldots,N-1} J \tag{2.4}$$

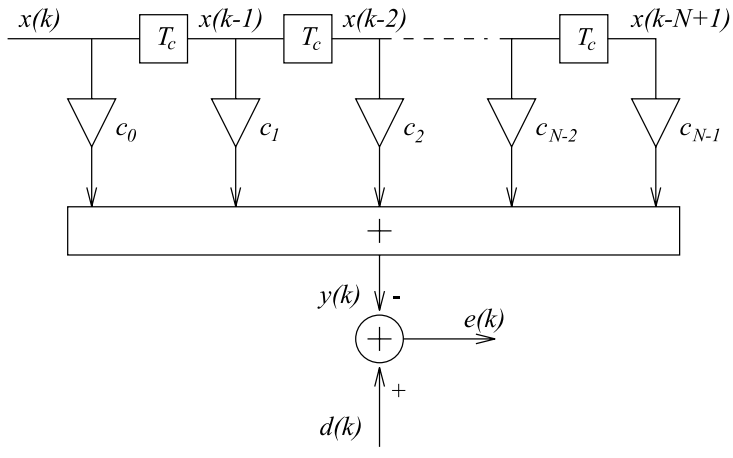


Figure 2.1. The Wiener filter with N coefficients.

The Wiener filter problem can be formulated as the problem of estimating $d(k)$ by a linear combination of $x(k), \ldots, x(k-N+1)$. A brief introduction to estimation theory is given in Appendix 2.A; in the second half of the Appendix, of which reading should be deferred until the end of this section, the formulation of the Wiener theory is further extended to the case of vector signals.

Matrix formulation

The problem introduced in the previous section is now formulated using matrix notation. We define:[1]

1. *Coefficient vector*

$$\mathbf{c}^T = [c_0, c_1, \ldots, c_{N-1}] \tag{2.5}$$

2. *Filter input vector at instant k*

$$\mathbf{x}^T(k) = [x(k), x(k-1), \ldots, x(k-N+1)] \tag{2.6}$$

The filter output at instant k is expressed as

$$y(k) = \mathbf{c}^T \mathbf{x}(k) = \mathbf{x}^T(k)\,\mathbf{c} \tag{2.7}$$

and the estimation error as

$$e(k) = d(k) - \mathbf{c}^T \mathbf{x}(k) \tag{2.8}$$

[1] The components of an N-dimensional vector are usually identified by an index varying either from 1 to N or from 0 to $N-1$.

Moreover,

$$y^*(k) = \mathbf{c}^H \mathbf{x}^*(k) = \mathbf{x}^H(k)\mathbf{c}^* \qquad e^*(k) = d^*(k) - \mathbf{c}^H \mathbf{x}^*(k) \tag{2.9}$$

We express now the cost function J as a function of the vector $\mathbf{c}$. We will then seek that particular vector $\mathbf{c}_{opt}$ that minimizes J. Recalling the definition

$$J = E[e(k)e^*(k)] = E[(d(k) - \mathbf{x}^T(k)\mathbf{c})(d^*(k) - \mathbf{c}^H \mathbf{x}^*(k))] \tag{2.10}$$

and computing the products, it follows

$$\begin{aligned} J = {} & E[d^*(k)d(k)] - \mathbf{c}^H E[\mathbf{x}^*(k)d(k)] + \\ & - E[d^*(k)\mathbf{x}^T(k)]\mathbf{c} + \mathbf{c}^H E[\mathbf{x}^*(k)\mathbf{x}^T(k)]\mathbf{c} \end{aligned} \tag{2.11}$$

Assuming that $\mathbf{x}$ and d are individually and jointly WSS, we introduce the following quantities:

1. *Variance of the desired signal*

$$\sigma_d^2 = E[d(k)d^*(k)] \tag{2.12}$$

2. *Correlation between the desired output at instant k and the filter input vector at the same instant*

$$\mathbf{r}_{d\mathbf{x}} = E[d(k)\mathbf{x}^*(k)] = \begin{bmatrix} E[d(k)x^*(k)] \\ E[d(k)x^*(k-1)] \\ \vdots \\ E[d(k)x^*(k-N+1)] \end{bmatrix} = \mathbf{p} \tag{2.13}$$

The components of $\mathbf{p}$ are given by

$$[\mathbf{p}]_n = E[d(k)x^*(k-n)] = r_{dx}(n) \qquad n = 0, 1, \ldots, N-1 \tag{2.14}$$

Moreover, it holds

$$\mathbf{p}^H = E[d^*(k)\mathbf{x}^T(k)] \tag{2.15}$$

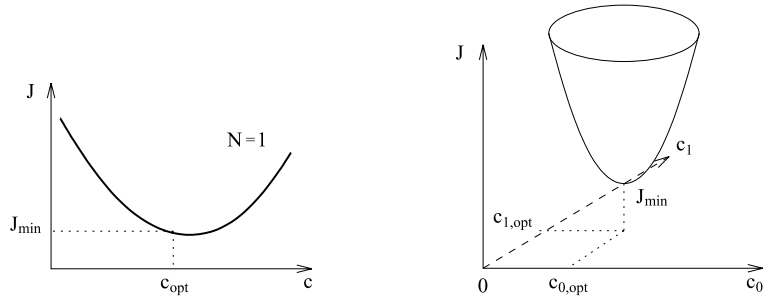
3. $N \times N$ *correlation matrix* of the filter input vector, as defined in (1.346),

$$\mathbf{R} = E[\mathbf{x}^*(k)\mathbf{x}^T(k)] \tag{2.16}$$

Then

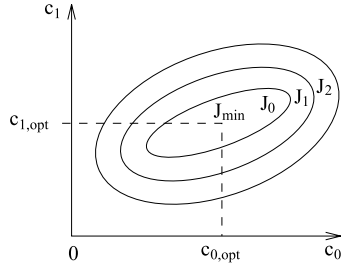
$$J = \sigma_d^2 - \mathbf{c}^H \mathbf{p} - \mathbf{p}^H \mathbf{c} + \mathbf{c}^H \mathbf{R}\mathbf{c} \tag{2.17}$$

The cost function J, considered as a function of $\mathbf{c}$, is a quadratic function. Then, if $\mathbf{R}$ is positive definite, J admits one and only one minimum value. Plots of J are shown in Figure 2.2, for the particular cases $N = 1$ and $N = 2$.



(a) Plot of J vs. c for N = 1.

(b) Plot of J vs. c for N = 2.



(c) Equal J contours for N = 2.

Figure 2.2. Plot of J for the cases $N = 1$ and $N = 2$.

Determination of the optimum filter coefficients

It is now a matter of finding the minimum of (2.17) with respect to $\mathbf{c}$. Recognizing that, as $\mathbf{c} = \mathbf{c}_I + j\mathbf{c}_Q$, the real independent variables are $2N$, to accomplish this task we define the derivative of J with respect to $\mathbf{c}$ as the gradient vector

$$\nabla_{\mathbf{c}} J = \frac{\partial J}{\partial \mathbf{c}} = \begin{bmatrix} \dfrac{\partial J}{\partial c_{0,I}} + j\dfrac{\partial J}{\partial c_{0,Q}} \\ \dfrac{\partial J}{\partial c_{1,I}} + j\dfrac{\partial J}{\partial c_{1,Q}} \\ \vdots \\ \dfrac{\partial J}{\partial c_{N-1,I}} + j\dfrac{\partial J}{\partial c_{N-1,Q}} \end{bmatrix} \tag{2.18}$$

and also

$$\nabla_{\mathbf{c}} J = \nabla_{\mathbf{c}_I} J + j\nabla_{\mathbf{c}_Q} J \tag{2.19}$$

In general, because the vector $\mathbf{p}$ and the autocorrelation matrix $\mathbf{R}$ are complex, we also write

$$\mathbf{p} = \mathbf{p}_I + j\mathbf{p}_Q \tag{2.20}$$

and

$$\mathbf{R} = \mathbf{R}_I + j\mathbf{R}_Q \tag{2.21}$$

If now we take the derivative of the terms of (2.17) using (2.19), we find

$$\nabla_{\mathbf{c}_I} \mathbf{p}^H \mathbf{c} = \nabla_{\mathbf{c}_I} \mathbf{p}^H (\mathbf{c}_I + j\mathbf{c}_Q) = \mathbf{p}^H \tag{2.22}$$

$$\nabla_{\mathbf{c}_Q} \mathbf{p}^H \mathbf{c} = \nabla_{\mathbf{c}_Q} \mathbf{p}^H (\mathbf{c}_I + j\mathbf{c}_Q) = j\mathbf{p}^H \tag{2.23}$$

$$\nabla_{\mathbf{c}_I} \mathbf{c}^H \mathbf{p} = \nabla_{\mathbf{c}_I} (\mathbf{c}_I^T - j\mathbf{c}_Q^T)\mathbf{p} = \mathbf{p} \tag{2.24}$$

$$\nabla_{\mathbf{c}_Q} \mathbf{c}^H \mathbf{p} = \nabla_{\mathbf{c}_Q} (\mathbf{c}_I^T - j\mathbf{c}_Q^T)\mathbf{p} = -j\mathbf{p} \tag{2.25}$$

$$\nabla_{\mathbf{c}_I} (\mathbf{c}^H \mathbf{R}\mathbf{c}) = 2\mathbf{R}_I \mathbf{c}_I - 2\mathbf{R}_Q \mathbf{c}_Q \tag{2.26}$$

$$\nabla_{\mathbf{c}_Q} (\mathbf{c}^H \mathbf{R}\mathbf{c}) = 2\mathbf{R}_I \mathbf{c}_Q + 2\mathbf{R}_Q \mathbf{c}_I \tag{2.27}$$

From the above equations we obtain

$$\nabla_{\mathbf{c}} \mathbf{p}^H \mathbf{c} = \mathbf{0} \tag{2.28}$$

$$\nabla_{\mathbf{c}} \mathbf{c}^H \mathbf{p} = 2\mathbf{p} \tag{2.29}$$

$$\nabla_{\mathbf{c}} \mathbf{c}^H \mathbf{R}\mathbf{c} = 2\mathbf{R}\mathbf{c} \tag{2.30}$$

Substituting the above results into (2.17), it turns out

$$\nabla_{\mathbf{c}} J = -2\mathbf{p} + 2\mathbf{R}\mathbf{c} = 2(\mathbf{R}\mathbf{c} - \mathbf{p}) \tag{2.31}$$

For the optimum coefficient vector $\mathbf{c}_{opt}$ the components of $\nabla_{\mathbf{c}} J$ are all equal to zero, hence we get

$$\mathbf{R}\mathbf{c}_{opt} = \mathbf{p} \tag{2.32}$$

Then (2.32) is called the *Wiener–Hopf equation* (W-H).

Observation 2.1
The computation of the optimum coefficients $\mathbf{c}_{opt}$ requires the knowledge only of the input correlation matrix $\mathbf{R}$ and of the cross-correlation vector $\mathbf{p}$ between the desired output and the input vector.

In scalar form, the Wiener–Hopf equation is a system of N equations in N unknowns:

$$\sum_{i=0}^{N-1} c_{opt,i}\, r_x(n-i) = r_{dx}(n) \qquad n = 0, 1, \ldots, N-1 \tag{2.33}$$

If $\mathbf{R}^{-1}$ exists, the solution of (2.32) is

$$\mathbf{c}_{opt} = \mathbf{R}^{-1}\mathbf{p} \tag{2.34}$$

The principle of orthogonality

It is interesting to observe the relation

$$E[e(k)\mathbf{x}^*(k)] = E[\mathbf{x}^*(k)(d(k) - \mathbf{x}^T(k)\mathbf{c})] = \mathbf{p} - \mathbf{R}\mathbf{c} \tag{2.35}$$

which for $\mathbf{c} = \mathbf{c}_{opt}$ yields

$$E[e(k)\mathbf{x}^*(k)] = \mathbf{0} \tag{2.36}$$

Formally the following is true.

Theorem 2.1 (Principle of orthogonality)
The condition of optimality for $\mathbf{c}$ is satisfied if $e(k)$ and $\mathbf{x}(k)$ are orthogonal.[2] In scalar form, the filter is optimum if $e(k)$ is orthogonal to $\{x(k-n)\}$, $n = 0, 1, \ldots, N-1$, that is

$$E[e(k)x^*(k-n)] = 0 \qquad n = 0, 1, \ldots, N-1 \tag{2.37}$$

Corollary 2.1
For $\mathbf{c} = \mathbf{c}_{opt}$, $e(k)$ and $y(k)$ are orthogonal:

$$E[e(k)y^*(k)] = 0 \qquad \text{for } \mathbf{c} = \mathbf{c}_{opt} \tag{2.38}$$

In fact, using the orthogonality principle,

$$\begin{aligned} E[e(k)y^*(k)] &= E[e(k)\mathbf{c}^H\mathbf{x}^*(k)] \\ &= \mathbf{c}^H E[e(k)\mathbf{x}^*(k)] \\ &= \mathbf{c}^H\mathbf{0} \\ &= 0 \end{aligned} \tag{2.39}$$

For an optimum filter, Figure 2.3 depicts the relation between the three signals $d(k)$, $e(k)$, and $y(k)$.

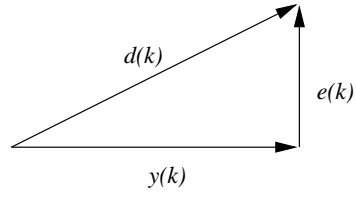


Figure 2.3. Orthogonality of signals for an optimum filter.

[2] Note that orthogonality holds only if e and $\mathbf{x}$ are considered at the same instant. In other words, the notion of orthogonality between random variables is used.

Expression of the minimum mean-square error

We now determine the value of the cost function J in correspondence of $\mathbf{c}_{opt}$. Substituting the expression (2.32) of $\mathbf{c}_{opt}$ in (2.17), we get

$$\begin{aligned} J_{min} &= \sigma_d^2 - \mathbf{c}_{opt}^H \mathbf{p} - \mathbf{p}^H \mathbf{c}_{opt} + \mathbf{c}_{opt}^H \mathbf{p} \\ &= \sigma_d^2 - \mathbf{p}^H \mathbf{c}_{opt} \end{aligned} \tag{2.40}$$

Another useful expression of J_{min} is obtained from (2.2):

$$d(k) = e(k) + y(k) \tag{2.41}$$

As $e(k)$ and $y(k)$ are orthogonal for $\mathbf{c} = \mathbf{c}_{opt}$, then

$$\sigma_d^2 = J_{min} + \sigma_y^2 \tag{2.42}$$

whereby it follows

$$J_{min} = \sigma_d^2 - \sigma_y^2 \tag{2.43}$$

Using (2.40) we can find an alternative expression to (2.17) for the cost function J:

$$J = J_{min} + (\mathbf{c} - \mathbf{c}_{opt})^H \mathbf{R} (\mathbf{c} - \mathbf{c}_{opt}) \tag{2.44}$$

Recalling that the autocorrelation matrix is positive semi-definite, it follows that the quantity $(\mathbf{c} - \mathbf{c}_{opt})^H \mathbf{R} (\mathbf{c} - \mathbf{c}_{opt})$ is non-negative and in particular it vanishes for $\mathbf{c} = \mathbf{c}_{opt}$.

Characterization of the cost function surface

The result expressed by (2.44) allows further observations on J. In fact, using the decomposition (1.369) we get

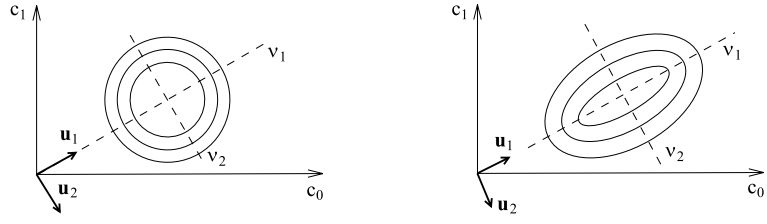
$$J = J_{min} + (\mathbf{c} - \mathbf{c}_{opt})^H \mathbf{U} \mathbf{\Lambda} \mathbf{U}^H (\mathbf{c} - \mathbf{c}_{opt}) \tag{2.45}$$

Let us now define

$$\boldsymbol{v} = \begin{bmatrix} v_1 \\ \vdots \\ v_N \end{bmatrix} = \mathbf{U}^H (\mathbf{c} - \mathbf{c}_{opt}) \tag{2.46}$$

where $v_i = \mathbf{u}_i^H (\mathbf{c} - \mathbf{c}_{opt})$. The vector $\boldsymbol{v}$ may be interpreted as a translation and a rotation of the vector $\mathbf{c}$. Then J assumes the form:

$$\begin{aligned} J &= J_{min} + \boldsymbol{v}^H \mathbf{\Lambda} \boldsymbol{v} \\ &= J_{min} + \sum_{i=1}^{N} \lambda_i |v_i|^2 \\ &= J_{min} + \sum_{i=1}^{N} \lambda_i |\mathbf{u}_i^H (\mathbf{c} - \mathbf{c}_{opt})|^2 \end{aligned} \tag{2.47}$$



(a) Case $\lambda_1 \simeq \lambda_2$. (b) Case $\lambda_1 < \lambda_2$.

Figure 2.4. Loci of points with constant J (contour plots).

The result (2.47) expresses the excess mean-square error $J - J_{min}$ as the sum of N components in the direction of each eigenvector of $\mathbf{R}$. Note that each component is proportional to the corresponding eigenvalue.

The above observation allows us to deduce that J increases more rapidly in the direction of the eigenvector corresponding to the maximum eigenvalue λ_{max}. Likewise the increase is slower in the direction of the eigenvector corresponding to the minimum eigenvalue λ_{min}. Let $\mathbf{u}_{\lambda_{max}}$ and $\mathbf{u}_{\lambda_{min}}$ denote the eigenvalues of $\mathbf{R}$ in correspondence of λ_{max} and λ_{min}, respectively; it follows that $\nabla_{\mathbf{c}} J$ is largest along $\mathbf{u}_{\lambda_{max}}$. This is also observed in Figure 2.4, where sets (loci) of points $\mathbf{c}$ for which a constant value of J is obtained are graphically represented. In the 2-dimensional case they trace ellipses with axes that are parallel to the direction of the eigenvectors and ratio of axes that is related to the value of the eigenvalues.

The Wiener filter in the z-domain

For a filter with an infinite number of coefficients, not necessarily causal, the equation (2.33) of the optimum filter becomes

$$\sum_{i=-\infty}^{+\infty} c_{opt,i}\, r_x(n-i) = r_{dx}(n) \qquad \forall n \tag{2.48}$$

Taking the z-transform of both members yields

$$C_{opt}(z) P_x(z) = P_{dx}(z) \tag{2.49}$$

Then the transfer function of the optimum filter is given by

$$C_{opt}(z) = \frac{P_{dx}(z)}{P_x(z)} \tag{2.50}$$

We note that while (2.34) is useful in evaluating the coefficients of the optimum FIR filter, equation (2.50) is employed to analyze the system in the general case of an IIR filter.

Example 2.1.1
Let $d(k) = h * x(k)$, as shown in Figure 2.5. In this case, from Table 1.3,

$$P_{dx}(z) = P_x(z) H(z) \tag{2.51}$$

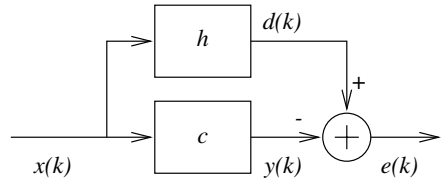


Figure 2.5. An application of the Wiener filter theory.

The optimum filter is given by

$$C_{opt}(z) = H(z) \tag{2.52}$$

From (2.40) in scalar notation, applying Fourier transform properties, we get

$$\begin{aligned} J_{min} &= \sigma_d^2 - \sum_{i=0}^{N-1} c_{opt,i} r_{dx}^*(i) \\ &= \sigma_d^2 - \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \mathcal{P}_{dx}^*(f) C_{opt}(e^{j2\pi f T_c})\, df \end{aligned} \tag{2.53}$$

Using (2.50):

$$\begin{aligned} J_{min} &= \sigma_d^2 - \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \mathcal{P}_{dx}^*(f) \frac{\mathcal{P}_{dx}(f)}{\mathcal{P}_x(f)}\, df \\ &= \sigma_d^2 - \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \frac{|\mathcal{P}_{dx}(f)|^2}{\mathcal{P}_x(f)}\, df \\ &= \sigma_d^2 - T_c \int_{-\frac{1}{2T_c}}^{\frac{1}{2T_c}} \frac{|P_{dx}(e^{j2\pi f T_c})|^2}{P_x(e^{j2\pi f T_c})}\, df \end{aligned} \tag{2.54}$$

Example 2.1.2
We want to filter the noise from a signal given by one complex sinusoid (tone) plus noise, i.e.

$$x(k) = A e^{j(\omega_0 k + \varphi)} + w(k) \tag{2.55}$$

In (2.55) $\omega_0 = 2\pi f_0 T_c$ is the tone radian frequency normalized to the sampling period, in radians. We assume the desired signal is given by

$$d(k) = B\, e^{j[\omega_0(k-D)+\varphi]} \tag{2.56}$$

where D is a known delay. We also assume that $\varphi \in \mathcal{U}(0, 2\pi)$, and w is white noise with zero mean and variance σ_w^2, uncorrelated with φ. The autocorrelation function of x and the

cross-correlation between d and x are given by

$$r_x(n) = A^2 e^{j\omega_0 n} + \sigma_w^2 \delta_n \tag{2.57}$$

$$r_{dx}(n) = AB\, e^{j\omega_0 (n-D)} \tag{2.58}$$

For a Wiener filter with N coefficients, the autocorrelation matrix $\mathbf{R}$ and the vector $\mathbf{p}$ have the following structure:

$$\mathbf{R} = \begin{bmatrix} A^2 + \sigma_w^2 & A^2 e^{-j\omega_0} & \dots & A^2 e^{-j\omega_0 (N-1)} \\ A^2 e^{j\omega_0} & A^2 + \sigma_w^2 & \dots & A^2 e^{-j\omega_0 (N-2)} \\ \vdots & \vdots & \ddots & \dots \\ A^2 e^{j\omega_0 (N-1)} & A^2 e^{j\omega_0 (N-2)} & \dots & A^2 + \sigma_w^2 \end{bmatrix} \tag{2.59}$$

$$\mathbf{p} = \begin{bmatrix} 1 \\ e^{j\omega_0} \\ \vdots \\ e^{j\omega_0 (N-1)} \end{bmatrix} AB\, e^{-j\omega_0 D} \tag{2.60}$$

Defining

$$\mathbf{E}^T(\omega) = [1, e^{j\omega}, \dots, e^{j\omega(N-1)}] \tag{2.61}$$

we can express $\mathbf{R}$ and $\mathbf{p}$ as

$$\mathbf{R} = \sigma_w^2 \mathbf{I} + A^2 \mathbf{E}(\omega_0)\mathbf{E}^H(\omega_0) \tag{2.62}$$

$$\mathbf{p} = AB e^{-j\omega_0 D} \mathbf{E}(\omega_0) \tag{2.63}$$

Observing that $\mathbf{E}^H(\omega)\mathbf{E}(\omega) = N$, the inverse of $\mathbf{R}$ is given by

$$\mathbf{R}^{-1} = \frac{1}{\sigma_w^2} \left[\mathbf{I} - \frac{A^2}{\sigma_w^2 + NA^2} \mathbf{E}(\omega_0)\mathbf{E}^H(\omega_0) \right] \tag{2.64}$$

Hence, using (2.34):

$$\mathbf{c}_{opt} = \frac{AB e^{-j\omega_0 D}}{\sigma_w^2 + NA^2} \mathbf{E}(\omega_0) = \frac{B}{A} \frac{\Lambda e^{-j\omega_0 D}}{1 + N\Lambda} \mathbf{E}(\omega_0) \tag{2.65}$$

where $\Lambda = A^2/\sigma_w^2$ is the signal-to-noise ratio. From (2.40) the minimum value of the cost function J is given by

$$J_{min} = B^2 - AB e^{j\omega_0 D} \mathbf{E}^H(\omega_0)\mathbf{E}(\omega_0) \frac{AB e^{-j\omega_0 D}}{\sigma_w^2 + NA^2} = \frac{B^2}{1 + N\Lambda} \tag{2.66}$$

Defining $\omega = 2\pi f T_c$, the optimum filter frequency response is given by

$$\begin{aligned} C_{opt}(e^{j\omega}) &= \sum_{i=0}^{N-1} c_{opt,i} e^{-j\omega i} \\ &= \mathbf{E}^H(\omega)\mathbf{c}_{opt} \\ &= \frac{B}{A}\frac{\Lambda e^{-j\omega_0 D}}{1+N\Lambda}\sum_{i=0}^{N-1} e^{-j(\omega-\omega_0)i} \end{aligned} \tag{2.67}$$

that is,

$$C_{opt}(e^{j\omega}) = \begin{cases} \dfrac{B}{A}\dfrac{N\Lambda e^{-j\omega_0 D}}{1+N\Lambda} & \omega = \omega_0 \\ \dfrac{B}{A}\dfrac{\Lambda e^{-j\omega_0 D}}{1+N\Lambda}\dfrac{1-e^{-j(\omega-\omega_0)N}}{1-e^{-j(\omega-\omega_0)}} & \omega \neq \omega_0 \end{cases} \tag{2.68}$$

We observe that, for $\Lambda \gg 1$,

1. J_{min} becomes negligible;
2. $\mathbf{c}_{opt} = \dfrac{B}{AN} e^{-j\omega_0 D}\mathbf{E}(\omega_0)$;
3. $|C_{opt}(e^{j\omega_0})| = \dfrac{B}{A}$.

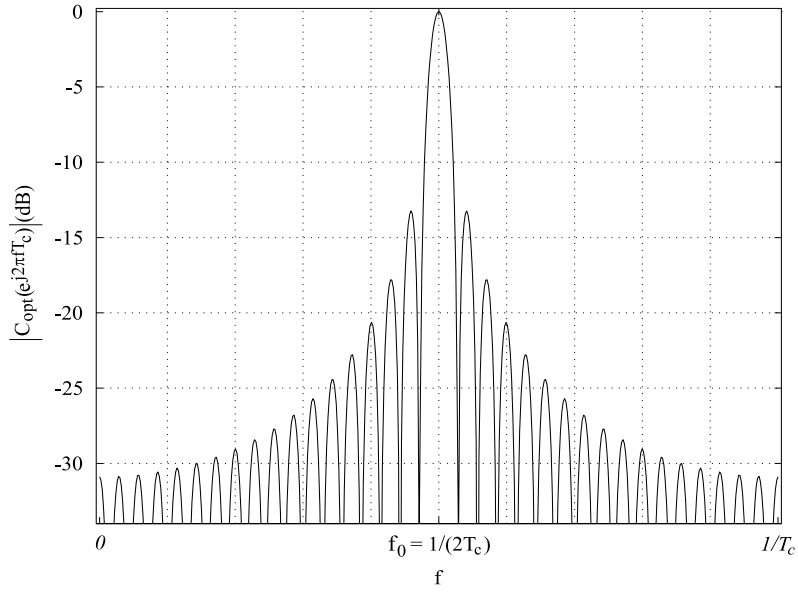


Figure 2.6. Magnitude of $C_{opt}(e^{j2\pi fT_c})$ given by (2.68) for $f_0 T_c = 1/2$, $B = A$, $\Lambda = 30$ dB, and $N = 35$.

Conversely, for $\Lambda \to 0$, i.e. when the power of the useful signal is negligible with respect to the power of the additive noise, it results in

1. $J_{min} = B^2$;
2. $\mathbf{c}_{opt} = \mathbf{0}$;
3. $|C_{opt}(e^{j\omega_0})| = 0$.

Indeed, as the signal-to-noise ratio vanishes the best choice is to set the output y to zero. The plot of $|C_{opt}(e^{j2\pi f T_c})|$ is given in Figure 2.6.

2.2 Linear prediction

The Wiener theory considered in the previous section has an important application to the solution of the following problem. Let x be a discrete-time WSS random process with zero mean; prediction consists in estimating a "future" value of the process starting from a set of known "past" values. In particular, let us define the vector

$$\mathbf{x}^T(k-1) = [x(k-1), x(k-2), \ldots, x(k-N)] \tag{2.69}$$

The *one-step forward predictor of order N*, given $\mathbf{x}^T(k-1)$, attempts to estimate the value of $x(k)$. There exists also the problem of predicting $x(k-N)$, given the values of $x(k-N+1), \ldots, x(k)$. In this case, the system is called the *one-step backward predictor of order N*.

Forward linear predictor

The estimate of $x(k)$ is expressed as a linear combination of the preceding N samples:

$$\hat{x}(k \mid \mathbf{x}(k-1)) = \sum_{i=1}^{N} c_i x(k-i) \tag{2.70}$$

The block diagram of the linear predictor is represented in Figure 2.7.

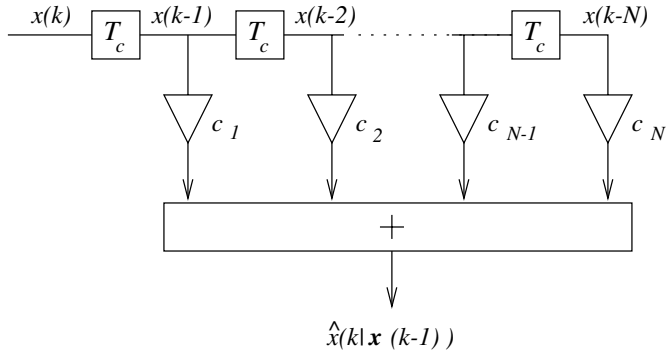


Figure 2.7. Linear predictor of order *N*.

This estimate will be subject to a *forward prediction error* given by

$$\begin{aligned} f_N(k) &= x(k) - \hat{x}(k \mid \mathbf{x}(k-1)) \\ &= x(k) - \sum_{i=1}^{N} c_i x(k-i) \end{aligned} \tag{2.71}$$

Optimum predictor coefficients

If we adopt the criterion of minimizing the mean-square prediction error,

$$J = E[|f_N(k)|^2] \tag{2.72}$$

to determine the predictor coefficients, we can use the optimization results according to Wiener. We recall the following definitions.

1. *Desired signal*

$$d(k) = x(k) \tag{2.73}$$

2. *Filter input vector* (defined by (2.69))

$$\mathbf{x}^T(k-1) \tag{2.74}$$

3. *Cost function J* (given by (2.72)).

Then it turns out:

$$\sigma_d^2 = E[x(k)x^*(k)] = \sigma_x^2 = \mathrm{r}_x(0) \tag{2.75}$$

$$E[\mathbf{x}^*(k-1)\mathbf{x}^T(k-1)] = \mathbf{R}_N \tag{2.76}$$

with $\mathbf{R}_N$ $N \times N$ correlation matrix, and

$$\mathbf{p} = E[d(k)\mathbf{x}^*(k-1)] = E[x(k)\mathbf{x}^*(k-1)] = \begin{bmatrix} \mathrm{r}_x(1) \\ \mathrm{r}_x(2) \\ \vdots \\ \mathrm{r}_x(N) \end{bmatrix} = \mathbf{r}_N \tag{2.77}$$

Applying (2.32), the optimum coefficients satisfy the equation

$$\mathbf{R}_N \mathbf{c}_{opt} = \mathbf{r}_N \tag{2.78}$$

Moreover, from (2.40) we get the minimum value of the cost function J,

$$J_{min} = J_N = \mathrm{r}_x(0) - \mathbf{r}_N^H \mathbf{c}_{opt} \tag{2.79}$$

We can combine the latter two equations to get an augmented form of the W-H equation for the linear predictor:

$$\begin{bmatrix} \mathrm{r}_x(0) & \mathbf{r}_N^H \\ \mathbf{r}_N & \mathbf{R}_N \end{bmatrix} \begin{bmatrix} 1 \\ -\mathbf{c}_{opt} \end{bmatrix} = \begin{bmatrix} J_N \\ \mathbf{0}_N \end{bmatrix} \tag{2.80}$$

where $\mathbf{0}_N$ is the column vector of N zeros.

Forward "prediction error filter"

We determine the filter that gives the forward linear prediction error f_N. For an optimum predictor,

$$f_N(k) = x(k) - \sum_{i=1}^{N} c_{opt,i}\, x(k-i) \tag{2.81}$$

We introduce the vector

$$\mathtt{a}'_{i,N} = \begin{cases} 1 & i = 0 \\ -c_{opt,i} & i = 1, 2, \ldots, N \end{cases} \tag{2.82}$$

which can be rewritten as

$$\mathbf{a}'_N = \begin{bmatrix} 1 \\ \mathbf{a} \end{bmatrix} \tag{2.83}$$

where $\mathbf{a} = -\mathbf{c}_{opt}$. Substituting (2.82) in (2.81) and taking care to extend the equation also to $i = 0$, we obtain

$$f_N(k) = \sum_{i=0}^{N} \mathtt{a}'_{i,N}\, x(k-i) \tag{2.84}$$

as shown in Figure 2.8.

The coefficients $\mathbf{a}'^{T}_N = [\mathtt{a}'_{0,N}, \mathtt{a}'_{1,N}, \ldots, \mathtt{a}'_{N,N}]$ are directly obtained by substituting (2.83) in (2.80):

$$\mathbf{R}_{N+1}\mathbf{a}'_N = \begin{bmatrix} J_N \\ \mathbf{0}_N \end{bmatrix} \tag{2.85}$$

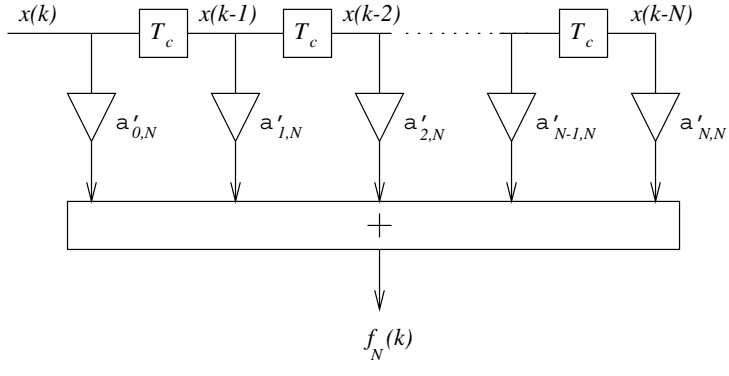


Figure 2.8. Forward prediction error filter.

With a similar procedure, we can derive the filter that gives the *backward linear prediction error*,

$$b_N(k) = x(k-N) - \sum_{i=1}^{N} g_i x(k-i+1) \tag{2.86}$$

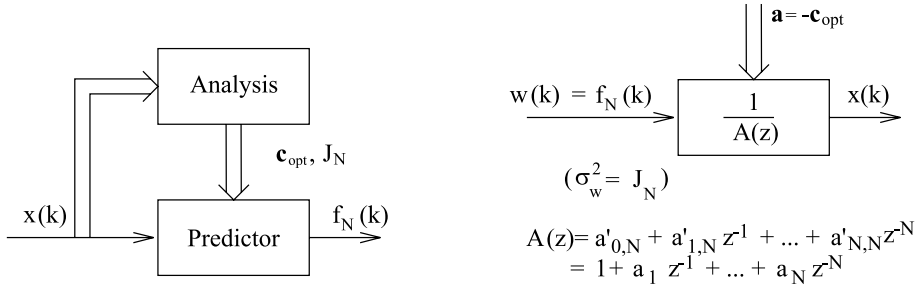
It can be shown that the optimum coefficients are given by

$$\mathbf{g}_{opt} = \mathbf{c}_{opt}^{B*} \tag{2.87}$$

where ^{B} is the *backward* operator that orders the elements of a vector backward, from the last to the first (see page 27).

Relation between linear prediction and AR models

The similarity of (2.78) with the Yule–Walker equation (1.537) allows us to state what follows: given an AR process x of order N, the optimum prediction coefficients $\mathbf{c}_{opt}$ coincide with the parameters $-\mathbf{a}$ of the process and, moreover, $J_N = \sigma_w^2$. Actually, for $\mathbf{c}_{opt} = -\mathbf{a}$, comparing (2.81) with (1.518) we find $f_N(k) = w(k)$, that is, the prediction error f_N coincides with white noise having statistical power J_N. In general, for a process x, if the order of the prediction error filter is large enough, we can observe that this filter has *whitening* properties, in that it is capable of removing the correlated signal component that is present at the input, producing at the output only the uncorrelated or "white" component. Moreover, while *prediction* can be interpreted as the *analysis* of an AR process, the *AR model* may be regarded as the *synthesis* of the process. As illustrated in Figure 2.9, given a realization of the process $\{x(k)\}$, by estimating the autocorrelation sequence over a suitable observation window, the parameters $\mathbf{c}_{opt}$ and J_N can be determined. Using the predictor then we determine the prediction error $\{f_N(k)\}$. To reproduce $\{x(k)\}$, an all-pole filter with coefficients $\mathbf{a}_N'^T = [1, -\mathbf{c}_{opt}]$, having white noise $\{w(k) = f_N(k)\}$ of power $\sigma_w^2 = J_N$ as input, can be used.



(a) Analysis (b) Synthesis

Figure 2.9. Analysis and synthesis of AR (N) processes.

First and second order solutions

We give below formulae to compute the predictor filter coefficients and prediction error filter coefficients for orders $N = 1$ and $N = 2$. These results extend to the complex case the formulae obtained in Section 1.12.2.

- $N = 1$. From

$$\begin{bmatrix} \mathtt{r}_x(0) & \mathtt{r}_x^*(1) \\ \mathtt{r}_x(1) & \mathtt{r}_x(0) \end{bmatrix} \begin{bmatrix} \mathtt{a}'_{0,1} \\ \mathtt{a}'_{1,1} \end{bmatrix} = \begin{bmatrix} J_1 \\ 0 \end{bmatrix} \tag{2.88}$$

it results

$$\begin{cases} \mathtt{a}'_{0,1} = \dfrac{J_1}{\Delta \mathtt{r}} \mathtt{r}_x(0) \\ \mathtt{a}'_{1,1} = -\dfrac{J_1}{\Delta \mathtt{r}} \mathtt{r}_x(1) \end{cases} \tag{2.89}$$

where

$$\Delta \mathtt{r} = \begin{vmatrix} \mathtt{r}_x(0) & \mathtt{r}_x^*(1) \\ \mathtt{r}_x(1) & \mathtt{r}_x(0) \end{vmatrix} = \mathtt{r}_x^2(0) - |\mathtt{r}_x(1)|^2 \tag{2.90}$$

As $\mathtt{a}'_{0,1} = 1$, it turns out

$$\begin{cases} J_1 = \dfrac{\Delta \mathtt{r}}{\mathtt{r}_x(0)} \\ \mathtt{a}'_{1,1} = -\dfrac{\mathtt{r}_x(1)}{\mathtt{r}_x(0)} \end{cases} \Rightarrow \begin{cases} c_{opt,1} = -\mathtt{a}'_{1,1} = \rho(1) \\ \dfrac{J_1}{\mathtt{r}_x(0)} = 1 - |\rho(1)|^2 \end{cases} \tag{2.91}$$

- $N = 2$.

$$\begin{cases} \mathtt{a}'_{1,2} = -\dfrac{\mathtt{r}_x(1)\mathtt{r}_x(0) - \mathtt{r}_x^*(1)\mathtt{r}_x(2)}{\mathtt{r}_x^2(0) - |\mathtt{r}_x(1)|^2} \\ \mathtt{a}'_{2,2} = -\dfrac{\mathtt{r}_x(0)\mathtt{r}_x(2) - \mathtt{r}_x^2(1)}{\mathtt{r}_x^2(0) - |\mathtt{r}_x(1)|^2} \end{cases} \Rightarrow \begin{cases} c_{opt,1} = \dfrac{\rho(1) - \rho^*(1)\rho(2)}{1 - |\rho(1)|^2} \\ c_{opt,2} = \dfrac{\rho(2) - \rho^2(1)}{1 - |\rho(1)|^2} \end{cases} \tag{2.92}$$

and

$$\frac{J_2}{\mathtt{r}_x(0)} = \frac{1 - 2|\rho(1)|^2 + \rho^{*2}(1)\rho(2) + |\rho(1)|^2\rho^*(2) - |\rho(2)|^2}{1 - |\rho(1)|^2} \tag{2.93}$$

We note that in the above equations $\rho(n)$ is the correlation coefficient of x, introduced in (1.540).

2.2.1 The Levinson–Durbin algorithm

The Levinson–Durbin algorithm (LDA) yields the solution of matrix equations like (2.85), in which $\mathbf{R}_{N+1}$ is positive definite, Hermitian, and Toeplitz, with a computational complexity proportional to N^2, instead of N^3 as happens with algorithms that make use of the inverse matrix. In the case of real signals, $\mathbf{R}_{N+1}$ is symmetric and the computational complexity of the Delsarte–Genin algorithm (DGA), given in Section 2.2.2, is halved with respect to that of LDA. Here, we report a step-by-step description of the LDA:

1. **Initialization**. We set:

$$J_0 = \mathtt{r}_x(0) \tag{2.94}$$

$$\Delta_0 = \mathtt{r}_x(1) \tag{2.95}$$

2. **n-th iteration**, $n = 1, 2, \ldots, N$. We calculate

$$\mathcal{C}_n = -\frac{\Delta_{n-1}}{J_{n-1}} \tag{2.96}$$

$$\mathbf{a}'_n = \begin{bmatrix} \mathbf{a}'_{n-1} \\ 0 \end{bmatrix} + \mathcal{C}_n \begin{bmatrix} 0 \\ \mathbf{a}'^{B*}_{n-1} \end{bmatrix} \tag{2.97}$$

Then (2.97) corresponds to the scalar equations:

$$\mathtt{a}'_{k,n} = \mathtt{a}'_{k,n-1} + \mathcal{C}_n \mathtt{a}'^{*}_{n-k,n-1} \qquad k = 0, 1, \ldots, n \tag{2.98}$$

with $\mathtt{a}'_{0,n-1} = 1$ and $\mathtt{a}'_{n,n-1} = 0$. Moreover,

$$\Delta_n = (\mathbf{r}^B_{n+1})^T \mathbf{a}'_n \tag{2.99}$$

$$J_n = J_{n-1}(1 - |\mathcal{C}_n|^2) \tag{2.100}$$

We now interpret the physical meaning of the parameters in the algorithm. J_n represents the *statistical power of the forward prediction error* at the n-th iteration:

$$J_n = E[|f_n(k)|^2] \tag{2.101}$$

It results in

$$0 \leq J_n \leq J_{n-1} \qquad n \geq 1 \tag{2.102}$$

with

$$J_0 = \mathtt{r}_x(0) \tag{2.103}$$

and

$$J_N = J_0 \prod_{n=1}^{N} (1 - |\mathcal{C}_n|^2) \tag{2.104}$$

The following relation holds for Δ_n:

$$\Delta_{n-1} = E[f_{n-1}(k)b^*_{n-1}(k-1)] \tag{2.105}$$

In other words, Δ_n can be interpreted as the cross-correlation between the forward linear prediction error and the backward linear prediction error delayed by one sample. $\mathcal{C}_n$ satisfies the following property:

$$\mathcal{C}_n = \mathtt{a}'_{n,n} \tag{2.106}$$

Finally, by substitution, from (2.96), along with (2.101) and (2.105), we get

$$\mathcal{C}_n = -\frac{E[f_{n-1}(k)b^*_{n-1}(k-1)]}{E[|f_{n-1}(k)|^2]} \tag{2.107}$$

and, noting that $E[|f_{n-1}(k)|^2] = E[|b_{n-1}(k)|^2] = E[|b_{n-1}(k-1)|^2]$, from (2.107) we have

$$|\mathcal{C}_n| \leq 1 \tag{2.108}$$

The coefficients $\{\mathcal{C}_n\}$ are called *reflection coefficients* or *partial correlation coefficients* (PARCOR).

Lattice filters

We have just described the Levinson–Durbin algorithm. Its analysis permits us to implement the prediction error filter via a modular structure. Defining

$$\mathbf{x}_{n+1}(k) = [x(k), \ldots, x(k-n)]^T \tag{2.109}$$

we can write:

$$\mathbf{x}_{n+1}(k) = \begin{bmatrix} \mathbf{x}_n(k) \\ x(k-n) \end{bmatrix} = \begin{bmatrix} x(k) \\ \mathbf{x}_n(k-1) \end{bmatrix} \tag{2.110}$$

We recall the relation for forward and backward linear prediction error filters of order n:

$$\begin{cases} f_n(k) = \mathtt{a}'_{0,n}x(k) + \cdots + \mathtt{a}'_{n,n}x(k-n) = \mathbf{a}'^T_n \mathbf{x}_{n+1}(k) \\ b_n(k) = \mathtt{a}'^*_{n,n}x(k) + \cdots + \mathtt{a}'^*_{0,n}x(k-n) = \mathbf{a}'^{BH}_n \mathbf{x}_{n+1}(k) \end{cases} \tag{2.111}$$

From (2.97) we obtain

$$\begin{aligned} f_n(k) &= \begin{bmatrix} \mathbf{a}'_{n-1} \\ 0 \end{bmatrix}^T \mathbf{x}_{n+1}(k) + \mathcal{C}_n \begin{bmatrix} 0 \\ \mathbf{a}'^{B*}_{n-1} \end{bmatrix}^T \mathbf{x}_{n+1}(k) \\ &= f_{n-1}(k) + \mathcal{C}_n b_{n-1}(k-1) \end{aligned} \tag{2.112}$$

By a similar procedure we also find

$$b_n(k) = b_{n-1}(k-1) + \mathcal{C}^*_n f_{n-1}(k) \tag{2.113}$$

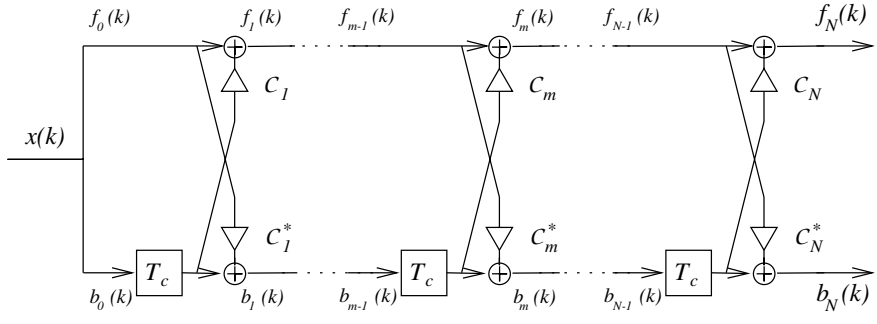


Figure 2.10. Lattice filter.

Finally, taking into account the initial conditions,

$$f_0(k) = b_0(k) = x(k) \quad \text{and} \quad \mathtt{a}'_{0,0} = 1 \tag{2.114}$$

the block diagram of Figure 2.10 is obtained, in which the output is given by f_N.

We list the following fundamental properties.

1. The optimum coefficients $\mathcal{C}_n$, $n = 1, \ldots, N$, are independent of the order of the filter; therefore one can change N without having to re-calculate all the coefficients. This property is useful if the filter length is unknown and must be estimated.

2. If the conditions $|\mathcal{C}_n| \leq 1$, $n = 1, \ldots, N$, are verified, the filter is minimum phase.

3. The lattice filters are quite insensitive to coefficient quantization.

Observation 2.2
From the above property 2 and (2.108), we find that all predictor error filters are *minimum phase*.

2.2.2 The Delsarte–Genin algorithm

In the case of real signals, the DGA, also known as the *split Levinson algorithm* [3], further reduces the number of operations with respect to the LDA, at least for $N \geq 10$.[3] Here is the step-by-step description.

1. **Initialization**. We set

$$\mathbf{v}_0 = 1 \qquad \beta_0 = \mathtt{r}_x(0) \qquad \gamma_0 = \mathtt{r}_x(1) \tag{2.115}$$

$$\mathbf{v}_1 = [1, 1]^T \qquad \beta_1 = \mathtt{r}_x(0) + \mathtt{r}_x(1) \qquad \gamma_1 = \mathtt{r}_x(1) + \mathtt{r}_x(2) \tag{2.116}$$

[3] Faster algorithms, with a complexity proportional to $N(\log N)^2$, have been proposed by Kumar [4].

2. **n-th iteration**, $n = 2, \ldots, N$. We compute

$$\alpha_n = \frac{(\beta_{n-1} - \gamma_{n-1})}{(\beta_{n-2} - \gamma_{n-2})} \tag{2.117}$$

$$\beta_n = 2\beta_{n-1} - \alpha_n \beta_{n-2} \tag{2.118}$$

$$\mathbf{v}_n = \begin{bmatrix} \mathbf{v}_{n-1} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{v}_{n-1} \end{bmatrix} - \alpha_n \begin{bmatrix} 0 \\ \mathbf{v}_{n-2} \\ 0 \end{bmatrix} \tag{2.119}$$

$$\gamma_n = \mathbf{r}_{n+1}^T \mathbf{v}_n = (\mathtt{r}_x(1) + \mathtt{r}_x(n+1)) + [\mathbf{v}_n]_2 (\mathtt{r}_x(2) + \mathtt{r}_x(n)) + \cdots \tag{2.120}$$

$$\lambda_n = \frac{\beta_n}{\beta_{n-1}} \tag{2.121}$$

$$\mathbf{a}'_n = \mathbf{v}_n - \lambda_n \begin{bmatrix} 0 \\ \mathbf{v}_{n-1} \end{bmatrix} \tag{2.122}$$

$$J_n = \beta_n - \lambda_n \gamma_{n-1} \tag{2.123}$$

$$\mathcal{C}_n = 1 - \lambda_n \tag{2.124}$$

We note that (2.120) exploits the symmetry of the vector $\mathbf{v}_n$; in particular it is $[\mathbf{v}_n]_1 = [\mathbf{v}_n]_{n+1} = 1$.

2.3 The least squares (LS) method

The Wiener filter will prove to be a powerful *analytical* tool in various applications, one of which is indeed prediction. However, from a practical point of view, often only realizations of the processes $\{x(k)\}$ and $\{d(k)\}$ are available. Therefore to get the solution it is necessary to determine estimates of $\mathtt{r}_x$ and $\mathtt{r}_{dx}$, and various alternatives emerge. Two possible methods are: 1) the *autocorrelation method*, in which from the estimate of $\mathtt{r}_x$ we construct $\mathbf{R}$ as a Toeplitz correlation matrix, and 2) the *covariance method*, in which we estimate each element of $\mathbf{R}$ by (2.130). In this case the matrix is only Hermitian and the solution that we are going to illustrate is of the LS type [1, 2].

We reconsider the problem of Section 2.1, introducing a new cost function. Based on the observation of the sequences

$$\{x(k)\} \qquad k = 0, \ldots, K-1 \quad \text{and} \quad \{d(k)\} \qquad k = 0, \ldots, K-1 \tag{2.125}$$

and of the error

$$e(k) = d(k) - y(k) \tag{2.126}$$

where $y(k)$ is given by (2.1), according to the least squares method the optimum filter coefficients yield the *minimum of the sum of the squared errors*:

$$\min_{\{c_n\}, n=0,1,\ldots,N-1} \mathcal{E} \tag{2.127}$$

where

$$\mathcal{E} = \sum_{k=N-1}^{K-1} |e(k)|^2 \tag{2.128}$$

Note that in the LS method a time average is substituted for the expectation (2.3), which gives the MSE.

Data windowing

In matrix notation, the output $\{y(k)\}$, $k = N-1, \ldots, K-1$, given by (2.1), can be expressed as

$$\begin{bmatrix} y(N-1) \\ y(N) \\ \vdots \\ y(K-1) \end{bmatrix} = \underbrace{\begin{bmatrix} x(N-1) & x(N-2) & \ldots & x(0) \\ x(N) & x(N-1) & \ldots & x(1) \\ \vdots & \vdots & \ddots & \vdots \\ x(K-1) & x(K-2) & \ldots & x(K-N) \end{bmatrix}}_{\text{data matrix } \mathbf{T}} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{N-1} \end{bmatrix} \tag{2.129}$$

In (2.129) we note that the input data sequence actually used goes from $x(0)$ to $x(K-1)$. Other choices are possible for the input data window. The case examined is called the *covariance method* and the data matrix $\mathbf{T}$, defined by (2.129), is $L \times N$ where $L = K-N+1$.

Matrix formulation

We define

$$\Phi(i,n) = \sum_{k=N-1}^{K-1} x^*(k-i)\, x(k-n) \qquad i,n = 0, 1, \ldots, N-1 \tag{2.130}$$

$$\vartheta(n) = \sum_{k=N-1}^{K-1} d(k)\, x^*(k-n) \qquad n = 0, 1, \ldots, N-1 \tag{2.131}$$

Using (1.478) for an *unbiased* estimate of the correlation, the following identities hold:

$$\Phi(i,n) = (K-N+1)\hat{r}_x(i-n) \tag{2.132}$$

$$\vartheta(n) = (K-N+1)\hat{r}_{dx}(n) \tag{2.133}$$

in which the values of $\Phi(i,n)$ depend on both indices (i,n) and not only upon their difference, especially if K is not very large. We give some definitions:

1. *Energy of* $\{d(k)\}$

$$\mathcal{E}_d = \sum_{k=N-1}^{K-1} |d(k)|^2 \tag{2.134}$$

2. *Cross-correlation vector between d and* $\mathbf{x}$

$$\boldsymbol{\vartheta}^T = [\vartheta(0), \vartheta(1), \ldots, \vartheta(N-1)] \tag{2.135}$$

3. *Input autocorrelation matrix*

$$\boldsymbol{\Phi} = \begin{bmatrix} \Phi(0,0) & \Phi(0,1) & \ldots & \Phi(0,N-1) \\ \Phi(1,0) & \Phi(1,1) & \ldots & \Phi(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi(N-1,0) & \Phi(N-1,1) & \ldots & \Phi(N-1,N-1) \end{bmatrix} \tag{2.136}$$

Then the cost function can be written as

$$\mathcal{E} = \mathcal{E}_d - \mathbf{c}^H \boldsymbol{\vartheta} - \boldsymbol{\vartheta}^H \mathbf{c} + \mathbf{c}^H \boldsymbol{\Phi} \mathbf{c} \tag{2.137}$$

Correlation matrix $\boldsymbol{\Phi}$

$\boldsymbol{\Phi}$ is the time average of $\mathbf{x}^*(k)\mathbf{x}^T(k)$, i.e.

$$\boldsymbol{\Phi} = \sum_{k=N-1}^{K-1} \mathbf{x}^*(k)\mathbf{x}^T(k) \tag{2.138}$$

Properties of $\boldsymbol{\Phi}$.

1. $\boldsymbol{\Phi}$ is Hermitian.
2. $\boldsymbol{\Phi}$ is positive semi-definite.
3. Eigenvalues of $\boldsymbol{\Phi}$ are real and non-negative.
4. $\boldsymbol{\Phi}$ can be written as

$$\boldsymbol{\Phi} = \mathbf{T}^H \mathbf{T} \tag{2.139}$$

with $\mathbf{T}$ input data matrix defined by (2.129). We note that the matrix $\mathbf{T}$ is Toeplitz.

Determination of the optimum filter coefficients

By analogy of (2.137) with (2.17), the gradient of (2.137) is given by

$$\nabla_{\mathbf{c}} \mathcal{E} = 2(\boldsymbol{\Phi}\mathbf{c} - \boldsymbol{\vartheta}) \tag{2.140}$$

Then the vector of optimum coefficients based on the LS method, $\mathbf{c}_{ls}$, satisfies the *normal equation*

$$\boldsymbol{\Phi}\mathbf{c}_{ls} = \boldsymbol{\vartheta} \tag{2.141}$$

If $\mathbf{\Phi}^{-1}$ exists, the solution to (2.141) is given by

$$\mathbf{c}_{ls} = \mathbf{\Phi}^{-1}\boldsymbol{\vartheta} \tag{2.142}$$

In the solution of the LS problem, the equation (2.141) corresponds to the Wiener–Hopf equation (2.32). As for an ergodic process (2.132) yields:

$$\frac{1}{K-N+1}\mathbf{\Phi} \xrightarrow[K\to\infty]{} \mathbf{R} \tag{2.143}$$

and

$$\frac{1}{K-N+1}\boldsymbol{\vartheta} \xrightarrow[K\to\infty]{} \mathbf{p} \tag{2.144}$$

We find that the LS solution tends toward the Wiener solution for sufficiently large K, that is

$$\mathbf{c}_{ls} \xrightarrow[K\to\infty]{} \mathbf{c}_{opt} \tag{2.145}$$

In other words, for $K \to \infty$ the *covariance method* gives the same solution as the *autocorrelation method*.

In scalar notation, (2.141) becomes a system of N equations in N unknowns:

$$\sum_{i=0}^{N-1} \Phi(n,i)c_{ls,i} = \vartheta(n) \qquad n = 0, 1, \ldots, N-1 \tag{2.146}$$

2.3.1 The principle of orthogonality

From (2.128), taking the gradient with respect to c_n, we have

$$\begin{aligned}\nabla_{c_n}\mathcal{E} &= \nabla_{c_n,I}\mathcal{E} + j\nabla_{c_n,Q}\mathcal{E} \\ &= \sum_{k=N-1}^{K-1} [-x^*(k-n)e(k) - x(k-n)e^*(k) \\ &\quad + j(jx^*(k-n)e(k) - jx(k-n)e^*(k))] \\ &= -2\sum_{k=N-1}^{K-1} x^*(k-n)e(k)\end{aligned} \tag{2.147}$$

If we denote with $\{e_{min}(k)\}$ the estimation error found with the optimum coefficient values, $\mathbf{c}_{ls}$, then the optimum coefficients must satisfy the conditions

$$\sum_{k=N-1}^{K-1} e_{min}(k)x^*(k-n) = 0 \qquad n = 0, 1, \ldots, N-1 \tag{2.148}$$

which represent the time-average version of the *statistical orthogonality principle* (2.36). Moreover, being $y(k)$ a linear combination of $\{x(k-n)\}$, $n = 0, 1, \ldots, N-1$, we have

$$\sum_{k=N-1}^{K-1} e_{min}(k) y^*(k) = 0 \tag{2.149}$$

Equation (2.149) expresses the fundamental result: the *optimum filter output* sequence is orthogonal to the *minimum estimation error* sequence.

Expressions of the minimum cost function

Substituting (2.141) in (2.137), the minimum cost function can be written as

$$\mathcal{E}_{min} = \mathcal{E}_d - \boldsymbol{\vartheta}^H \mathbf{c}_{ls} \tag{2.150}$$

An alternative expression to $\mathcal{E}_{min}$ uses the energy of the output sequence:

$$\mathcal{E}_y = \sum_{k=N-1}^{K-1} |y(k)|^2 = \mathbf{c}^H \boldsymbol{\Phi} \mathbf{c} \tag{2.151}$$

observing (2.130). Note that for $\mathbf{c} = \mathbf{c}_{ls}$ we have

$$d(k) = y(k) + e_{min}(k) \tag{2.152}$$

then, because of the orthogonality (2.149) between y and e_{min}, it follows that

$$\mathcal{E}_d = \mathcal{E}_y + \mathcal{E}_{min} \tag{2.153}$$

from which, substituting (2.141) in (2.151), we get

$$\mathcal{E}_{min} = \mathcal{E}_d - \mathcal{E}_y \tag{2.154}$$

where $\mathcal{E}_y = \mathbf{c}_{ls}^H \boldsymbol{\vartheta}$.

The normal equation using the $\mathbf{T}$ *matrix*

Defining the *vector of desired samples*

$$\mathbf{d}^T = [d(N-1), d(N), \ldots, d(K-1)] \tag{2.155}$$

from the definition (2.131) of $\vartheta(n)$ we get

$$\begin{bmatrix} \vartheta(0) \\ \vartheta(1) \\ \vdots \\ \vartheta(N-1) \end{bmatrix} = \begin{bmatrix} x^*(N-1) & x^*(N) & \ldots & x^*(K-1) \\ x^*(N-2) & x^*(N-1) & \ldots & x^*(K-2) \\ \vdots & \vdots & \ddots & \vdots \\ x^*(0) & x^*(2) & \ldots & x^*(K-N) \end{bmatrix} \begin{bmatrix} d(N-1) \\ d(N) \\ \vdots \\ d(K-1) \end{bmatrix} \tag{2.156}$$

that is

$$\boldsymbol{\vartheta} = \mathbf{T}^H \mathbf{d} \tag{2.157}$$

Thus, using the (2.139) and (2.157), the normal equation (2.141) becomes

$$\mathbf{T}^H \mathbf{T} \mathbf{c}_{ls} = \mathbf{T}^H \mathbf{d} \tag{2.158}$$

Associated with system (2.158), it is useful to introduce the system of equations for the minimization of $\mathcal{E}$,

$$\mathbf{T}\mathbf{c} = \mathbf{d} \tag{2.159}$$

From (2.158), if $(\mathbf{T}^H\mathbf{T})^{-1}$ exists, the solution is

$$\mathbf{c}_{ls} = (\mathbf{T}^H\mathbf{T})^{-1}\,\mathbf{T}^H\mathbf{d} \tag{2.160}$$

and correspondingly (2.150) becomes

$$\mathcal{E}_{min} = \mathbf{d}^H\mathbf{d} - \mathbf{d}^H\mathbf{T}(\mathbf{T}^H\mathbf{T})^{-1}\mathbf{T}^H\mathbf{d} \tag{2.161}$$

We note how both formulae (2.160) and (2.161) depend only on the desired signal samples and input samples. Moreover, the solution $\mathbf{c}$ is unique only if the columns of $\mathbf{T}$ are linearly independent, that is the case of non-singular $\mathbf{T}^H\mathbf{T}$. This requires at least $K - N + 1 > N$, that is the system of equations (2.159) must be overdetermined with more equations than unknowns.

Geometric interpretation: the projection operator

In general, from (2.129) the vector of filter output samples

$$\mathbf{y}^T = [y(N-1), y(N), \ldots, y(K-1)] \tag{2.162}$$

can be related to the input data matrix $\mathbf{T}$ as

$$\mathbf{y} = \mathbf{T}\mathbf{c} \tag{2.163}$$

This relation will still be valid for $\mathbf{c} = \mathbf{c}_{ls}$, and from (2.160) we get

$$\mathbf{y} = \mathbf{T}\mathbf{c}_{ls} = \mathbf{T}(\mathbf{T}^H\mathbf{T})^{-1}\mathbf{T}^H\mathbf{d} \tag{2.164}$$

Correspondingly, the estimation vector error is given by

$$\mathbf{e}_{min} = \mathbf{d} - \mathbf{y} \tag{2.165}$$

The matrix $\boldsymbol{\mathcal{O}} = \mathbf{T}(\mathbf{T}^H\mathbf{T})^{-1}\mathbf{T}^H$ can be thought of as a projection operator defined on the space generated by the columns of $\mathbf{T}$. Let $\mathbf{I}$ be the identity matrix: the difference

$$\boldsymbol{\mathcal{O}}_\perp = \mathbf{I} - \boldsymbol{\mathcal{O}} = \mathbf{I} - \mathbf{T}(\mathbf{T}^H\mathbf{T})^{-1}\mathbf{T}^H \tag{2.166}$$

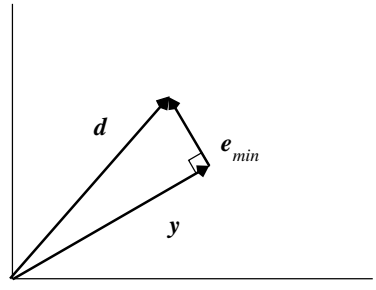


Figure 2.11. Relations among vectors in the LS minimization.

is the complementary projection operator, orthogonal to $\mathcal{O}$. In fact, from (2.164)

$$\mathbf{y} = \mathcal{O}\mathbf{d} \tag{2.167}$$

and from (2.165)

$$\mathbf{e}_{min} = \mathbf{d} - \mathbf{y} = \mathcal{O}_{\perp}\mathbf{d} \tag{2.168}$$

where $\mathbf{e}_{min} \perp \mathbf{y}$ (see (2.149)). Moreover, (2.161) can be written as

$$\mathcal{E}_{min} = \mathbf{e}_{min}^H \mathbf{e}_{min} = \mathbf{d}^H \mathbf{e}_{min} = \mathbf{d}^H \mathcal{O}_{\perp}\mathbf{d} \tag{2.169}$$

In Figure 2.11 an example illustrating the relation among $\mathbf{d}$, $\mathbf{y}$, and $\mathbf{e}_{min}$ is given.

2.3.2 Solutions to the LS problem

If the inverse of $(\mathbf{T}^H\mathbf{T})$ does not exist, the solution of the LS problem (2.160) must be re-examined. This is what we will do in this section after taking a closer look at the associated system of equations (2.159). In general, let us consider the solutions to a linear system of equations

$$\mathbf{T}\mathbf{c} = \mathbf{d} \tag{2.170}$$

with $\mathbf{T}$ $N \times N$ *square matrix*. If $\mathbf{T}^{-1}$ exists, the solution $\mathbf{c} = \mathbf{T}^{-1}\mathbf{d}$ is unique and can be obtained in various ways [5]:

1. If $\mathbf{T}$ is *triangular* and non-singular, a solution to the system (2.170) can be found by the successive substitutions method with $O(N^2)$ operations.

2. In general, if $\mathbf{T}$ is non-singular, one can use the Gauss method, which involves three steps:

   a. Factorization of $\mathbf{T}$

   $$\mathbf{T} = \mathbf{L}\mathbf{U} \tag{2.171}$$

   with $\mathbf{L}$ lower triangular having all ones along the diagonal and $\mathbf{U}$ upper triangular;

b. Solution of the system in $\mathbf{z}$

$$\mathbf{L}\mathbf{z} = \mathbf{d} \tag{2.172}$$

through the successive substitutions method;

c. Solution of the system in $\mathbf{c}$

$$\mathbf{U}\mathbf{c} = \mathbf{z} \tag{2.173}$$

through the successive substitutions method.

This method requires $O(N^3)$ operations and $O(N^2)$ memory locations.

3. If $\mathbf{T}$ is *Hermitian* and non-singular, the factorization (2.171) becomes the *Cholesky decomposition*:

$$\mathbf{T} = \mathbf{L}\mathbf{L}^H \tag{2.174}$$

with $\mathbf{L}$ lower triangular having non-zero elements on the diagonal. This method requires $O(N^3)$ operations, about half as many as the Gauss method.

4. If $\mathbf{T}$ is *Toeplitz* and non-singular, one can use the generalized Shur algorithm with a complexity of $O(N^2)$: generally it is applicable to all $\mathbf{T}$ *structured matrices* [6]. We also recall the Kumar fast algorithm [4].

However, if $\mathbf{T}^{-1}$ does not exist, e.g., because $\mathbf{T}$ is not a square matrix, it is necessary to use alternative methods to solve the system (2.170) [5]: in particular we will consider the method of the pseudo-inverse. First, we will state the following result.

Singular value decomposition (SVD) of $\mathbf{T}$

We have seen in (1.369) how the $N \times N$ Hermitian matrix $\mathbf{R}$ can be decomposed in terms of a matrix $\mathbf{U}$ of eigenvectors and a diagonal matrix $\mathbf{\Lambda}$ of eigenvalues. Now we extend this concept to an arbitrary complex matrix $\mathbf{T}$. Given an $L \times N$ matrix $\mathbf{T}$ of rank R, two unitary matrices $\mathbf{V}$ and $\mathbf{U}$ exist, so that

$$\mathbf{T} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H \tag{2.175}$$

with

$$\mathbf{\Sigma} = \left(\begin{array}{c|c} \mathbf{D} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} \end{array}\right)_{L\times N} \tag{2.176}$$

$$\mathbf{D} = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_R) \qquad \sigma_1 > \sigma_2 > \cdots > \sigma_R > 0 \tag{2.177}$$

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_L]_{L\times L} \qquad \mathbf{U}\mathbf{U}^H = \mathbf{I}_{L\times L} \tag{2.178}$$

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N]_{N\times N} \qquad \mathbf{V}\mathbf{V}^H = \mathbf{I}_{N\times N} \tag{2.179}$$

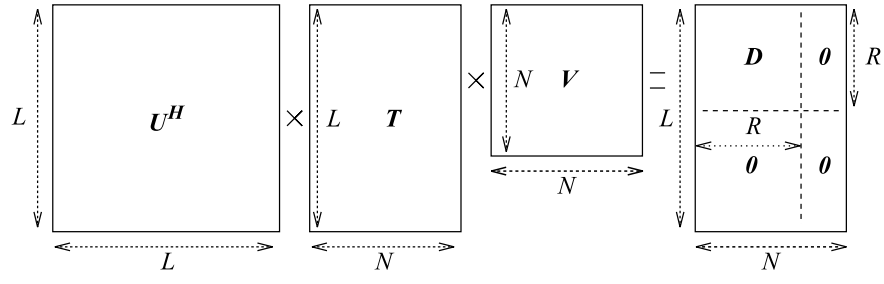


Figure 2.12. Singular value decomposition of matrix **T**.

In (2.177) the $\{\sigma_i\}$, $i = 1, \ldots, R$, are singular values of **T**. Being **U** and **V** unitary, it follows

$$\mathbf{U}^H \mathbf{T} \mathbf{V} = \mathbf{\Sigma} \tag{2.180}$$

as illustrated in Figure 2.12.

Definition 2.1
The *pseudo-inverse* of **T**, $L \times N$, of rank R, is given by the matrix

$$\mathbf{T}^{\#} = \mathbf{V} \mathbf{\Sigma}^{\#} \mathbf{U}^H = \sum_{i=1}^{R} \sigma_i^{-1} \mathbf{v}_i \mathbf{u}_i^H \tag{2.181}$$

where

$$\mathbf{\Sigma}^{\#} = \begin{bmatrix} \mathbf{D}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad \mathbf{D}^{-1} = \operatorname{diag}\left(\sigma_1^{-1}, \sigma_2^{-1}, \ldots, \sigma_R^{-1}\right) \tag{2.182}$$

We find an expression of $\mathbf{T}^{\#}$ for the two cases in which **T** has full rank,[4] that is $R = \min(L, N)$.

Case of an overdetermined system ($L > N$) and $R = N$. Note that in this case the system (2.170) has more equations than unknowns. Using the above relations it can be shown that

$$\mathbf{T}^{\#} = (\mathbf{T}^H \mathbf{T})^{-1} \mathbf{T}^H \tag{2.183}$$

In this case $\mathbf{T}^{\#}\mathbf{d}$ coincides with the solution of system (2.141).

Case of an underdetermined system ($L < N$) and $R = L$. Note that in this case there are fewer equations than unknowns, hence there are infinite solutions to the system (2.170). Again, it can be shown that

$$\mathbf{T}^{\#} = \mathbf{T}^H (\mathbf{T} \mathbf{T}^H)^{-1} \tag{2.184}$$

[4] We will denote the rank of **T** by rank(**T**).

Minimum norm solution

Definition 2.2
The *solution of a least squares problem* is given by the vector

$$\mathbf{c}_{ls} = \mathbf{T}^{\#}\mathbf{d} \tag{2.185}$$

where $\mathbf{T}^{\#}$ is the pseudo-inverse of $\mathbf{T}$.

By applying (2.185), the pseudo-inverse matrix $\mathbf{T}^{\#}$ gives the LS solution of minimum norm; in other words it solves the problem of finding the vector $\mathbf{c}$ that minimizes the squared error (2.128), $\mathcal{E} = \|\mathbf{e}\|^2 = \|\mathbf{y} - \mathbf{d}\|^2 = \|\mathbf{T}\mathbf{c} - \mathbf{d}\|^2$, and simultaneously minimizes the norm of the solution, $\|\mathbf{c}\|^2$. The constraint on $\|\mathbf{c}\|^2$ is needed in those cases in which there is more than one vector that minimizes $\|\mathbf{T}\mathbf{c} - \mathbf{d}\|^2$.

We list the different cases:

1. If $L = N$ and $\text{rank}(\mathbf{T}) = N$, i.e. $\mathbf{T}$ is non-singular,

$$\mathbf{T}^{\#} = \mathbf{T}^{-1} \tag{2.186}$$

2. If $L > N$ and

   a. $\text{rank}(\mathbf{T}) = N$, then

$$\mathbf{T}^{\#} = (\mathbf{T}^H\mathbf{T})^{-1}\mathbf{T}^H \tag{2.187}$$

   and $\mathbf{c}_{ls}$ is the LS solution of an overdetermined system of equations (2.170).

   b. $\text{rank}(\mathbf{T}) = R$ (also $< N$), from (2.185)

$$\mathbf{c}_{ls} = \sum_{i=1}^{R} \frac{\mathbf{v}_i^H\mathbf{T}^H\mathbf{d}}{\sigma_i^2}\mathbf{v}_i \tag{2.188}$$

3. If $L < N$ and

   a. $\text{rank}(\mathbf{T}) = L$, then

$$\mathbf{T}^{\#} = \mathbf{T}^H(\mathbf{T}\mathbf{T}^H)^{-1} \tag{2.189}$$

   and $\mathbf{c}_{ls}$ is the minimum norm solution of an underdetermined system of equations.

   b. $\text{rank}(\mathbf{T}) = R$ (also $< L$),

$$\mathbf{c}_{ls} = \sum_{i=1}^{R} \frac{\mathbf{u}_i^H\mathbf{d}}{\sigma_i^2}\mathbf{T}^H\mathbf{u}_i \tag{2.190}$$

Only solutions (2.185) in the cases (2.186) and (2.187) coincide with the solution (2.142).

The computation of the pseudo-inverse $\mathbf{T}^{\#}$ directly from SVD and the expansion of $\mathbf{c}$ in terms of $\{\mathbf{u}_i\}$, $\{\mathbf{v}_i\}$ and $\{\sigma_i^2\}$ have two advantages with respect to the direct computation of $\mathbf{T}^{\#}$ in the form (2.187), for $L > N$ and $\text{rank}(\mathbf{T}) = N$, or in the form (2.189), for $L < N$ and $\text{rank}(\mathbf{T}) = L$:

1. The SVD also gives the rank of $\mathbf{T}$ through the number of non-zero singular values.
2. The required accuracy in computing $\mathbf{T}^{\#}$ via SVD is almost halved with respect to the computation of $(\mathbf{T}^H\mathbf{T})^{-1}$ or $(\mathbf{T}\mathbf{T}^H)^{-1}$.

There are two algorithms to determine the SVD of $\mathbf{T}$: the Jacobi algorithm and the Householder transformation [7].

We conclude citing two texts [8, 9], which report examples of realizations of the algorithms described in this section.

Bibliography

[1] S. Haykin, *Adaptive filter theory*. Englewood Cliffs, NJ: Prentice-Hall, 3rd ed., 1996.

[2] M. L. Honig and D. G. Messerschmitt, *Adaptive filters: structures, algorithms and applications*. Boston, MA: Kluwer Academic Publishers, 1984.

[3] P. Delsarte and Y. V. Genin, "The split Levinson algorithm", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 34, pp. 470–478, June 1986.

[4] R. Kumar, "A fast algorithm for solving a Toeplitz system of equations", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 33, pp. 254–267, Feb. 1985.

[5] G. H. Golub and C. F. van Loan, *Matrix computations*. Baltimore and London: The Johns Hopkins University Press, 2nd ed., 1989.

[6] N. Al-Dhahir and J. M. Cioffi, "Fast computation of channel-estimate based equalizers in packet data transmission", *IEEE Trans. on Signal Processing*, vol. 43, pp. 2462–2473, Nov. 1995.

[7] S. A. T. W. H. Press, B. P. Flannery and W. T. Vetterling, *Numerical Recipes*. New York: Cambridge University Press, 3rd ed., 1988.

[8] L. S. Marple Jr., *Digital spectral analysis with applications*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[9] S. M. Kay, *Modern spectral estimation-theory and applications*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[10] S. M. Kay, *Fundamentals of statistical signal processing: estimation theory*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

Appendix 2.A *The estimation problem*

The estimation problem for random variables

Let d and x be two r.v.s, somehow related via the function f, that is $x = f(d)$. On the basis of an observation, let the value of x equal to β, that is $x = \beta$. The estimation problem is to determine what the corresponding value of d is.

Obviously, if f were known and the inverse function f^{-1} existed, the solution would be trivial; however, we often know only the joint probability density function of the two r.v.s, $p_{dx}(\alpha, \beta)$. In any case, using as *estimate of* d the function

$$\hat{d} = h(x) \tag{2.191}$$

the estimation error is given by

$$e = d - \hat{d} \tag{2.192}$$

MMSE estimation

Let $p_d(\alpha)$ and $p_x(\beta)$ be the probability density functions of d and x, respectively, and $p_{d|x}(\alpha \mid \beta)$ the conditional probability density function of d given $x = \beta$; moreover let $p_x(\beta) \neq 0$, $\forall \beta$. We wish to determine the function h that minimizes the mean-square error, that is

$$\begin{aligned} J = E[e^2] &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} [\alpha - h(\beta)]^2 \, p_{dx}(\alpha, \beta) \, d\alpha \, d\beta \\ &= \int_{-\infty}^{+\infty} p_x(\beta) \int_{-\infty}^{+\infty} [\alpha - h(\beta)]^2 \, p_{d|x}(\alpha \mid \beta) \, d\alpha \, d\beta \end{aligned} \tag{2.193}$$

where the relation $p_{dx}(\alpha, \beta) = p_x(\beta) p_{d|x}(\alpha \mid \beta)$ is used.

Theorem 2.2
The estimator $h(\beta)$ that minimizes J is given by the expected value of d given $x = \beta$,

$$h(\beta) = E[d \mid x = \beta] \tag{2.194}$$

Proof. The integral (2.193) is minimum when the function

$$\int_{-\infty}^{+\infty} [\alpha - h(\beta)]^2 \, p_{d|x}(\alpha \mid \beta) \, d\alpha \tag{2.195}$$

is minimized for every value of β. Using the variational method (see Appendix 8.A), we find that this occurs if

$$2 \int_{-\infty}^{+\infty} [\alpha - h(\beta)] \, p_{d|x}(\alpha \mid \beta) \, d\alpha = 0 \qquad \forall \beta \tag{2.196}$$

that is for

$$h(\beta) = \int_{-\infty}^{+\infty} \alpha \, p_{d|x}(\alpha \mid \beta) \, d\alpha = \int_{-\infty}^{+\infty} \alpha \, \frac{p_{dx}(\alpha, \beta)}{p_x(\beta)} \, d\alpha \tag{2.197}$$

from which the (2.194) follows.

An alternative to the MMSE criterion for determining $\hat{d}$ is given by the *maximum a posteriori probability* (MAP) criterion, which yields

$$\hat{d} = \arg\max_{\alpha} p_{d|x}(\alpha \mid \beta) \tag{2.198}$$

where the notation arg max is defined in (6.21). If the distribution of d is uniform, the MAP criterion becomes the *maximum likelihood* (ML) criterion, where

$$\hat{d} = \arg\max_{\alpha} p_{x|d}(\beta \mid \alpha) \tag{2.199}$$

Examples of both MAP and ML criteria are given in Chapters 6 and 14.

Example 2.A.1
Let d and x be two jointly Gaussian r.v.s with mean values m_d and m_x, respectively, and covariance $\mathrm{c} = E[(d - \mathrm{m}_d)(x - \mathrm{m}_x)]$. After several steps, it can be shown that [10]

$$h(\beta) = \mathrm{m}_d + \frac{\mathrm{c}}{\sigma_x^2} (\beta - \mathrm{m}_x) \tag{2.200}$$

The corresponding mean-square error is equal to

$$J_{min} = \sigma_d^2 - \left(\frac{\mathrm{c}}{\sigma_x} \right)^2 \tag{2.201}$$

Example 2.A.2
Let $x = d + w$, where d and w are two statistically independent r.v.s. For $w \in \mathcal{N}(0, 1)$ and $d \in \{-1, 1\}$ with $P[d = -1] = P[d = 1] = 1/2$, it can be shown that

$$h(\beta) = \tanh(\beta) \tag{2.202}$$

Extension to multiple observations

In the case of several observations,

$$x_1 = \beta_1, \ldots, x_N = \beta_N \tag{2.203}$$

the estimation of d is obtained by applying the following theorem, whose proof is similar to the case of a single observation.

Theorem 2.3
The estimator of d, $\hat{d} = h(x_1, \ldots, x_N)$, that minimizes $J = E[(d - \hat{d})^2]$ is given by

$$\begin{aligned} h(\beta_1, \ldots, \beta_N) &= E[d \mid x_1 = \beta_1, \ldots, x_N = \beta_N] \\ &= \int_{-\infty}^{+\infty} \alpha \, p_{d|x_1 \ldots x_N}(\alpha \mid x_1 = \beta_1, \ldots, x_N = \beta_N) \, d\alpha \\ &= \int \alpha \, \frac{p_{d, x_1 \ldots x_N}(\alpha, \beta_1, \ldots, \beta_N)}{p_{x_1 \ldots x_N}(\beta_1, \ldots, \beta_N)} \, d\alpha \end{aligned} \tag{2.204}$$

In the following, to simplify the formulation we will refer to r.v.s with zero mean.

Example 2.A.3
Let d, $\mathbf{x} = [x_1, \ldots, x_N]^T$, be real-valued jointly Gaussian r.v.s with *zero mean* and the following second order description:

- *Correlation matrix of observations*

$$\mathbf{R} = E[\mathbf{x}\, \mathbf{x}^T] \tag{2.205}$$

- *Cross-correlation vector*

$$\mathbf{p} = E[d\mathbf{x}] \tag{2.206}$$

For $\mathbf{x} = \boldsymbol{\beta}$, it can be shown that

$$h(\boldsymbol{\beta}) = \mathbf{p}^T \, \mathbf{R}^{-1} \, \boldsymbol{\beta} \tag{2.207}$$

and

$$J_{min} = \sigma_d^2 - \mathbf{p}^T \, \mathbf{R}^{-1} \, \mathbf{p} \tag{2.208}$$

MMSE linear estimation

For a low complexity of implementation, it is often convenient to consider a linear function h. Letting $\mathbf{c} = [c_1, \ldots, c_N]^T$, in the case of multiple observations the estimate is a linear combination of observations, and

$$\hat{d} = \mathbf{c}^T \, \mathbf{x} + b \tag{2.209}$$

where b is a constant.

In the case of real-valued r.v.s, using the definitions (2.205) and (2.206) it is easy to prove the following theorem (see page 130).

Theorem 2.4
Given the vector of observations $\mathbf{x}$, the MMSE linear estimator of d has the following expression

$$\hat{d} = \mathbf{p}^T \, \mathbf{R}^{-1} \, \mathbf{x} \tag{2.210}$$

In other words,

$$\mathbf{c}_{opt} = \mathbf{R}^{-1}\,\mathbf{p}$$

and the corresponding mean-square error is

$$J_{min} = \sigma_d^2 - \mathbf{p}^T\,\mathbf{R}^{-1}\,\mathbf{p} \tag{2.211}$$

Note that the r.v.s are assumed to have zero mean.

Observation 2.3
Comparing (2.210) and (2.211) with (2.207) and (2.208), respectively, we note that, in the case of jointly Gaussian r.v.s, linear estimation coincides with optimum MMSE estimation.

MMSE linear estimation for random vectors

We extend the results of the previous section to the case of complex-valued r.v.s, and for a desired vector signal. Let $\mathbf{x}$ be an observation, modeled as a vector of N r.v.s,

$$\mathbf{x}^T = [x_1, x_2, \ldots, x_N] \tag{2.212}$$

Moreover, let $\mathbf{d}$ be the desired vector, modeled as a vector of M r.v.s,

$$\mathbf{d}^T = [d_1, d_2, \ldots, d_M] \tag{2.213}$$

We introduce the following correlation matrices:

$$\mathbf{r}_{d_i\mathbf{x}} = E[d_i\,\mathbf{x}^*] \tag{2.214}$$

$$\mathbf{R}_{\mathbf{xd}} = E[\mathbf{x}^*\mathbf{d}^T] = [\mathbf{r}_{d_1\mathbf{x}}, \mathbf{r}_{d_2\mathbf{x}}, \ldots, \mathbf{r}_{d_M\mathbf{x}}] \tag{2.215}$$

$$\mathbf{R}_{\mathbf{dx}} = E[\mathbf{d}^*\mathbf{x}^T] = \mathbf{R}_{\mathbf{xd}}^H \tag{2.216}$$

$$\mathbf{R}_{\mathbf{x}} = E[\mathbf{x}^*\mathbf{x}^T] \tag{2.217}$$

$$\mathbf{R}_{\mathbf{d}} = E[\mathbf{d}^*\mathbf{d}^T] \tag{2.218}$$

The problem is to determine a linear transformation of $\mathbf{x}$, given by

$$\hat{\mathbf{d}} = \mathbf{C}^T\mathbf{x} + \mathbf{b} \tag{2.219}$$

such that $\hat{\mathbf{d}}$ is a close replica of $\mathbf{d}$ in the mean-square error sense.

Definition 2.3
The *linear minimum mean-square error* (LMMSE) estimator, consisting of the $N \times M$ matrix $\mathbf{C}$, and of the $M \times 1$ vector $\mathbf{b}$, coincides with the linear function of the observations (2.219) that minimizes the cost function

$$J = E[\|\mathbf{d} - \hat{\mathbf{d}}\|^2] = \sum_{m=1}^{M} E[|d_m - \hat{d}_m|^2] \tag{2.220}$$

In other words, the optimum coefficients $\mathbf{C}$ and $\mathbf{b}$ are the solution of the following problem:

$$\min_{\mathbf{C},\mathbf{b}} J \tag{2.221}$$

We note that in the formulation of Section 2.1 we have

$$\mathbf{x}^T = [x(k), \ldots, x(k-N+1)] \tag{2.222}$$

$$\mathbf{d}^T = [d(k)] \tag{2.223}$$

that is $M = 1$, and the matrix $\mathbf{C}$ becomes a column vector.

We determine now the expression of $\mathbf{C}$ and $\mathbf{b}$ in terms of the correlation matrices introduced above. First of all, we observe that if $\mathbf{d}$ and $\mathbf{x}$ have zero mean, then $\mathbf{b} = \mathbf{0}$, since the choice of $\mathbf{b} = \tilde{\mathbf{b}} \neq \mathbf{0}$ implies an estimator $\tilde{\mathbf{C}}^T\mathbf{x} + \tilde{\mathbf{b}}$ with a larger value of the cost function. In fact

$$\begin{aligned} J &= E[\|\mathbf{d} - \tilde{\mathbf{C}}^T\mathbf{x} - \tilde{\mathbf{b}}\|^2] \\ &= E[\|\mathbf{d} - \tilde{\mathbf{C}}^T\mathbf{x}\|^2] - 2Re\{E[(\mathbf{d} - \tilde{\mathbf{C}}^T\mathbf{x})^H \tilde{\mathbf{b}}]\} + \|\tilde{\mathbf{b}}\|^2 \\ &= E[\|\mathbf{d} - \tilde{\mathbf{C}}^T\mathbf{x}\|^2] + \|\tilde{\mathbf{b}}\|^2 \end{aligned} \tag{2.224}$$

being $E[\mathbf{d}] = E[\mathbf{x}] = \mathbf{0}$. The (2.224) implies that the choice $\tilde{\mathbf{b}} = \mathbf{0}$ yields the minimum value of J. Without loss of generality, we will assume that both $\mathbf{x}$ and $\mathbf{d}$ are zero mean random vectors.

Scalar case. For $M = 1$, $d = d_1$, and

$$\hat{d} = \hat{d}_1 = \mathbf{c}_1^T\mathbf{x} = \mathbf{x}^T\mathbf{c}_1 \tag{2.225}$$

with $\mathbf{c}_1$ column vector with N coefficients. In this case the problem (2.221) leads again to the Wiener filter; the solution is given by

$$\mathbf{R}_\mathbf{x}\mathbf{c}_1 = \mathbf{r}_{d_1\mathbf{x}} \tag{2.226}$$

where $\mathbf{r}_{d_1\mathbf{x}}$ is defined by (2.214).

Vector case. For $M > 1$, $\mathbf{d}$ and $\hat{\mathbf{d}}$ are M-dimensional vectors. Nevertheless, since the function (2.220) operates on single components, the vector problem (2.221) leads to M scalar problems, each with input $\mathbf{x}$ and output $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_M$, respectively. Therefore the columns of the matrix $\mathbf{C}$, $\mathbf{c}_m$, satisfy equations of the type (2.226),

$$\mathbf{R}_\mathbf{x}\mathbf{c}_m = \mathbf{r}_{d_m\mathbf{x}} \qquad m = 1, \ldots, M \tag{2.227}$$

hence, based on the definition (2.215), it results in

$$\mathbf{C} = \mathbf{R}_\mathbf{x}^{-1}\,\mathbf{R}_{\mathbf{xd}} \tag{2.228}$$

Thus, the optimum estimator in the LMMSE sense is given by

$$\hat{\mathbf{d}} = (\mathbf{R}_\mathbf{x}^{-1}\,\mathbf{R}_{\mathbf{xd}})^T\mathbf{x} \tag{2.229}$$

Value of the cost function. On the basis of the estimation error

$$\mathbf{e} = \mathbf{d} - \hat{\mathbf{d}} \tag{2.230}$$

with correlation matrix

$$\mathbf{R_e} = E[\mathbf{e}^* \mathbf{e}^T] = \mathbf{R_d} - \mathbf{R_{dx}}\mathbf{C} - \mathbf{C}^H \mathbf{R_{xd}} + \mathbf{C}^H \mathbf{R_x} \mathbf{C} \tag{2.231}$$

the cost function (2.220) is given by the trace of $\mathbf{R_e}$,

$$J = \mathrm{tr}[\mathbf{R_e}] \tag{2.232}$$

Substituting (2.228) in (2.232), yields

$$J_{min} = \mathrm{tr}[\mathbf{R_d} - \mathbf{R_{dx}}\, \mathbf{R_x^{-1}}\, \mathbf{R_{xd}}] \tag{2.233}$$

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 3

Adaptive transversal filters

We reconsider the Wiener filter introduced in Section 2.1. Given two random processes x and d, we want to determine the coefficients of a FIR filter having input x, so that the filter output y is a replica as accurate as possible of the process d. Adopting, for example, the mean-square error criterion, it is required that the autocorrelation matrix $\mathbf{R}$ of the filter input vector, and the cross-correlation $\mathbf{p}$ between the desired output and the input vector be known (see (2.15)). Estimating these correlations is usually difficult.[1] Moreover, the optimum solution requires solving a system of equations with a computational complexity that is at least proportional to the square of the number of filter coefficients. In this chapter, we develop iterative algorithms with low computational complexity to obtain an approximation of the Wiener solution.

We will consider transversal FIR filters[2] with N coefficients. In general the coefficients may vary with time. The filter structure at instant k is illustrated in Figure 3.1. We define:

1. Coefficient vector at instant k:

$$\mathbf{c}^T(k) = [c_0(k), c_1(k), \ldots, c_{N-1}(k)] \tag{3.1}$$

2. Input vector at instant k:

$$\mathbf{x}^T(k) = [x(k), x(k-1), \ldots, x(k-N+1)] \tag{3.2}$$

The output signal is given by

$$y(k) = \sum_{i=0}^{N-1} c_i(k)x(k-i) = \mathbf{x}^T(k)\mathbf{c}(k) \tag{3.3}$$

Comparing $y(k)$ with the desired response $d(k)$, we obtain the estimation error[3]

$$e(k) = d(k) - y(k) \tag{3.4}$$

---

[1] Two estimation methods are presented in Section 1.11.2.

[2] For the analysis of IIR adaptive filters we refer the reader to [1, 2].

[3] In this chapter the definition of the estimation error is given as the difference between the desired signal and the filter output. Depending on the application, the estimation error may be defined using the opposite sign. Some caution is therefore necessary in using the equations of an adaptive filter.

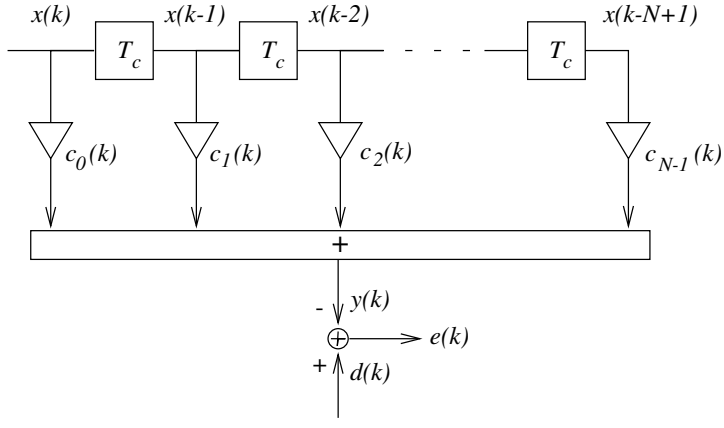


Figure 3.1. Structure of an adaptive transversal filter at instant k.

Depending on the cost function associated with $\{e(k)\}$, in Chapter 2 two classes of algorithms have been developed:

1. mean-square error (MSE),
2. least squares (LS).

In the following sections we will present iterative algorithms for each of the two classes.

3.1 Adaptive transversal filter: MSE criterion

The cost function, or functional, to minimize is

$$J(k) = E[|e(k)|^2] \tag{3.5}$$

Assuming that x and d are individually and jointly WSS, analogously to (2.17), $J(k)$ can be written as

$$J(k) = \sigma_d^2 - \mathbf{c}^H(k)\mathbf{p} - \mathbf{p}^H\mathbf{c}(k) + \mathbf{c}^H(k)\mathbf{R}\mathbf{c}(k) \tag{3.6}$$

where $\mathbf{R}$ and $\mathbf{p}$ are defined respectively in (2.16) and (2.15). The optimum Wiener–Hopf solution is $\mathbf{c}(k) = \mathbf{c}_{opt}$, where $\mathbf{c}_{opt}$ is given by (2.34). The corresponding minimum value of $J(k)$ is J_{min}, given by (2.40).

3.1.1 Steepest descent or gradient algorithm

Our first step is to realize a *deterministic iterative* procedure to compute $\mathbf{c}_{opt}$. We will see that this method avoids the computation of the inverse $\mathbf{R}^{-1}$, however, it requires that $\mathbf{R}$ and $\mathbf{p}$ be known.

The steepest descent or gradient algorithm is defined as:

$$\mathbf{c}(k+1) = \mathbf{c}(k) - \tfrac{1}{2}\mu \nabla_{\mathbf{c}(k)} J(k) \tag{3.7}$$

where $\nabla_{\mathbf{c}(k)} J(k)$ denotes the gradient of $J(k)$ with respect to $\mathbf{c}$ (see (2.18)), μ is the adaptation gain, a real-valued positive constant, and k is the iteration index, in general not necessarily coinciding with time instants.

As $J(k)$ is a quadratic function of the vector of coefficients, from (2.31) we find

$$\nabla_{\mathbf{c}(k)} J(k) = 2(\mathbf{R}\mathbf{c}(k) - \mathbf{p}) \tag{3.8}$$

hence

$$\mathbf{c}(k+1) = \mathbf{c}(k) - \mu(\mathbf{R}\mathbf{c}(k) - \mathbf{p}) \tag{3.9}$$

In the scalar case ($N = 1$), for real-valued signals the above relations become:

$$J(k) = J_{min} + r_x(0)(c_0(k) - c_{opt,0})^2 \tag{3.10}$$

and

$$\nabla_{c_0} J(k) = \frac{\partial}{\partial c_0} J(k) = 2r_x(0)(c_0(k) - c_{opt,0}) \tag{3.11}$$

The iterative algorithm is given by:

$$c_0(k+1) = c_0(k) - \mu r_x(0)(c_0(k) - c_{opt,0}) \tag{3.12}$$

The behavior of J and the sign of $\nabla_{c_0} J(k)$ as a function of c_0 is illustrated in Figure 3.2. In the two-dimensional case ($N = 2$), the trajectory of $\nabla_{\mathbf{c}} J(k)$ is illustrated in Figure 3.3. We recall that in general the gradient vector for $\mathbf{c} = \mathbf{c}(k)$ is orthogonal to the locus of points with constant J that includes $\mathbf{c}(k)$.

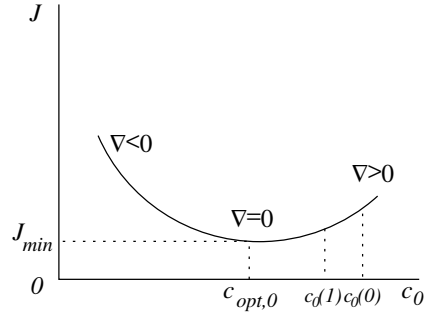


Figure 3.2. Behavior of J and sign of the gradient vector $\nabla_{\mathbf{c}}$ in the scalar case ($N = 1$).

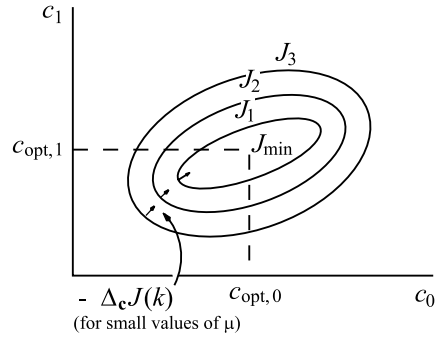


Figure 3.3. Loci of points with constant J and trajectory of $\nabla_{\mathbf{c}}J$ in the two-dimensional case ($N = 2$).

Stability of the steepest descent algorithm

Substituting for $\mathbf{p}$ the expression given by (2.32), the iterative algorithm (3.9) can be written as

$$\begin{aligned}\mathbf{c}(k+1) &= \mathbf{c}(k) - \mu[\mathbf{R}\mathbf{c}(k) - \mathbf{R}\mathbf{c}_{opt}] \\ &= [\mathbf{I} - \mu\mathbf{R}]\mathbf{c}(k) + \mu\mathbf{R}\mathbf{c}_{opt}\end{aligned} \tag{3.13}$$

Defining the *coefficient error* vector as

$$\Delta\mathbf{c}(k) = \mathbf{c}(k) - \mathbf{c}_{opt} \tag{3.14}$$

from (3.13) we obtain

$$\begin{aligned}\Delta\mathbf{c}(k+1) &= [\mathbf{I} - \mu\mathbf{R}]\mathbf{c}(k) + [\mu\mathbf{R} - \mathbf{I}]\mathbf{c}_{opt} \\ &= [\mathbf{I} - \mu\mathbf{R}]\Delta\mathbf{c}(k)\end{aligned} \tag{3.15}$$

Starting at $k = 0$ with arbitrary $\mathbf{c}(0)$, that is with $\Delta\mathbf{c}(0) = \mathbf{c}(0) - \mathbf{c}_{opt}$, we determine now the conditions for the convergence of $\mathbf{c}(k)$ to $\mathbf{c}_{opt}$ or, equivalently, for the convergence of $\Delta\mathbf{c}(k+1)$ to $\mathbf{0}$.

Using the decomposition (1.369), $\mathbf{R} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^H$, where $\mathbf{U}$ is the unitary matrix formed of eigenvectors of $\mathbf{R}$, and $\boldsymbol{\Lambda}$ is the diagonal matrix of eigenvalues $\{\lambda_i\}$, $i = 1, \ldots, N$, and setting (see (2.46))

$$\boldsymbol{v}(k) = \mathbf{U}^H\Delta\mathbf{c}(k) \tag{3.16}$$

equation (3.15) becomes

$$\boldsymbol{v}(k+1) = [\mathbf{I} - \mu\boldsymbol{\Lambda}]\boldsymbol{v}(k) \tag{3.17}$$

Conditions for convergence

As $\mathbf{\Lambda}$ is diagonal, the i-th component of the vector $\boldsymbol{\nu}(k)$ in (3.17) satisfies the difference equation:

$$\nu_i(k+1) = (1 - \mu\lambda_i)\nu_i(k) \qquad i = 1, 2, \ldots, N \tag{3.18}$$

Hence, $\nu_i(k)$ as a function of k is given by (see Figure 3.4):

$$\nu_i(k) = (1 - \mu\lambda_i)^k \nu_i(0) \tag{3.19}$$

The i-th component of the vector $\boldsymbol{\nu}(k)$ converges, that is

$$\nu_i(k) \xrightarrow[k\to\infty]{} 0 \qquad \forall \nu_i(0) \tag{3.20}$$

if and only if

$$|1 - \mu\lambda_i| < 1 \tag{3.21}$$

or, equivalently,

$$-1 < 1 - \mu\lambda_i < 1 \tag{3.22}$$

As λ_i is positive (see (1.360)), we have that the algorithm converges if and only if

$$0 < \mu < \frac{2}{\lambda_i} \qquad i = 1, 2, \ldots, N \tag{3.23}$$

If λ_{max} (λ_{min}) is the largest (smallest) eigenvalue of $\mathbf{R}$, observing (3.23) the convergence condition can be expressed as

$$0 < \mu < \frac{2}{\lambda_{max}} \tag{3.24}$$

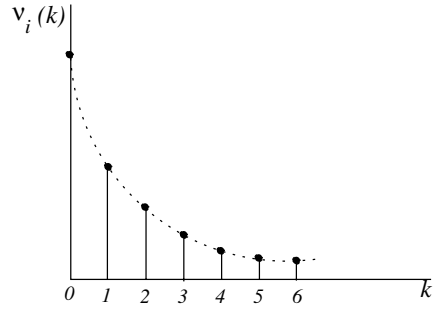


Figure 3.4. Plot of $\nu_i(k)$ as a function of k for $\mu\lambda_i < 1$ and $|1 - \mu\lambda_i| < 1$. In the case $\mu\lambda_i > 1$ and $|1 - \mu\lambda_i| < 1$, $\nu_i(k)$ is still decreasing in magnitude, but it assumes alternating positive and negative values.

Correspondingly, observing (3.16) and (3.19) we obtain the expression of the vector of coefficients, given by

$$\begin{aligned}\mathbf{c}(k) &= \mathbf{c}_{opt} + \mathbf{U}\boldsymbol{\nu}(k) \\ &= \mathbf{c}_{opt} + [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N]\boldsymbol{\nu}(k) \\ &= \mathbf{c}_{opt} + \sum_{i=1}^{N} \mathbf{u}_i \nu_i(k) \\ &= \mathbf{c}_{opt} + \sum_{i=1}^{N} \mathbf{u}_i (1 - \mu\lambda_i)^k \nu_i(0)\end{aligned} \tag{3.25}$$

Therefore, for each coefficient it results in

$$c_n(k) = c_{opt,n} + \sum_{i=1}^{N} u_{i,n}(1 - \mu\lambda_i)^k \nu_i(0) \qquad n = 0, \ldots, N-1 \tag{3.26}$$

where $u_{i,n}$ is the n-th component of $\mathbf{u}_i$. In (3.26) the term $u_{i,n}(1-\mu\lambda_i)^k \nu_i(0)$ characterizes the *i-th mode of convergence*. Note that, if the convergence condition (3.23) is satisfied, each coefficient c_n, $n = 0, \ldots, N-1$, converges to the optimum solution as a weighted sum of N exponentials, each with the time constant[4]

$$\tau_i = -\frac{1}{\ln|1 - \mu\lambda_i|} \simeq \frac{1}{\mu\lambda_i} \qquad i = 1, 2, \ldots, N \tag{3.27}$$

where the approximation is valid if $\mu\lambda_i \ll 1$.

Choice of the adaptation gain for fastest convergence

The *speed of convergence*, which is related to the inverse of the convergence time, depends on the choice of μ. We define as μ_{opt} the value of μ that minimizes the time constant of the slowest mode. If we let

$$\xi(\mu) = \max_i |1 - \mu\lambda_i| \tag{3.28}$$

then we need to determine

$$\min_{\mu} \xi(\mu) \tag{3.29}$$

As illustrated in Figure 3.5, the solution is obtained for

$$1 - \mu\lambda_{min} = -(1 - \mu\lambda_{max}) \tag{3.30}$$

[4] The time constant is the number of iterations needed to reduce the signal associated with the i-th mode of a factor e.

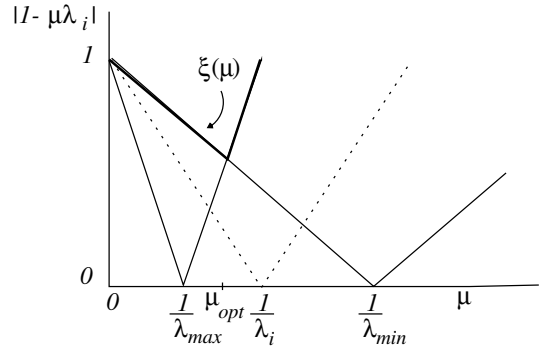


Figure 3.5. Plot of ξ and $|1-\mu\lambda_i|$ as a function of μ for different values of λ_i: $\lambda_{min} < \lambda_i < \lambda_{max}$.

from which we get

$$\mu_{opt} = \frac{2}{\lambda_{max} + \lambda_{min}} \tag{3.31}$$

and

$$\xi(\mu_{opt}) = 1 - \frac{2}{\lambda_{max} + \lambda_{min}}\lambda_{min} = \frac{\lambda_{max} - \lambda_{min}}{\lambda_{max} + \lambda_{min}} = \frac{\chi(\mathbf{R}) - 1}{\chi(\mathbf{R}) + 1} \tag{3.32}$$

where $\chi(\mathbf{R}) = \lambda_{max}/\lambda_{min}$ is the *eigenvalue spread* (1.376). We note that other values of μ (associated with λ_{max} or λ_{min}) cause a slower mode.

We emphasize that $\xi(\mu_{opt})$ is a monotonic function of the eigenvalue spread (see Figure 3.6), and consequently the larger the eigenvalue spread the slower the convergence.

Transient behavior of the MSE

From (2.47) the general relation holds

$$J(k) = J_{min} + \sum_{i=1}^{N} \lambda_i |v_i(k)|^2 \tag{3.33}$$

Now using (3.19) we have

$$J(k) = J_{min} + \sum_{i=1}^{N} \lambda_i (1 - \mu\lambda_i)^{2k} v_i^2(0) \tag{3.34}$$

Consequently, if the condition for convergence is verified, for $k \to \infty$, $J(k) \to J_{min}$ as a weighted sum of exponentials. The i-th mode will have a time constant given by

$$\tau_{\mathrm{MSE},i} = -\frac{1}{2\ln|1 - \mu\lambda_i|} \simeq \frac{1}{2\mu\lambda_i} \tag{3.35}$$

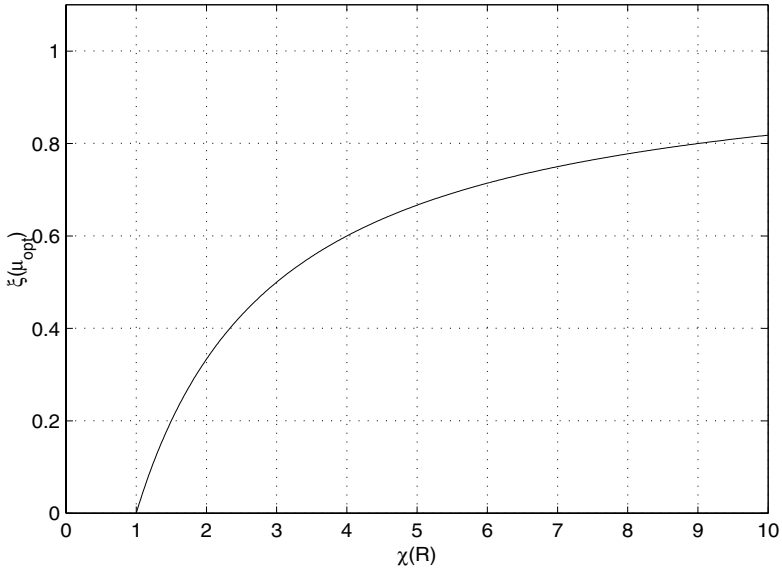


Figure 3.6. $\xi(\mu_{opt})$ as a function of the eigenvalue spread $\chi(\mathbf{R}) = \lambda_{max}/\lambda_{min}$.

assuming $\mu\lambda_i \ll 1$. We note that (3.34) is different from (3.26) because of the presence of λ_i as weight of the i-ith mode: consequently, modes associated with small eigenvalues tend to weigh less in the convergence of $J(k)$. In particular, let us examine the two-dimensional case ($N = 2$). Recalling the observation that $J(k)$ increases more rapidly (slowly) in the direction of the eigenvector corresponding to $\lambda = \lambda_{max}$ ($\lambda = \lambda_{min}$) (see Figure 2.4), we have the following two cases.

Case 1 for $\lambda_1 \ll \lambda_2$. Choosing $\mathbf{c}(0)$ on the ν_2 axis (in correspondence of λ_{max}) we have the following situations:

$$\text{if } \mu \begin{cases} < \dfrac{1}{\lambda_{max}} & \text{the iterative algorithm has a non-oscillatory behavior} \\ = \dfrac{1}{\lambda_{max}} & \text{the iterative algorithm converges in one iteration} \\ > \dfrac{1}{\lambda_{max}} & \text{the iterative algorithm has a trajectory that oscillates around the minimum} \end{cases} \tag{3.36}$$

Let $\mathbf{u}_{\lambda_{min}}$ and $\mathbf{u}_{\lambda_{max}}$ be the eigenvectors corresponding to λ_{min} and λ_{max}, respectively. If no further information is given regarding the initial condition $\mathbf{c}(0)$, choosing $\mu = \mu_{opt}$ the algorithm exhibits monotonic convergence along $\mathbf{u}_{\lambda_{min}}$, and an oscillatory behavior around the minimum along $\mathbf{u}_{\lambda_{max}}$.

Case 2 for $\lambda_2 = \lambda_1$. Choosing $\mu = 1/\lambda_{max}$ the algorithm converges in one iteration, independently of the initial condition $\mathbf{c}(0)$.

3.1.2 The least mean-square (LMS) algorithm

The LMS or stochastic gradient algorithm is an algorithm with low computational complexity that provides an approximation to the optimum Wiener–Hopf solution without requiring knowledge of $\mathbf{R}$ and $\mathbf{p}$. Actually, the following instantaneous estimates are used:

$$\hat{\mathbf{R}}(k) = \mathbf{x}^*(k)\mathbf{x}^T(k) \tag{3.37}$$

and

$$\hat{\mathbf{p}}(k) = d(k)\mathbf{x}^*(k) \tag{3.38}$$

The gradient vector (3.8) is thus estimated to be[5]

$$\begin{aligned} \widehat{\nabla_{\mathbf{c}(k)} J(k)} &= -2d(k)\mathbf{x}^*(k) + 2\mathbf{x}^*(k)\mathbf{x}^T(k)\mathbf{c}(k) \\ &= -2\mathbf{x}^*(k)[d(k) - \mathbf{x}^T(k)\mathbf{c}(k)] \\ &= -2\mathbf{x}^*(k)e(k) \end{aligned} \tag{3.39}$$

The equation for the adaptation of the filter coefficients, where k now denotes a given instant, becomes

$$\mathbf{c}(k+1) = \mathbf{c}(k) - \tfrac{1}{2}\mu \widehat{\nabla_{\mathbf{c}(k)} J(k)} \tag{3.40}$$

that is

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \mu e(k)\mathbf{x}^*(k) \tag{3.41}$$

Observation 3.1
The same equation is obtained for a cost function equal to $|e(k)|^2$, whose gradient is given by

$$\widehat{\nabla_{\mathbf{c}(k)} J(k)} = -2e(k)\mathbf{x}^*(k) \tag{3.42}$$

Implementation

The block diagram for the implementation of the LMS algorithm is shown in Figure 3.7, with reference to the following parameters and equations.

[5] We note that (3.39) represents an *unbiased* estimate of the gradient (3.8), but in general it also exhibits a large variance.

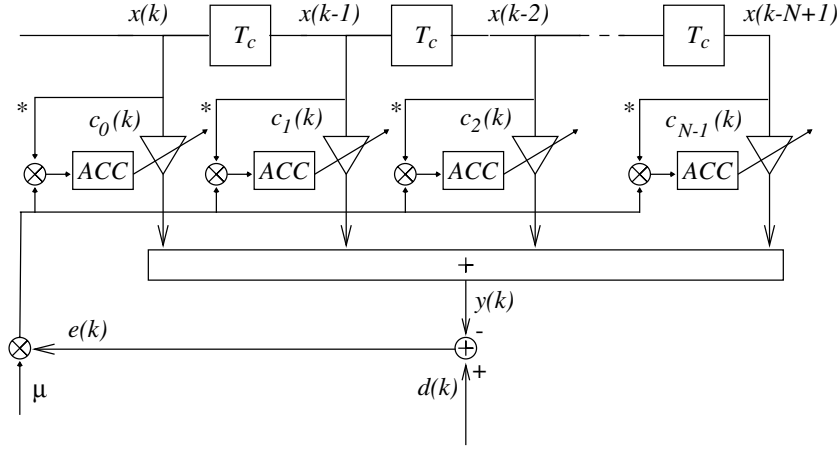


Figure 3.7. Block diagram of an adaptive transversal filter adapted according to the LMS algorithm.

Parameters. Required parameters are:

1. N, number of coefficients of the filter.
2. $0 < \mu < \dfrac{2}{\text{statistical power of input vector}}$.

Filter. The filter output is given by

$$y(k) = \mathbf{x}^T(k)\mathbf{c}(k) \tag{3.43}$$

Adaptation.

1. Estimation error

$$e(k) = d(k) - y(k) \tag{3.44}$$

2. Coefficient vector adaptation

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \mu e(k)\mathbf{x}^*(k) \tag{3.45}$$

Initialization. If no *a priori* information is available, we set

$$\mathbf{c}(0) = \mathbf{0} \tag{3.46}$$

The accumulators (ACC) in Figure 3.7 are used to memorize coefficients, which are updated by the current value of $\mu e(k)\mathbf{x}^*(k)$.

Computational complexity

For every iteration we have $2N + 1$ complex multiplication (N due to filtering and $N + 1$ to adaptation) and $2N$ complex additions. Therefore the LMS algorithm has a complexity of $O(N)$.

Canonical model

The LMS algorithm operates with complex-valued signals and coefficients. We can rewrite complex-valued quantities as follows.

$$\textit{Input vector} \qquad \mathbf{x}(k) = \mathbf{x}_I(k) + j\mathbf{x}_Q(k) \tag{3.47}$$

$$\textit{Desired signal} \qquad d(k) = d_I(k) + jd_Q(k) \tag{3.48}$$

$$\textit{Coefficient vector} \quad \mathbf{c}(k) = \mathbf{c}_I(k) + j\mathbf{c}_Q(k) \tag{3.49}$$

$$\textit{Output filter} \qquad y(k) = y_I(k) + jy_Q(k) \tag{3.50}$$

$$\textit{Estimation error} \quad e(k) = e_I(k) + je_Q(k) \tag{3.51}$$

Using the above definitions and considering separately real and imaginary terms, we derive the new equations:

$$y_I(k) = \mathbf{x}_I^T(k)\mathbf{c}_I(k) - \mathbf{x}_Q^T(k)\mathbf{c}_Q(k) \tag{3.52}$$

$$y_Q(k) = \mathbf{x}_I^T(k)\mathbf{c}_Q(k) + \mathbf{x}_Q^T(k)\mathbf{c}_I(k) \tag{3.53}$$

$$e_I(k) = d_I(k) - y_I(k) \tag{3.54}$$

$$e_Q(k) = d_Q(k) - y_Q(k) \tag{3.55}$$

$$\mathbf{c}_I(k+1) = \mathbf{c}_I(k) + \mu[e_I(k)\mathbf{x}_I(k) + e_Q(k)\mathbf{x}_Q(k)] \tag{3.56}$$

$$\mathbf{c}_Q(k+1) = \mathbf{c}_Q(k) - \mu[e_I(k)\mathbf{x}_Q(k) - e_Q(k)\mathbf{x}_I(k)] \tag{3.57}$$

Therefore a complex-valued LMS algorithm is equivalent to a set of two real-valued LMS algorithms with *cross-coupling*. This scheme is adopted in practice if only processors that use real arithmetic are available.

Conditions for convergence

Recalling that the objective of the LMS algorithm is to approximate the Wiener–Hopf solution, we introduce two criteria for convergence.

Convergence of the mean.

$$E[\mathbf{c}(k)] \xrightarrow[k\to\infty]{} \mathbf{c}_{opt} \tag{3.58}$$

$$E[e(k)] \xrightarrow[k\to\infty]{} 0 \tag{3.59}$$

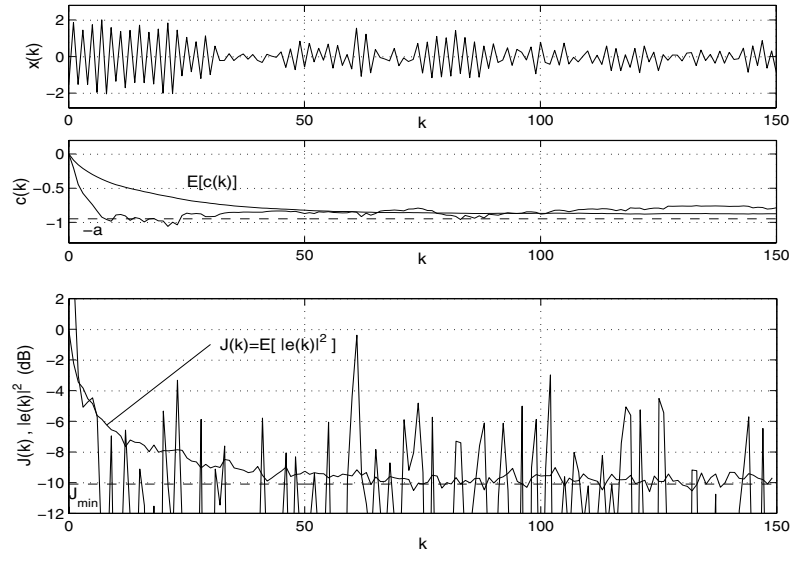


Figure 3.8. Realizations of input $\{x(k)\}$, coefficient $\{c(k)\}$ and squared error $\{|e(k)|^2\}$ for a one-coefficient predictor ($N = 1$), adapted according to the LMS algorithm.

In other words, it is required that the mean of the iterative solution converges to the Wiener–Hopf solution and the mean of the estimation error approaches zero. To show the weakness of this criterion, in Figure 3.8 we illustrate the results of a simple experiment for an input x given by a real-valued AR(1) random process:

$$x(k) = -\mathtt{a}\, x(k-1) + w(k) \qquad w(k) \in \mathcal{N}\left(0, \sigma_w^2\right) \tag{3.60}$$

where $\mathtt{a} = 0.95$, and $\sigma_x^2 = 1$ (i.e. $\sigma_w^2 = 0.097$).

For a first-order predictor we adapt the coefficient, $c(k)$, according to the LMS algorithm with $\mu = 0.1$. For $c(0) = 0$ and $k \geq 0$, we compute

1. Predictor output

$$y(k) = c(k)x(k-1) \tag{3.61}$$

2. Prediction error

$$e(k) = d(k) - y(k) = x(k) - y(k) \tag{3.62}$$

3. Coefficient update

$$c(k+1) = c(k) + \mu e(k)x(k-1) \tag{3.63}$$

In Figure 3.8, realizations of the processes $\{x(k)\}$, $\{c(k)\}$ and $\{|e(k)|^2\}$ are illustrated, as well as mean values $E[c(k)]$ and $J(k) = E[|e(k)|^2]$, estimated by averaging over 500 realizations; c_{opt} and J_{min} represent the Wiener–Hopf solution. From the plots in Figure 3.8 we observe two facts:

1. The random processes x and c exhibit a completely different behavior, for which they may be considered uncorrelated. It is interesting to observe that this hypothesis corresponds to assuming the filter input vectors, $\{\mathbf{x}(k)\}$, statistically independent. Actually, for small values of μ, $\mathbf{c}$ follows *mean statistical parameters* associated with the process x and not the process itself.

2. Convergence of the mean is an easily reachable objective. By itself, however, it does not yield the desired results, because the iterative solution $\mathbf{c}$ may exhibit very large oscillations around the optimum solution. A constraint on the amplitude of the oscillations must be introduced.

Convergence in the mean-square sense.

$$E[\|\mathbf{c}(k) - \mathbf{c}_{opt}\|^2] \xrightarrow[k\to\infty]{} \text{constant} \tag{3.64}$$

$$J(k) = E[|e(k)|^2] \xrightarrow[k\to\infty]{} J(\infty) \qquad \text{constant} \tag{3.65}$$

In other words, at convergence, both the mean of the coefficient error vector norm and the output mean-square error must be finite. The quantity $J(\infty) - J_{min} = J_{ex}(\infty)$ is the MSE in excess and represents the price paid for using a random adaptation algorithm for the coefficients rather than a deterministic one, such as the steepest-descent algorithm. In any case, we will see that the ratio $J_{ex}(\infty)/J_{min}$ can be made small by choosing a small adaptation gain μ. We note that the coefficients are obtained by averaging in time the quantity $\mu e(k)\mathbf{x}^*(k)$. Choosing a small μ the adaptation will be slow and the effect of the gradient noise on the coefficients will be strongly attenuated.

3.1.3 Convergence analysis of the LMS algorithm

We recall the following definitions.

1. Coefficient error vector

$$\Delta\mathbf{c}(k) = \mathbf{c}(k) - \mathbf{c}_{opt} \tag{3.66}$$

2. Optimum filter output error

$$e_{min}(k) = d(k) - \mathbf{x}^T(k)\mathbf{c}_{opt} \tag{3.67}$$

We also make the following assumptions.

1. $\mathbf{c}(k)$ is statistically independent of $\mathbf{x}(k)$.
2. The components of the coefficient error vector, transformed according to $\mathbf{U}^H$, $\boldsymbol{\nu}(k) = [\nu_1(k), \ldots, \nu_N(k)] = \mathbf{U}^H \Delta\mathbf{c}(k)$ (see (3.16)), are orthogonal:

$$E[\nu_i(k)\nu_n^*(k)] = 0 \qquad i \neq n \qquad i, n = 1, \ldots, N \tag{3.68}$$

   This assumption is justified by the observation that the linear transformation that orthogonalizes both $\mathbf{x}(k)$ (see (1.368)) and $\Delta\mathbf{c}(k)$ (see (3.16)) in the gradient algorithm is given by $\mathbf{U}^H$.
3. Fourth-order moments can be expressed as products of second-order moments (see (3.97)).

The adaptation equation of the LMS algorithm (3.41) can thus be written as

$$\Delta\mathbf{c}(k+1) = \Delta\mathbf{c}(k) + \mu\mathbf{x}^*(k)[d(k) - \mathbf{x}^T(k)\mathbf{c}(k)] \tag{3.69}$$

Adding and subtracting $\mathbf{x}^T(k)\mathbf{c}_{opt}$ to the terms within parentheses we obtain

$$\begin{aligned} \Delta\mathbf{c}(k+1) &= \Delta\mathbf{c}(k) + \mu\mathbf{x}^*(k)[e_{min}(k) - \mathbf{x}^T(k)\Delta\mathbf{c}(k)] \\ &= [\mathbf{I} - \mu\mathbf{x}^*(k)\mathbf{x}^T(k)]\Delta\mathbf{c}(k) + \mu e_{min}(k)\mathbf{x}^*(k) \end{aligned} \tag{3.70}$$

We note that $\Delta\mathbf{c}(k)$ depends only on the terms $\mathbf{x}(k-1), \mathbf{x}(k-2), \ldots$

Moreover, with the change of variables (3.16), observing (3.33), the cost function[6]

$$J(k) = \underset{\mathbf{x},\mathbf{c}}{E}[|e(k)|^2] \tag{3.71}$$

can be written as

$$J(k) = J_{min} + \sum_{i=1}^{N} \lambda_i E[|\nu_i(k)|^2] \tag{3.72}$$

Convergence of the mean

Taking the expectation of (3.70) and exploiting the statistical independence between $\mathbf{x}(k)$ and $\Delta\mathbf{c}(k)$, we get

$$E[\Delta\mathbf{c}(k+1)] = [\mathbf{I} - \mu E[\mathbf{x}^*(k)\mathbf{x}^T(k)]]E[\Delta\mathbf{c}(k)] + \mu E[e_{min}(k)\mathbf{x}^*(k)] \tag{3.73}$$

As $E[\mathbf{x}^*(k)\mathbf{x}^T(k)] = \mathbf{R}$ and the second term on the right-hand side of (3.73) vanishes for the orthogonality property (2.36) of the optimum filter, we obtain the same equation as in the case of the *steepest descent* algorithm:

$$E[\Delta\mathbf{c}(k+1)] = [\mathbf{I} - \mu\mathbf{R}]E[\Delta\mathbf{c}(k)] \tag{3.74}$$

---

[6] $\underset{\mathbf{x},\mathbf{c}}{E}$ denotes the expectation with respect to $\mathbf{x}$ and $\mathbf{c}$.

Consequently, for the LMS algorithm the convergence of the mean is obtained if

$$0 < \mu < \frac{2}{\lambda_{max}} \tag{3.75}$$

Observing (3.25) and (3.32), and choosing the value of $\mu = 2/(\lambda_{max} + \lambda_{min})$, the vector $E[\Delta\mathbf{c}(k)]$ is reduced at each iteration at least by the factor $(\lambda_{max} - \lambda_{min})$ / $(\lambda_{max} + \lambda_{min})$. We can therefore assume that $E[\Delta\mathbf{c}(k)]$ becomes rapidly negligible with respect to the mean-square error during the process of convergence.

Convergence in the mean-square sense (real scalar case)

The assumption of a filter with real-valued input and only one coefficient $c(k)$ allows us to deduce by a simple analysis important properties of the convergence in the mean-square sense. From

$$\Delta c(k+1) = (1 - \mu x^2(k))\Delta c(k) + \mu e_{min}(k)x(k) \tag{3.76}$$

because $x(k)$ and $\Delta c(k)$ are assumed to be statistically independent and $x(k)$ is orthogonal to $e_{min}(k)$, and *assuming* furthermore that $x(k)$ and $e_{min}(k)$ are statistically independent, we get

$$\begin{aligned} E[\Delta c^2(k+1)] = {} & E[1 + \mu^2 x^4(k) - 2\mu x^2(k)]E[\Delta c^2(k)] + \mu^2 E[x^2(k)]J_{min} \\ & + 2E[(1 - \mu x^2(k))\Delta c(k)\mu x(k)e_{min}(k)] \end{aligned} \tag{3.77}$$

where the last term vanishes, as $\Delta c(k)$ has zero mean and is statistically independent of all other terms. Assuming[7] moreover, $E[x^4(k)] = E[x^2(k)x^2(k)] = \mathrm{r}_x^2(0)$, (3.77) becomes

$$E[\Delta c^2(k+1)] = (1 + \mu^2 \mathrm{r}_x^2(0) - 2\mu \mathrm{r}_x(0))E[\Delta c^2(k)] + \mu^2 \mathrm{r}_x(0) J_{min} \tag{3.78}$$

Let

$$\gamma = 1 + \mu^2 \mathrm{r}_x^2(0) - 2\mu \mathrm{r}_x(0) \tag{3.79}$$

whose behavior as a function of μ is given in Figure 3.9. Then for the convergence of the difference equation (3.78) it must be $|\gamma| < 1$. Consequently μ must satisfy the condition

$$0 < \mu < \frac{2}{\mathrm{r}_x(0)} \tag{3.80}$$

Moreover, assuming $\mu \mathrm{r}_x(0) \ll 1$, we get

$$\begin{aligned} E[\Delta c^2(\infty)] &= \frac{\mu^2 \mathrm{r}_x(0)}{\mu \mathrm{r}_x(0)(2 - \mu \mathrm{r}_x(0))} J_{min} \\ &= \frac{\mu}{(2 - \mu \mathrm{r}_x(0))} J_{min} \\ &\simeq \frac{\mu}{2} J_{min} \end{aligned} \tag{3.81}$$

[7] In other texts the Gaussian assumption is made, whereby $E[x^4(k)] = 3\mathrm{r}_x^2(0)$. The conclusions of the analysis are similar.

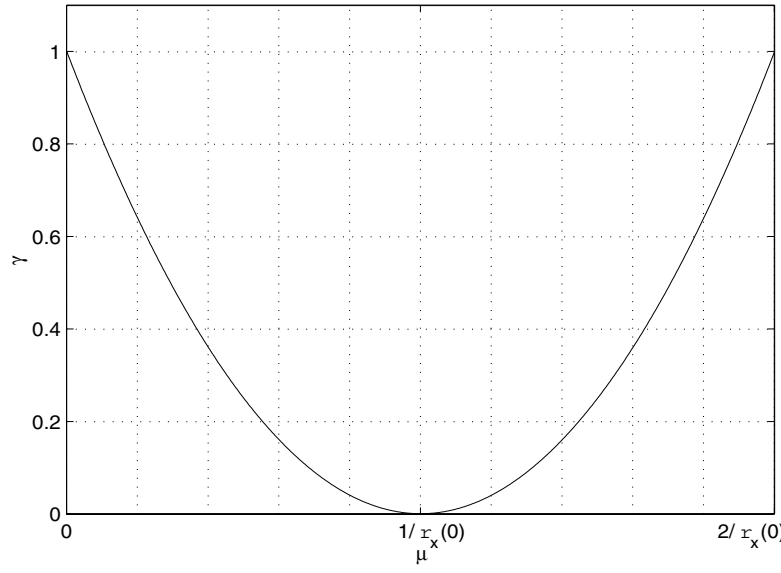


Figure 3.9. Plot of γ as a function of μ.

Likewise, from

$$e(k) = d(k) - x(k)c(k) = e_{min}(k) - \Delta c(k)x(k) \tag{3.82}$$

it turns out

$$E[e^2(k)] = E[e_{min}^2(k)] + E[x^2(k)]E[\Delta c^2(k)] \tag{3.83}$$

that is

$$J(k) = J_{min} + \mathrm{r}_x(0)E[\Delta c^2(k)] \tag{3.84}$$

In particular, for $k \to \infty$, we have

$$J(\infty) \simeq J_{min} + \mathrm{r}_x(0)\frac{\mu}{2}J_{min} \tag{3.85}$$

The relative MSE deviation, or *misadjustment*, is:

$$\mathrm{MSD} = \frac{J(\infty) - J_{min}}{J_{min}} = \frac{J_{ex}(\infty)}{J_{min}} = \mathrm{r}_x(0)\frac{\mu}{2} \tag{3.86}$$

Convergence in the mean-square sense (general case)

The convergence theory given here follows the method developed in [3]. With the change of variables (3.16), (3.70) becomes

$$\boldsymbol{\nu}(k+1) = [\mathbf{I} - \mu\mathbf{U}^H\mathbf{x}^*(k)\mathbf{x}^T(k)\mathbf{U}]\boldsymbol{\nu}(k) + \mu e_{min}(k)\mathbf{U}^H\mathbf{x}^*(k) \tag{3.87}$$

Let us define

$$\tilde{\mathbf{x}}(k) = [\tilde{x}_1(k), \ldots, \tilde{x}_N(k)]^T = \mathbf{U}^T \mathbf{x}(k) \tag{3.88}$$

and

$$\boldsymbol{\Omega} = \tilde{\mathbf{x}}^*(k)\tilde{\mathbf{x}}^T(k) = \mathbf{U}^H \mathbf{x}^*(k)\mathbf{x}^T(k)\mathbf{U} \tag{3.89}$$

$N \times N$ matrix with elements

$$\boldsymbol{\Omega}(i, n) = \tilde{x}_i^*(k)\tilde{x}_n(k) \qquad i, n = 1, \ldots, N \tag{3.90}$$

From (1.368) we get

$$E[\boldsymbol{\Omega}] = \boldsymbol{\Lambda} \tag{3.91}$$

hence the components $\{\tilde{x}_i(k)\}$ are mutually orthogonal. Then (3.87) becomes

$$\boldsymbol{\nu}(k+1) = [\mathbf{I} - \mu\boldsymbol{\Omega}]\boldsymbol{\nu}(k) + \mu e_{min}(k)\tilde{\mathbf{x}}^*(k) \tag{3.92}$$

Recalling Assumption 1 of the convergence analysis, and assuming $e_{min}(k)$ and $\mathbf{x}(k)$ are not only orthogonal but also statistically independent, and consequently $e_{min}(k)$ independent of $\tilde{\mathbf{x}}(k)$, the correlation matrix of $\boldsymbol{\nu}^*$ at the instant $k+1$ can be expressed as

$$E[\boldsymbol{\nu}^*(k+1)\boldsymbol{\nu}^T(k+1)] = E[(\mathbf{I} - \mu\boldsymbol{\Omega}^*)\boldsymbol{\nu}^*(k)\boldsymbol{\nu}^T(k)(\mathbf{I} - \mu\boldsymbol{\Omega}^T)] + \mu^2 J_{min} E[\boldsymbol{\Omega}^*] \tag{3.93}$$

Observing (3.91), the second term on the right-hand side of (3.93) is equal to $\mu^2 J_{min}\boldsymbol{\Lambda}$. Moreover, considering that $\boldsymbol{\nu}(k)$ is statistically independent of $\mathbf{x}(k)$ and $\tilde{\mathbf{x}}(k)$, the first term can be written as

$$\begin{aligned} \underset{\mathbf{x},\boldsymbol{\nu}}{E}[(\mathbf{I} - \mu\boldsymbol{\Omega}^*)\boldsymbol{\nu}^*(k)\boldsymbol{\nu}^T(k)(\mathbf{I} - \mu\boldsymbol{\Omega}^T)] &= \underset{\mathbf{x}}{E}[(\mathbf{I} - \mu\boldsymbol{\Omega}^*)\underset{\boldsymbol{\nu}}{E}]\boldsymbol{\nu}^*(k)\boldsymbol{\nu}^T(k)[(\mathbf{I} - \mu\boldsymbol{\Omega}^T)] \\ &= E[(\mathbf{I} - \mu\boldsymbol{\Omega}^*)(\mathbf{I} - \mu\boldsymbol{\Omega}^T)]E[\boldsymbol{\nu}^*(k)\boldsymbol{\nu}^T(k)] \end{aligned} \tag{3.94}$$

Recalling Assumption 2 of the convergence analysis, we find that the matrix $E[\boldsymbol{\nu}^*(k)\ \boldsymbol{\nu}^T(k)]$ is diagonal, with elements on the main diagonal given by the vector

$$\boldsymbol{\eta}^T(k) = [\eta_1(k), \ldots, \eta_N(k)] = [E[|\nu_1(k)|^2], \ldots, E[|\nu_N(k)|^2]] \tag{3.95}$$

Observing (3.90), the elements with indices (i, i) of the matrix expressed by (3.94) are given by

$$\begin{aligned} &E\left[\eta_i(k) + \mu^2 \sum_{n=1}^{N} \boldsymbol{\Omega}^*(i, n)\boldsymbol{\Omega}(n, i)\eta_n(k) - 2\mu\boldsymbol{\Omega}(i, i)\eta_i(k)\right] \\ &\quad = \eta_i(k) + \mu^2 \sum_{n=1}^{N} E[|\tilde{x}_i(k)|^2|\tilde{x}_n(k)|^2]\eta_n(k) - 2\mu\lambda_i\eta_i(k) \\ &\quad = \eta_i(k) + \mu^2 \sum_{n=1}^{N} \lambda_i\lambda_n\eta_n(k) - 2\mu\lambda_i\eta_i(k) \end{aligned} \tag{3.96}$$

where, recalling Assumption 3 of the convergence analysis,

$$E[|\tilde{x}_i(k)|^4] = E[|\tilde{x}_i(k)|^2]E[|\tilde{x}_i(k)|^2] = \lambda_i^2 \tag{3.97}$$

Let

$$\boldsymbol{\lambda}^T = [\lambda_1, \ldots, \lambda_N] \tag{3.98}$$

be the vector of eigenvalues of $\mathbf{R}$, and

$$\mathbf{B} = \begin{bmatrix} (1-\mu\lambda_1)^2 & \mu^2\lambda_1\lambda_2 & \ldots & \mu^2\lambda_1\lambda_N \\ \mu^2\lambda_2\lambda_1 & (1-\mu\lambda_2)^2 & \ldots & \mu^2\lambda_2\lambda_N \\ \vdots & \vdots & \ddots & \vdots \\ \mu^2\lambda_N\lambda_1 & \mu^2\lambda_N\lambda_2 & \ldots & (1-\mu\lambda_N)^2 \end{bmatrix} \tag{3.99}$$

$N \times N$ symmetric positive definite matrix with positive elements. From (3.93) and (3.96), we obtain the relation

$$\boldsymbol{\eta}(k+1) = \mathbf{B}\boldsymbol{\eta}(k) + \mu^2 J_{min}\boldsymbol{\lambda} \tag{3.100}$$

Using the properties of $\mathbf{B}$, the general decomposition (2.175) becomes

$$\mathbf{B} = \mathbf{V}\,\mathrm{diag}(\sigma_1, \ldots, \sigma_N)\mathbf{V}^H \tag{3.101}$$

where $\{\sigma_i\}$ denote the eigenvalues of $\mathbf{B}$, and $\mathbf{V}$ is the unitary matrix formed by the eigenvectors $\{\mathbf{v}_i\}$ of $\mathbf{B}$. After simple steps, similar to those applied to get (3.25) from (3.13), and using the relation

$$N r_x(0) = \mathrm{tr}[\mathbf{R}] = \sum_{i=1}^{N} \lambda_i \tag{3.102}$$

the solution of the vector difference equation (3.100) is given by:

$$\boldsymbol{\eta}(k) = \sum_{i=1}^{N} \mathcal{K}_i \sigma_i^k \mathbf{v}_i + \frac{\mu J_{min}}{2 - \mu N r_x(0)}\mathbf{1} \qquad k \geq 0 \tag{3.103}$$

where $\mathbf{1} = [1, 1, \ldots, 1]^T$. In (3.103) the constants $\{\mathcal{K}_i\}$ are determined by the initial conditions

$$\mathcal{K}_i = \mathbf{v}_i^H \left(\boldsymbol{\eta}(0) - \frac{\mu J_{min}}{2 - \mu N r_x(0)}\mathbf{1} \right) \qquad i = 1, \ldots, N \tag{3.104}$$

where the components of $\boldsymbol{\eta}(0)$ depend on the choice of $\mathbf{c}(0)$ according to (3.95) and (3.16):

$$\eta_n(0) = E[|\nu_n(0)|^2] = E[|\mathbf{u}_n^H \Delta\mathbf{c}(0)|^2] \qquad n = 1, \ldots, N \tag{3.105}$$

Using (3.95) and (3.98), the cost function $J(k)$ given by (3.72) becomes

$$J(k) = J_{min} + \boldsymbol{\lambda}^T \boldsymbol{\eta}(k) \tag{3.106}$$

Substituting the result (3.103) in (3.106), we find

$$J(k) = \sum_{i=1}^{N} \mathcal{C}_i \sigma_i^k + \frac{2}{2 - \mu N r_x(0)} J_{min} \tag{3.107}$$

where

$$\mathcal{C}_i = \mathcal{K}_i \boldsymbol{\lambda}^T \mathbf{v}_i \tag{3.108}$$

The first term on the right-hand side of (3.107) describes the convergence behavior of the mean-square error, whereas the second term gives the steady-state value. Therefore further investigation of the properties of the matrix $\mathbf{B}$ will allow us to characterize the transient behavior of J.

Basic results

From the above convergence analysis, we will obtain some fundamental properties of the LMS algorithm.

1. The transient behavior of J does not exhibit oscillations; this result is obtained by observing the properties of the eigenvalues of $\mathbf{B}$.

2. The LMS algorithm converges if the adaptation gain μ satisfies the condition

$$0 < \mu < \frac{2}{\text{statistical power of input vector}} \tag{3.109}$$

In fact the adaptive system is stable and J converges to a constant steady-state value under the conditions $|\sigma_i| < 1$, $i = 1, \ldots, N$. This happens if

$$0 < \mu < \frac{2}{N r_x(0)} \tag{3.110}$$

Conversely, if μ satisfies (3.110), from (3.99) the sum of the elements of the i-th row of $\mathbf{B}$ satisfies

$$\sum_{n=1}^{N} [\mathbf{B}]_{i,n} = 1 - \mu \lambda_i (2 - \mu N r_x(0)) < 1 \tag{3.111}$$

A matrix with these properties and whose elements are all positive has eigenvalues with absolute value less than one. In particular, being

$$\begin{aligned} \sum_{i=1}^{N} \lambda_i &= \text{tr}[\mathbf{R}] \\ &= N r_x(0) \\ &= \sum_{i=0}^{N-1} E[|x(k-i)|^2] \\ &= \text{statistical power of input vector} \end{aligned} \tag{3.112}$$

the equation (3.110) becomes (3.109).

We recall that, for convergence of the mean, it must be

$$0 < \mu < \frac{2}{\lambda_{max}} \tag{3.113}$$

but since

$$\sum_{i=1}^{N} \lambda_i > \lambda_{max} \tag{3.114}$$

the condition for convergence in the mean-square implies convergence of the mean. In other words, convergence in the mean-square imposes a tighter bound to allowable values of the adaptation gain μ than that imposed by convergence of the mean (3.113). The new bound depends on the number of coefficients, rather than on the eigenvalue distribution of the matrix $\mathbf{R}$. The relation (3.110) can be intuitively explained noting that, for a given value of μ, an increase in the number of coefficients causes an increase in the excess mean-square error due to fluctuations of the coefficients around the mean value. Increasing the number of coefficients without reducing the value of μ leads to instability of the adaptive system.

3. Equation (3.107) reveals a simple relation between the adaptation gain μ and the value $J(k)$ in the steady state ($k \to \infty$):

$$J(\infty) = \frac{2}{2 - \mu N r_x(0)} J_{min} \tag{3.115}$$

from which the excess MSE is given by

$$J_{ex}(\infty) = J(\infty) - J_{min} = \frac{\mu N r_x(0)}{2 - \mu N r_x(0)} J_{min} \tag{3.116}$$

and the misadjustment has the expression

$$\text{MSD} = \frac{J_{ex}(\infty)}{J_{min}} = \frac{\mu N r_x(0)}{2 - \mu N r_x(0)} \simeq \frac{\mu}{2} N r_x(0) \tag{3.117}$$

for $\mu \ll 2/(N r_x(0))$.

Observations

1. For $\mu \to 0$ all eigenvalues of $\mathbf{B}$ tend toward 1.

2. As shown below, a small eigenvalue of the matrix $\mathbf{R}$ ($\lambda_i \to 0$) determines a large time constant for one of the convergence modes of J, as $\sigma_i \to 1$. However, a large time constant of one of the modes implies a low probability that the corresponding term contributes significantly to the mean-square error.

   Proof. If $\lambda_i = 0$, the i-th row of $\mathbf{B}$ becomes $(0, \ldots, 0, [\mathbf{B}]_{i,i} = 1, 0, \ldots, 0)$.

   Consequently $\sigma_i = 1$ and $\mathbf{v}_i^T = (0, \ldots, 0, \ v_{i,i} = 1, 0, \ldots, 0)$. As $\boldsymbol{\lambda}^T \mathbf{v}_i = 0$, from (3.108) we get $\mathcal{C}_i = 0$.

It is generally correct to state that a large *eigenvalue spread* of $\mathbf{R}$ determines a slow convergence of J to the steady state. However, the fact that modes with a large time constant usually contribute to J less than the modes that converge more rapidly, mitigates this effect. Therefore the convergence of J is less influenced by the eigenvalue spread of $\mathbf{R}$ than would be the convergence of $\Delta\mathbf{c}(k)$.

3. If all eigenvalues of the matrix $\mathbf{R}$ are equal,[8] $\lambda_i = r_x(0), i = 1, \ldots, N$, the maximum eigenvalue of the matrix $\mathbf{B}$ is given by

$$\sigma_{i_{max}} = 1 - \mu r_x(0)(2 - \mu N r_x(0)) \tag{3.118}$$

The remaining eigenvalues of $\mathbf{B}$ do not influence the transient behavior of J, since $\mathcal{C}_i = 0, i \neq i_{max}$.

Proof. It is easily verified that $\sigma_{i_{max}}$ is an eigenvalue of $\mathbf{B}$ and $\mathbf{v}_{i_{max}}^T = N^{-1/2}$ $[1, 1, \ldots, 1]$ is the corresponding eigenvector. Moreover, the Perron–Frobenius theorem affirms that the maximum eigenvalue of a positive matrix $\mathbf{B}$ is a positive real number and that the elements of the corresponding eigenvector are positive real numbers [4]. Since all elements of $\mathbf{v}_{i_{max}}$ are positive, it follows that $\sigma_{i_{max}}$ is the maximum eigenvalue of $\mathbf{B}$. Moreover, because $\mathbf{v}_{i_{max}}$ is parallel to $\boldsymbol{\lambda}^T$, the other eigenvectors of $\mathbf{B}$ are orthogonal to $\boldsymbol{\lambda}$. Hence $\mathcal{C}_i = 0, i \neq i_{max}$.

4. If all eigenvalues of the matrix $\mathbf{R}$ are equal, $\lambda_i = r_x(0), i = 1, \ldots, N$, combining (3.107) with the (3.118) and considering a time varying adaptation gain $\mu(k)$, we obtain

$$J(k+1) \simeq [1 - \mu(k) r_x(0)(2 - \mu(k) N r_x(0))] J(k) + 2\mu(k) r_x(0) J_{min} \tag{3.119}$$

The maximum convergence rate of J is obtained for the adaptation gain

$$\mu_{opt}(k) = \frac{1}{N r_x(0)} \frac{J(k) - J_{min}}{J(k)} \tag{3.120}$$

As the condition $J(k) \gg J_{min}$ is normally verified at the beginning of the iteration process, it results

$$\mu_{opt}(k) \simeq \frac{1}{N r_x(0)} \tag{3.121}$$

and

$$J(k+1) \simeq \left(1 - \frac{1}{N}\right) J(k) \tag{3.122}$$

[8] This occurs, for example, if the input x is white noise.

We note that the number of iterations required to reduce the value of $J(k)$ by one order of magnitude is approximately $2.3N$.

5. Thus (3.103) indicates that at steady state all elements of $\boldsymbol{\eta}$ become equal. Consequently, recalling Assumption 2 of the convergence analysis, in steady state the filter coefficients are uncorrelated random variables with equal variance. The mean corresponds to the optimum vector $\mathbf{c}_{opt}$.

6. In case the LMS algorithm is used to estimate the coefficients of a system that slowly changes in time, the adaptation gain μ has a lower bound larger than 0. In this case, the value of J "in steady state" varies with time and is given by the sum of three terms:

$$J_{tot}(k) = J_{min}(k) + J_{ex}(\infty) + J_{\ell} \tag{3.123}$$

where $J_{min}(k)$ corresponds to the Wiener–Hopf solution, $J_{ex}(\infty)$ depends instead on the LMS algorithm and is directly proportional to μ, and J_{ℓ} depends on the ability of the LMS algorithm to track the system variations and expresses the lag error in the estimate of the coefficients. It turns out that J_{ℓ} is inversely proportional to μ. Therefore, for time varying systems μ must be chosen as a compromise between J_{ex} and J_{ℓ} and cannot be arbitrarily small [5, 6, 7].

Final remarks

1. The LMS algorithm is easy to implement.

2. The relatively slow convergence is influenced by μ, the number of coefficients and the eigenvalues of $\mathbf{R}$. In particular it must be

$$0 < \mu < \frac{2}{N r_x(0)} = \frac{2}{\text{statistical power of input vector}} \tag{3.124}$$

3. Choosing a small μ results in a slow adaptation, and in a small excess MSE at convergence $J_{ex}(\infty)$. For a large μ, instead, the adaptation is fast at the expense of a large $J_{ex}(\infty)$.

4. $J_{ex}(\infty)$ is determined by the large eigenvalues of $\mathbf{R}$, whereas the speed of convergence of $E[\mathbf{c}(k)]$ is imposed by λ_{min}.
If the eigenvalue spread of $\mathbf{R}$ increases, the convergence of $E[\mathbf{c}(k)]$ becomes slower; on the other hand the convergence of $J(k)$ is less sensitive to this parameter. Note, however, that the convergence behavior depends on the initial condition $\mathbf{c}(0)$ [8].

3.1.4 Other versions of the LMS algorithm

In the previous section, the basic version of the LMS algorithm was presented. We now give a brief introduction to other versions that can be used for various applications.

Leaky LMS

The leaky LMS algorithm is a variant of the LMS algorithm that uses the following adaptation equation:

$$\mathbf{c}(k+1) = (1 - \mu\alpha)\mathbf{c}(k) + \mu e(k)\mathbf{x}^*(k) \tag{3.125}$$

with $0 < \alpha \ll r_x(0)$. This equation corresponds to the following cost function:

$$J(k) = E[|e(k)|^2] + \alpha E[\|\mathbf{c}(k)\|^2] \tag{3.126}$$

where, as usual,

$$e(k) = d(k) - \mathbf{c}^T(k)\mathbf{x}(k) \tag{3.127}$$

In other words, the cost function includes an additional term proportional to the norm of the vector of coefficients. In steady state we get

$$\lim_{k\to\infty} E[\mathbf{c}(k)] = (\mathbf{R} + \alpha\mathbf{I})^{-1}\mathbf{p} \tag{3.128}$$

It is interesting to give another interpretation to what has been stated. Observing (3.128), the application of the leaky LMS algorithm results in the addition of a small constant α to the terms on the main diagonal of the correlation matrix of the input process; one obtains the same result by summing white noise with statistical power α to the input process. Both approaches are useful to make irreversible an ill-conditioned matrix $\mathbf{R}$, or to accelerate the convergence of the LMS algorithm. It is usually sufficient to choose α two or three orders of magnitude smaller than $r_x(0)$, in order not to modify substantially the original Wiener–Hopf solution. Therefore the leaky LMS algorithm is used in cases where the Wiener problem is ill-conditioned, and multiple solutions exist.

Sign algorithm

There are adaptation equations that are simpler to implement, at the expense of a lower speed of convergence, for the same $J(\infty)$. Three versions are:[9]

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \mu \begin{cases} \operatorname{sgn}(e(k))\mathbf{x}^*(k) \\ e(k)\operatorname{sgn}(\mathbf{x}^*(k)) \\ \operatorname{sgn}(e(k))\operatorname{sgn}(\mathbf{x}^*(k)) \end{cases} \tag{3.130}$$

Note that the first version has as objective the minimization of the cost function

$$J(k) = E[|e(k)|] \tag{3.131}$$

---

[9] The sign of a vector of complex-valued elements is defined as follows:

$$\operatorname{sgn}(\mathbf{x}(k)) = [\operatorname{sgn}(x_I(k)) + j\operatorname{sgn}(x_Q(k)), \ldots, \operatorname{sgn}(x_I(k-N+1)) + j\operatorname{sgn}(x_Q(k-N+1))] \tag{3.129}$$

Sigmoidal algorithm

As extensions of the algorithms given in (3.128), the following adaptation equations may be considered:

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \mu \begin{cases} \varphi(e(k))\mathbf{x}^*(k) \\ e(k)\varphi(\mathbf{x}^*(k)) \\ \varphi(e(k))\varphi(\mathbf{x}^*(k)) \end{cases} \tag{3.132}$$

where $\varphi(a)$ is the *sigmoidal* function (see Figure 3.10) [9]:

$$\varphi(a) = \tanh\left(\frac{\beta a}{2}\right) = \frac{1 - e^{-\beta a}}{1 + e^{-\beta a}} \tag{3.133}$$

where β is a positive parameter.

There also exists a piecewise linear version of the sigmoidal function defined as

$$\varphi(a) = \begin{cases} -1 & \text{per } a < -A \\ \dfrac{a}{A} & \text{per } -A \le a \le A \\ 1 & \text{per } a > A \end{cases} \tag{3.134}$$

where A is a positive parameter.

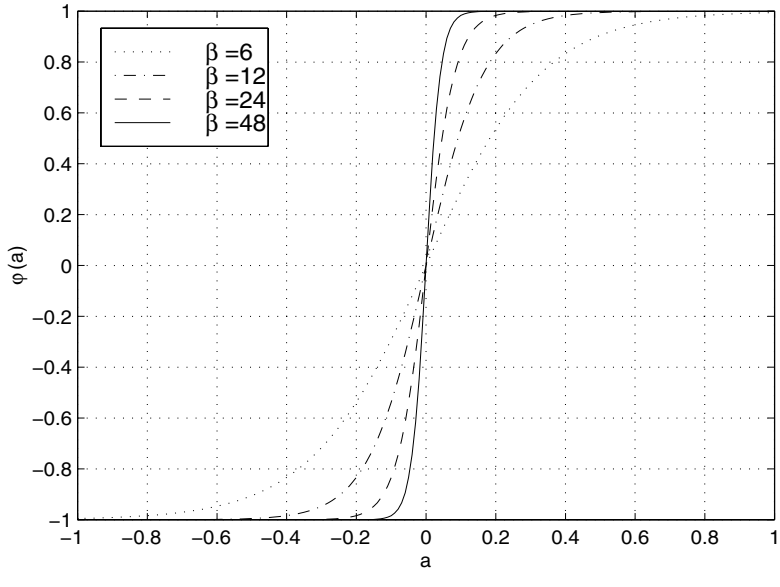


Figure 3.10. Sigmoidal function for various values of the parameter β.

Normalized LMS

In the LMS algorithm, if some $\mathbf{x}(k)$ assume large values, the adaptation algorithm is affected by strong noise in the gradient. This problem can be overcome by choosing an adaptation gain μ of the type:

$$\mu = \frac{\tilde{\mu}}{p + \hat{\mathrm{M}}_{\mathbf{x}}(k)} \tag{3.135}$$

where $0 < \tilde{\mu} < 2$, and

$$\hat{\mathrm{M}}_{\mathbf{x}}(k) = \|\mathbf{x}\|^2 = \sum_{i=0}^{N-1} |x(k-i)|^2 \tag{3.136}$$

or, alternatively,

$$\hat{\mathrm{M}}_{\mathbf{x}}(k) = N\hat{\mathrm{M}}_x(k) \tag{3.137}$$

where $\hat{\mathrm{M}}_x(k)$ is the estimate of the statistical power of $x(k)$. A simple estimate is obtained by the iterative equation (see (1.468)):

$$\hat{\mathrm{M}}_x(k) = a\hat{\mathrm{M}}_x(k-1) + (1-a)|x(k)|^2 \tag{3.138}$$

where $0 < a < 1$, with time constant given by

$$\tau = \frac{-1}{\ln a} \simeq \frac{1}{1-a} \tag{3.139}$$

for $a \simeq 1$. In (3.135), p is a positive parameter that is introduced to avoid the denominator becoming too small; typically

$$p \simeq \frac{1}{10}\mathrm{M}_{\mathbf{x}} \tag{3.140}$$

The normalized LMS algorithm has a speed of convergence that is potentially higher than the standard algorithm, for uncorrelated as well as correlated input signals [10]. To be able to apply the normalized algorithm, however, some knowledge of the input process is necessary, in order to assign the values of M_x and p so that the adaptation process does not become unstable.

Variable adaptation gain

In the following variants of the LMS algorithms the coefficient μ varies with time.

1. *Two values of* μ.
    a. Initially a large value of μ is chosen for fast convergence, for example, $\mu = 1/(N\mathrm{r}_x(0))$.
    b. Subsequently μ is reduced to achieve a smaller $J(\infty)$.

For a choice of μ of the type

$$\mu = \begin{cases} \mu_1 & \text{per } 0 \leq k \leq K_1 \\ \mu_2 & \text{per } k \geq K_1 \end{cases} \tag{3.141}$$

the behavior of J will be illustrated in Figure 3.11.

2. *Decreasing* μ. For a time-invariant system, the adaptation gain usually selected for application with the sign algorithm (3.130) is given by

$$\mu(k) = \frac{\mu_1}{\mu_2 + k} \qquad k \geq 0 \tag{3.142}$$

3. μ *proportional to* $e(k)$. The following expression of μ is used:

$$\mu(k+1) = \alpha_1 \mu(k) + \alpha_2 |e(k)|^2 \tag{3.143}$$

with μ limited to the range $[\mu_{min}, \mu_{max}]$. Typical values are $\alpha_1 \simeq 1$ and $\alpha_2 \ll 1$.

4. *Vector of values of* μ. Let $\boldsymbol{\mu}^T = [\mu_0, \ldots, \mu_{N-1}]$; two approaches are possible.

   a. Initially larger values μ_i are chosen in correspondence of those coefficients c_i that have larger amplitude.

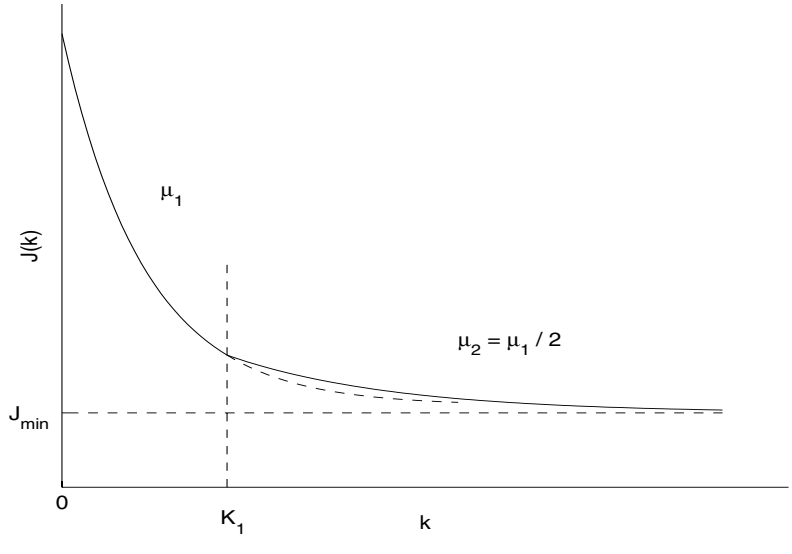


Figure 3.11. Behavior of $J(k)$ obtained by using two values of μ.

b. μ_i changes with time following the rule

$$\mu_i(k+1) = \begin{cases} \mu_i(k)\dfrac{1}{\alpha} & \text{if the } i\text{-th component of the gradient } \textit{has always} \text{ changed sign in the last } m_0 \text{ iterations} \\ \mu_i(k)\alpha & \text{if the } i\text{-th component of the gradient } \textit{has never} \text{ changed sign in the last } m_1 \text{ iterations} \end{cases} \tag{3.144}$$

with μ limited to the range $[\mu_{min}, \mu_{max}]$. Typical values are $m_0, m_1 \in \{1, 3\}$ and $\alpha = 2$.

LMS for lattice filters

We saw in Section 2.2.1 that filters with a lattice structure have some interesting properties. The application of the LMS algorithm for lattice filters, however, is not as simple as for transversal filters. For this reason such filters are now rarely used, although they were popular in the past when fast hardware implementations were rather costly. For the study of the LMS algorithm for lattice filters we refer the reader to [11, 12].

3.1.5 Example of application: the predictor

We consider a real AR(2) process of unit power, described by the equation

$$x(k) = -\mathtt{a}_1 x(k-1) - \mathtt{a}_2 x(k-2) + w(k) \tag{3.145}$$

with w additive white Gaussian noise (AWGN), and

$$\mathtt{a}_1 = -1.3 \qquad \mathtt{a}_2 = 0.995 \tag{3.146}$$

From (1.547), the roots of $A(z)$ are given by $\varrho e^{\pm\varphi_0}$, where

$$\varrho = \sqrt{\mathtt{a}_2} = 0.997 \tag{3.147}$$

and

$$\varphi_0 = \cos^{-1}\left(-\frac{\mathtt{a}_1}{2\varrho}\right) = 2.28\,\text{rad} \tag{3.148}$$

Being $\mathtt{r}_x(0) = \sigma_x^2 = 1$, from the (1.552) we find that the statistical power of w is given by

$$\sigma_w^2 = \frac{1-\mathtt{a}_2}{1+\mathtt{a}_2}[(1+\mathtt{a}_2)^2 - \mathtt{a}_1^2] = 0.0057 = -22.4\,\text{dB} \tag{3.149}$$

We construct a predictor for x of order $N = 2$ with coefficients $\mathbf{c}^T = [c_1, c_2]$, as illustrated in Figure 3.12, using the LMS algorithm and some of its variants [13]. From (2.83) we expect to find in steady state

$$\mathbf{c} \simeq -\mathbf{a} \tag{3.150}$$

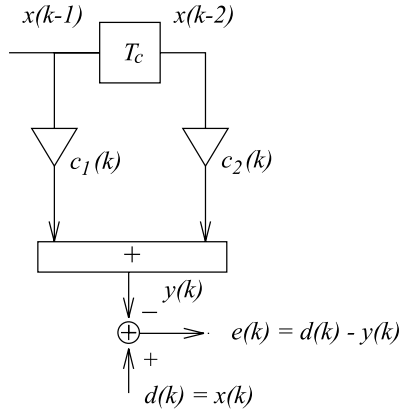


Figure 3.12. Predictor of order $N = 2$.

that is $c_1 \simeq -\mathtt{a}_1$, $c_2 \simeq -\mathtt{a}_2$, and $\sigma_e^2 \simeq \sigma_w^2$. In any case, the predictor output is given by

$$\begin{aligned} y(k) &= \mathbf{c}^T(k)\mathbf{x}(k-1) \\ &= c_1(k)x(k-1) + c_2(k)\,x(k-2) \end{aligned} \tag{3.151}$$

with prediction error

$$e(k) = x(k) - y(k) \tag{3.152}$$

For the predictor of Figure 3.12 we now consider various versions of the adaptive LMS algorithm and their relative performance.

Example 3.1.1 (Standard LMS)
The equation for updating the coefficient vector is

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \mu e(k)\mathbf{x}(k-1) \tag{3.153}$$

Convergence curves are plotted in Figure 3.13 for a single realization and for the mean (estimated over 500 realizations) of the coefficients and of the squared prediction error, for $\mu = 0.04$. In Figure 3.14 a comparison is made between the curves of convergence of the mean for three values of μ. We observe that, by decreasing μ, the excess error decreases, thus giving a more accurate solution, but the convergence time increases.

Example 3.1.2 (Leaky LMS)
The equation for updating the coefficient vector is

$$\mathbf{c}(k+1) = (1 - \mu\alpha)\mathbf{c}(k) + \mu e(k)\mathbf{x}(k-1) \tag{3.154}$$

Convergence curves are plotted in Figure 3.15 for a single realization and for the mean (estimated over 500 realizations) of the coefficients and of the squared prediction error,

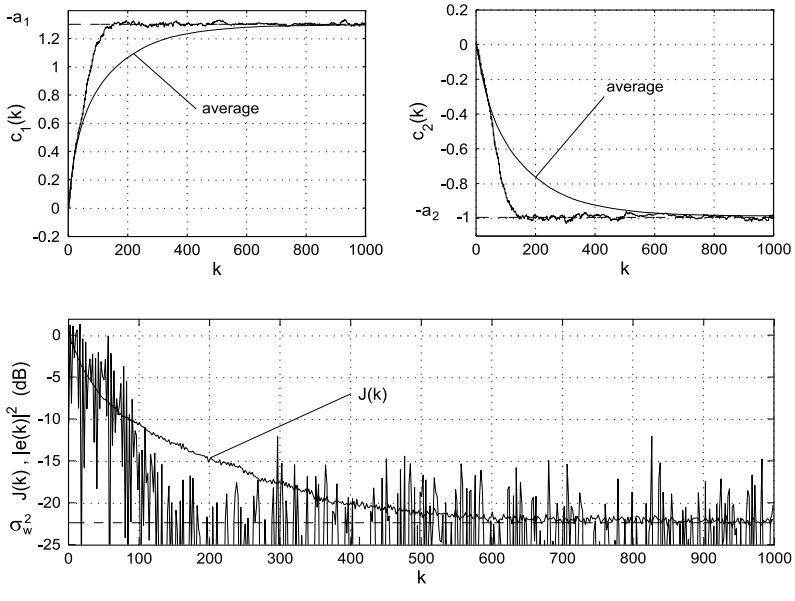


Figure 3.13. Convergence curves for the predictor of order $N = 2$, obtained by the standard LMS algorithm.

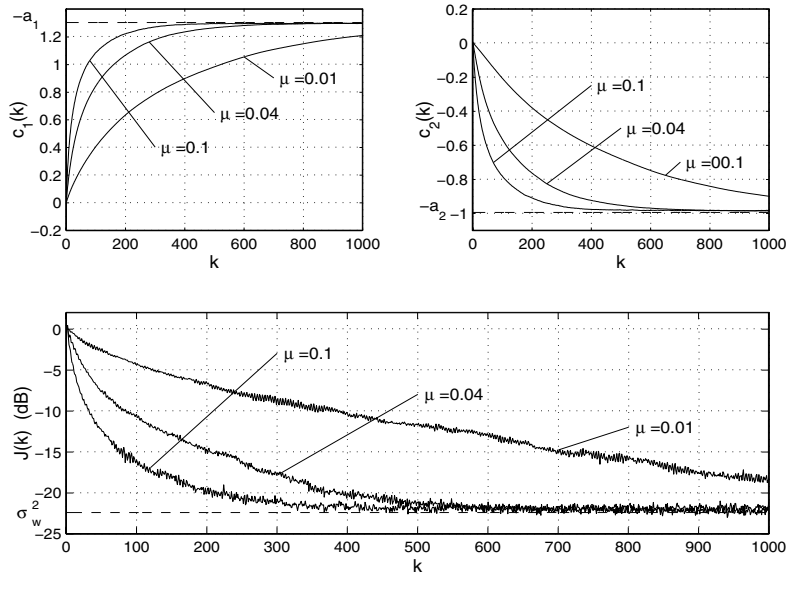


Figure 3.14. Comparison among curves of convergence of the mean obtained by the standard LMS algorithm for three values of μ.

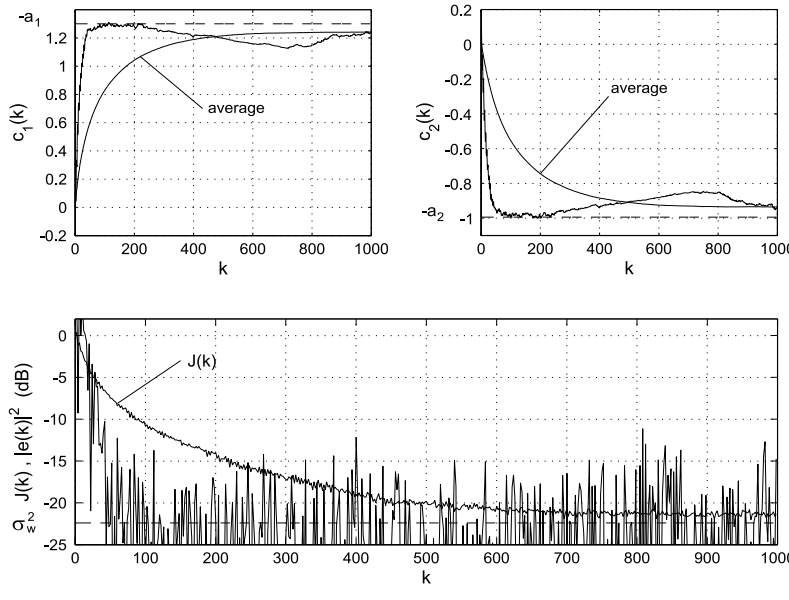


Figure 3.15. Convergence curves for the predictor of order $N = 2$, obtained by the leaky LMS.

for $\mu = 0.04$ and $\alpha = 0.01$. We note that the steady-state values are worse than in the previous case.

Example 3.1.3 (Normalized LMS)
The equation for updating the coefficient vector is

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \mu(k)e(k)\mathbf{x}(k-1) \tag{3.155}$$

The adaptation gain μ is of the type

$$\mu(k) = \frac{\tilde{\mu}}{p + N\hat{\sigma}_x^2(k)} \tag{3.156}$$

where

$$\hat{\sigma}_x^2(k) = a\hat{\sigma}_x^2(k-1) + (1-a)|x(k)|^2 \qquad k \geq 0 \tag{3.157}$$

$$\hat{\sigma}_x^2(-1) = \tfrac{1}{2}[|x(-1)|^2 + |x(-2)|^2] \tag{3.158}$$

with

$$a = 1 - 2^{-5} = 0.97 \tag{3.159}$$

and

$$p = \frac{1}{10}E[\|\mathbf{x}\|^2] = 0.2 \tag{3.160}$$

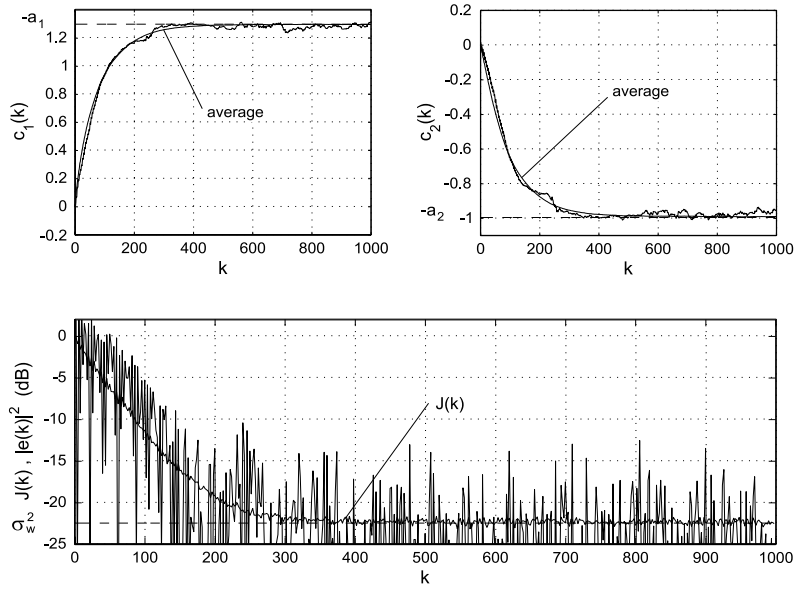


Figure 3.16. Convergence curves for the predictor of order $N = 2$, obtained by the normalized LMS algorithm.

Convergence curves are plotted in Figure 3.16 for a single realization and for the mean (estimated over 500 realizations) of the coefficients and of the squared prediction error, for $\tilde{\mu} = 0.08$. We note that, with respect to the standard LMS algorithm, the convergence is considerably faster.

A direct comparison of the convergence curves obtained in the previous examples is given in Figure 3.17.

Example 3.1.4 (Sign LMS algorithm)
We consider the three versions of the sign LMS algorithm:

(1) $\mathbf{c}(k+1) = \mathbf{c}(k) + \mu\,\mathrm{sgn}(e(k))\mathbf{x}(k-1)$,

(2) $\mathbf{c}(k+1) = \mathbf{c}(k) + \mu e(k)\,\mathrm{sgn}(\mathbf{x}(k-1))$,

(3) $\mathbf{c}(k+1) = \mathbf{c}(k) + \mu\,\mathrm{sgn}(e(k))\,\mathrm{sgn}(\mathbf{x}(k-1))$.

A comparison of convergence curves is given in Figure 3.18 for the three versions of the sign LMS algorithm, for $\mu = 0.04$.

It turns out that version (2), where the estimation error in the adaptation equation is not quantized, yields the best performance in steady state. Version (3), however, yields fastest convergence. To decrease the prediction error in steady state for versions (1) and (3), the value of μ could be further lowered, at the expense of reducing the speed of convergence.

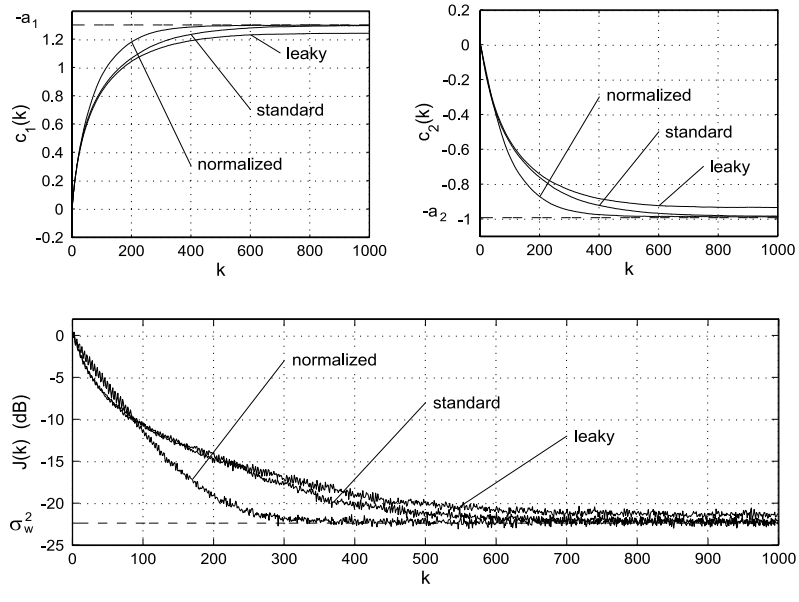


Figure 3.17. Comparison of convergence curves for the predictor of order $N = 2$, obtained by three versions of the LMS algorithm.

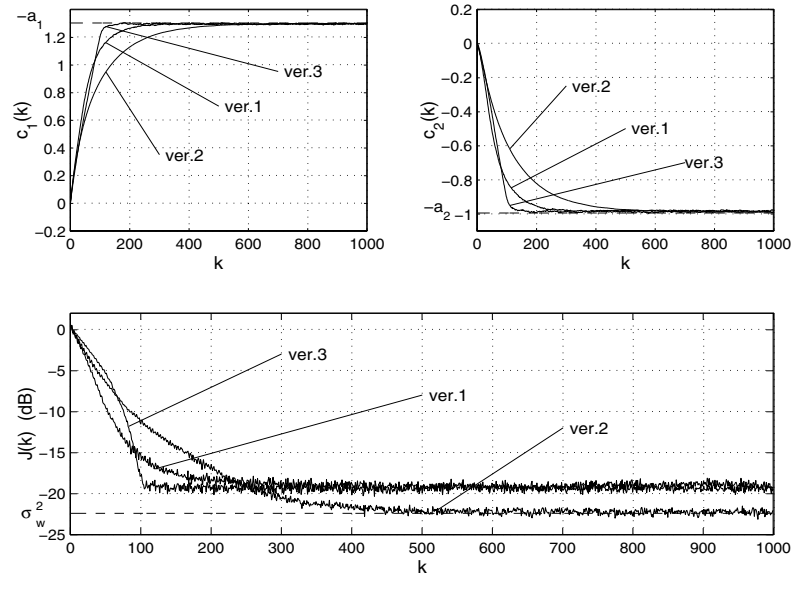


Figure 3.18. Comparison of convergence curves obtained by three versions of the sign LMS algorithm.

Observation 3.2
As observed on page 97, for an AR process x, if the order of the predictor is greater than the required minimum, the correlation matrix result is ill-conditioned with a large eigenvalue spread. Thus the convergence of the LMS prediction algorithm can be extremely slow and can lead to a solution quite different from the Yule–Walker solution. In this case it is necessary to adopt a method that ensures the stability of the error prediction filter, such as the leaky LMS.

3.2 The recursive least squares (RLS) algorithm

We now consider a recursive algorithm to estimate the vector of coefficients $\mathbf{c}$ by an LS method, named *recursive least squares* (RLS) algorithm. The RLS algorithm is characterized by a speed of convergence that can be one order of magnitude faster than the LMS algorithm, obtained at the expense of a larger computational complexity.

With reference to the system illustrated in Figure 3.19, we introduce the following quantities:

1. Input vector at instant i

$$\mathbf{x}^T(i) = [x(i), x(i-1), \ldots, x(i-N+1)] \tag{3.161}$$

2. Coefficient vector at instant k

$$\mathbf{c}^T(k) = [c_0(k), c_1(k), \ldots, c_{N-1}(k)] \tag{3.162}$$

3. Filter output signal at instant i, obtained for the vector of coefficients $\mathbf{c}(k)$

$$y(i) = \mathbf{c}^T(k)\mathbf{x}(i) = \mathbf{x}^T(i)\mathbf{c}(k) \tag{3.163}$$

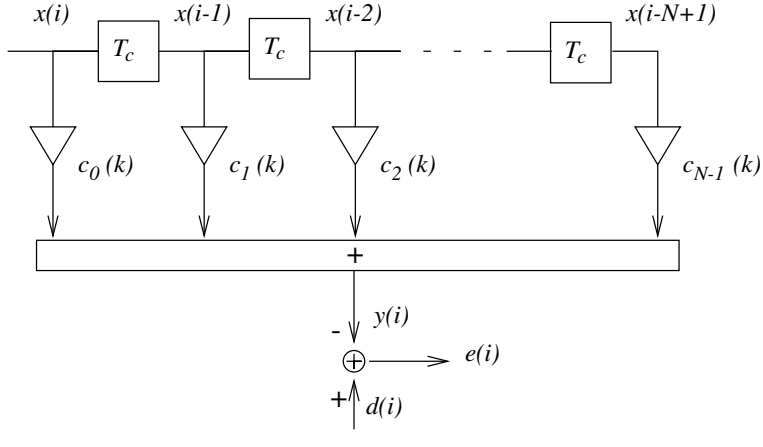


Figure 3.19. Reference system for a RLS adaptive algorithm.

4. Desired output at instant i

$$d(i) \tag{3.164}$$

At instant k, based on the observation of the sequences

$$\{x(i)\} \quad \{d(i)\} \qquad i = 1, 2, \ldots, k \tag{3.165}$$

the criterion for the optimization of the vector of coefficients $\mathbf{c}(k)$ is the *minimum sum of squared errors up to instant k*. Defining

$$\mathcal{E}(k) = \sum_{i=1}^{k} \lambda^{k-i} |e(i)|^2 \tag{3.166}$$

we want to find

$$\min_{\mathbf{c}(k)} \mathcal{E}(k) \tag{3.167}$$

where the error signal is $e(i) = d(i) - \mathbf{x}^T(i)\mathbf{c}(k)$.

Two observations arise:

- λ is a forgetting factor, that enables proper filtering operations even with non-stationary signals or slowly time-varying systems. The memory of the algorithm is approximately $1/(1-\lambda)$.
- This problem is the classical LS problem (2.128), applied to a sequence of pre-windowed samples with the exponential weighting factor λ^k.

Normal equation

Using the gradient method, the optimum value of $\mathbf{c}(k)$ satisfies the *normal equation*

$$\mathbf{\Phi}(k)\mathbf{c}(k) = \boldsymbol{\vartheta}(k) \tag{3.168}$$

where

$$\mathbf{\Phi}(k) = \sum_{i=1}^{k} \lambda^{k-i} \mathbf{x}^*(i)\mathbf{x}^T(i) \tag{3.169}$$

$$\boldsymbol{\vartheta}(k) = \sum_{i=1}^{k} \lambda^{k-i} d(i)\mathbf{x}^*(i) \tag{3.170}$$

From (3.168), if $\mathbf{\Phi}^{-1}(k)$ exists, the solution is given by

$$\mathbf{c}(k) = \mathbf{\Phi}^{-1}(k)\boldsymbol{\vartheta}(k) \tag{3.171}$$

Derivation of the RLS algorithm

To solve the normal equation by the inversion of $\mathbf{\Phi}(k)$ may be too hard, especially if N is large. Therefore we seek a recursive algorithm for $k = 1, 2, \ldots$.

Both expressions of $\mathbf{\Phi}(k)$ and $\boldsymbol{\vartheta}(k)$ can be written recursively.
From

$$\mathbf{\Phi}(k) = \sum_{i=1}^{k-1} \lambda^{k-i} \mathbf{x}^*(i)\mathbf{x}^T(i) + \mathbf{x}^*(k)\mathbf{x}^T(k) \tag{3.172}$$

it follows that

$$\mathbf{\Phi}(k) = \lambda \mathbf{\Phi}(k-1) + \mathbf{x}^*(k)\mathbf{x}^T(k) \tag{3.173}$$

and similarly

$$\boldsymbol{\vartheta}(k) = \lambda \boldsymbol{\vartheta}(k-1) + d(k)\mathbf{x}^*(k) \tag{3.174}$$

We now recall the following identity known as *matrix inversion lemma* [12]. Let

$$\mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^H \tag{3.175}$$

where $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{D}$ are positive definite matrices. Then

$$\mathbf{A}^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}(\mathbf{D} + \mathbf{C}^H\mathbf{B}\mathbf{C})^{-1}\mathbf{C}^H\mathbf{B} \tag{3.176}$$

For

$$\mathbf{A} = \mathbf{\Phi}(k) \qquad \mathbf{B}^{-1} = \lambda\mathbf{\Phi}(k-1) \qquad \mathbf{C} = \mathbf{x}^*(k) \qquad \mathbf{D} = 1 \tag{3.177}$$

the equation (3.176) becomes

$$\mathbf{\Phi}^{-1}(k) = \lambda^{-1}\mathbf{\Phi}^{-1}(k-1) - \frac{\lambda^{-1}\mathbf{\Phi}^{-1}(k-1)\mathbf{x}^*(k)\mathbf{x}^T(k)\lambda^{-1}\mathbf{\Phi}^{-1}(k-1)}{1 + \mathbf{x}^T(k)\lambda^{-1}\mathbf{\Phi}^{-1}(k-1)\mathbf{x}^*(k)} \tag{3.178}$$

We introduce two quantities:

$$\mathbf{P}(k) = \mathbf{\Phi}^{-1}(k) \tag{3.179}$$

and

$$\mathbf{k}^*(k) = \frac{\lambda^{-1}\mathbf{\Phi}^{-1}(k-1)\mathbf{x}^*(k)}{1 + \lambda^{-1}\mathbf{x}^T(k)\mathbf{\Phi}^{-1}(k-1)\mathbf{x}^*(k)} \tag{3.180}$$

also called the *Kalman vector gain*. From (3.178) we have the recursive relation

$$\mathbf{P}(k) = \lambda^{-1}\mathbf{P}(k-1) - \lambda^{-1}\mathbf{k}^*(k)\mathbf{x}^T(k)\mathbf{P}(k-1) \tag{3.181}$$

We derive now a simpler expression for $\mathbf{k}^*(k)$. From (3.180) we obtain

$$\mathbf{k}^*(k)[1 + \lambda^{-1}\mathbf{x}^T(k)\mathbf{\Phi}^{-1}(k-1)\mathbf{x}^*(k)] = \lambda^{-1}\mathbf{\Phi}^{-1}(k-1)\mathbf{x}^*(k) \tag{3.182}$$

from which we get

$$\begin{aligned}\mathbf{k}^*(k) &= \lambda^{-1}\mathbf{P}(k-1)\mathbf{x}^*(k) - \lambda^{-1}\mathbf{k}^*(k)\mathbf{x}^T(k)\mathbf{P}(k-1)\mathbf{x}^*(k) \\ &= [\lambda^{-1}\mathbf{P}(k-1) - \lambda^{-1}\mathbf{k}^*(k)\mathbf{x}^T(k)\mathbf{P}(k-1)]\mathbf{x}^*(k)\end{aligned} \tag{3.183}$$

Using (3.181), it follows

$$\mathbf{k}^*(k) = \mathbf{P}(k)\mathbf{x}^*(k) \tag{3.184}$$

Using the (3.174), the recursive equation to update the estimate of $\mathbf{c}$ is given by

$$\begin{aligned}\mathbf{c}(k) &= \mathbf{\Phi}^{-1}(k)\boldsymbol{\vartheta}(k) \\ &= \mathbf{P}(k)\boldsymbol{\vartheta}(k) \\ &= \lambda\mathbf{P}(k)\boldsymbol{\vartheta}(k-1) + \mathbf{P}(k)\mathbf{x}^*(k)d(k)\end{aligned} \tag{3.185}$$

Substituting the recursive expression for $\mathbf{P}(k)$ in the first term, we get

$$\begin{aligned}\mathbf{c}(k) &= \lambda[\lambda^{-1}\mathbf{P}(k-1) - \lambda^{-1}\mathbf{k}^*(k)\mathbf{x}^T(k)\mathbf{P}(k-1)]\boldsymbol{\vartheta}(k-1) + \mathbf{P}(k)\mathbf{x}^*(k)d(k) \\ &= \mathbf{P}(k-1)\boldsymbol{\vartheta}(k-1) - \mathbf{k}^*(k)\mathbf{x}^T(k)\mathbf{P}(k-1)\boldsymbol{\vartheta}(k-1) + \mathbf{P}(k)\mathbf{x}^*(k)d(k) \\ &= \mathbf{c}(k-1) + \mathbf{k}^*(k)[d(k) - \mathbf{x}^T(k)\mathbf{c}(k-1)]\end{aligned} \tag{3.186}$$

where in the last step (3.184) has been used. Defining the *a priori estimation error*,

$$\epsilon(k) = d(k) - \mathbf{x}^T(k)\mathbf{c}(k-1) \tag{3.187}$$

we note that $\mathbf{x}^T(k)\mathbf{c}(k-1)$ is the filter output at instant k obtained by using the old coefficient estimate. In other words, from the *a posteriori estimation error*

$$e(k) = d(k) - \mathbf{x}^T(k)\mathbf{c}(k) \tag{3.188}$$

we could say that $\epsilon(k)$ is an approximated value of $e(k)$, that is computed before updating $\mathbf{c}$. In any case the relation holds

$$\mathbf{c}(k) = \mathbf{c}(k-1) + \mathbf{k}^*(k)\epsilon(k) \tag{3.189}$$

In summary, the RLS algorithm consists of four equations:

$$\mathbf{k}^*(k) = \frac{\mathbf{P}(k-1)\mathbf{x}^*(k)}{\lambda + \mathbf{x}^T(k)\mathbf{P}(k-1)\mathbf{x}^*(k)} \tag{3.190}$$

$$\epsilon(k) = d(k) - \mathbf{x}^T(k)\mathbf{c}(k-1) \tag{3.191}$$

$$\mathbf{c}(k) = \mathbf{c}(k-1) + \epsilon(k)\mathbf{k}^*(k) \tag{3.192}$$

$$\mathbf{P}(k) = \lambda^{-1}\mathbf{P}(k-1) - \lambda^{-1}\mathbf{k}^*(k)\mathbf{x}^T(k)\mathbf{P}(k-1) \tag{3.193}$$

In (3.190), $\mathbf{k}(k)$ is the input vector filtered by $\mathbf{P}(k-1)$ and normalized by the $\lambda + \mathbf{x}^T(k)\mathbf{P}(k-1)\mathbf{x}^*(k)$. The term $\mathbf{x}^T(k)\mathbf{P}(k-1)\mathbf{x}^*(k)$ may be interpreted as the energy of the filtered input.

Initialization of the RLS algorithm

We need to assign a value to $\mathbf{P}(0)$. We modify the definition of $\mathbf{\Phi}(k)$ in

$$\mathbf{\Phi}(k) = \sum_{i=1}^{k} \lambda^{k-i}\mathbf{x}^*(i)\mathbf{x}^T(i) + \delta\lambda^k\mathbf{I} \qquad \text{with } \delta \ll 1 \tag{3.194}$$

so that

$$\mathbf{\Phi}(0) = \delta\mathbf{I} \tag{3.195}$$

This is equivalent to having for $k \leq 0$ an all zero input with the exception of $x(-N+1) = (\lambda^{-N+1}\delta)^{1/2}$. Consequently

$$\mathbf{P}(0) = \delta^{-1}\mathbf{I} \qquad \delta \ll \mathtt{r}_x(0) \tag{3.196}$$

Typically

$$\delta^{-1} = \frac{100}{\mathtt{r}_x(0)} \tag{3.197}$$

where $\mathtt{r}_x(0)$ is the statistical power of the input signal. In Table 3.1 we give a version of the RLS algorithm that exploits the fact that $\mathbf{P}(k)$ (inverse of the Hermitian matrix $\mathbf{\Phi}(k)$) is Hermitian, hence

$$\mathbf{x}^T(k)\mathbf{P}(k-1) = [\mathbf{P}(k-1)\mathbf{x}^*(k)]^H = \boldsymbol{\pi}^T(k) \tag{3.198}$$

Table 3.1 RLS algorithm.

Initialization	
	$\mathbf{c}(0) = \mathbf{0}$
	$\mathbf{P}(0) = \delta^{-1}\mathbf{I}$
For $k = 1, 2, \ldots$	
	$\boldsymbol{\pi}^*(k) = \mathbf{P}(k-1)\mathbf{x}^*(k)$
	$r(k) = \dfrac{1}{\lambda + \mathbf{x}^T(k)\boldsymbol{\pi}^*(k)}$
	$\mathbf{k}^*(k) = r(k)\boldsymbol{\pi}^*(k)$
	$\epsilon(k) = d(k) - \mathbf{x}^T(k)\mathbf{c}(k-1)$
	$\mathbf{c}(k) = \mathbf{c}(k-1) + \epsilon(k)\mathbf{k}^*(k)$
	$\mathbf{P}(k) = \lambda^{-1}(\mathbf{P}(k-1) - \mathbf{k}^*(k)\boldsymbol{\pi}^T(k))$

Recursive form of $\mathcal{E}_{min}$

We set

$$\mathcal{E}_d(k) = \sum_{i=1}^{k} \lambda^{k-i} |d(i)|^2 = \lambda \mathcal{E}_d(k-1) + |d(k)|^2 \tag{3.199}$$

From the general LS expression (2.150),

$$\mathcal{E}_{min}(k) = \mathcal{E}_d(k) - \boldsymbol{\vartheta}^H(k)\mathbf{c}(k) \tag{3.200}$$

observing (3.174) and (3.192) we get

$$\begin{aligned}
\mathcal{E}_{min}(k) &= \lambda \mathcal{E}_d(k-1) + |d(k)|^2 \\
&\quad - [\lambda \boldsymbol{\vartheta}^H(k-1) + \mathbf{x}^T(k) d^*(k)][\mathbf{c}(k-1) + \epsilon(k)\mathbf{k}^*(k)] \\
&= \lambda \mathcal{E}_d(k-1) - \lambda \boldsymbol{\vartheta}^H(k-1)\mathbf{c}(k-1) \\
&\quad + d(k)d^*(k) - d^*(k)\mathbf{x}^T(k)\mathbf{c}(k-1) - \boldsymbol{\vartheta}^H(k)\mathbf{k}^*(k)\epsilon(k) \\
&= \lambda \mathcal{E}_{min}(k-1) + d^*(k)\epsilon(k) - \boldsymbol{\vartheta}^H(k)\mathbf{k}^*(k)\epsilon(k)
\end{aligned} \tag{3.201}$$

Using the expression (3.179), and recalling that $\boldsymbol{\Phi}(k)$ is Hermitian, from (3.184) we obtain

$$\begin{aligned}
\boldsymbol{\vartheta}^H(k)\mathbf{k}^*(k) &= \boldsymbol{\vartheta}^H(k)\boldsymbol{\Phi}^{-1}(k)\mathbf{x}^*(k) \\
&= [\boldsymbol{\Phi}^{-1}(k)\boldsymbol{\vartheta}(k)]^H \mathbf{x}^*(k)
\end{aligned} \tag{3.202}$$

Moreover from (3.184) and (3.171) it follows that

$$\boldsymbol{\vartheta}^H(k)\mathbf{k}^*(k) = \mathbf{c}^H(k)\mathbf{x}^*(k) = \mathbf{x}^H(k)\mathbf{c}^*(k) \tag{3.203}$$

Then (3.201) becomes

$$\begin{aligned}
\mathcal{E}_{min}(k) &= \lambda \mathcal{E}_{min}(k-1) + d^*(k)\epsilon(k) - \mathbf{c}^H(k)\mathbf{x}^*(k)\epsilon(k) \\
&= \lambda \mathcal{E}_{min}(k-1) + \epsilon(k)[d^*(k) - (\mathbf{x}^T(k)\mathbf{c}(k))^*]
\end{aligned} \tag{3.204}$$

Finally, the recursive relation is given by

$$\mathcal{E}_{min}(k) = \lambda \mathcal{E}_{min}(k-1) + \epsilon(k)e^*(k) \tag{3.205}$$

We note that, as $\mathcal{E}_{min}(k)$ is real, we get

$$\epsilon(k)e^*(k) = \epsilon^*(k)e(k) \tag{3.206}$$

that is $\epsilon(k)e^*(k)$ is a real scalar value.

Convergence of the RLS algorithm

We make some remarks on the convergence of the RLS algorithm.

- The RLS algorithm converges in the mean-square sense in about $2N$ iterations, independently of the eigenvalue spread of $\mathbf{R}$.
- For $k \to \infty$ there is no excess error and the misadjustment MSD is zero. This is true for $\lambda = 1$.
- In any case, when $\lambda < 1$ the "memory" of the algorithm is approximately $1/(1-\lambda)$ and

$$\text{MSD} = \frac{1-\lambda}{1+\lambda} N \tag{3.207}$$

- From the above observation it follows that the RLS algorithm for $\lambda < 1$ gives origin to noisy estimates.
- On the other hand the RLS algorithm for $\lambda < 1$ can be used for tracking slowly time-varying systems.

Computational complexity of the RLS algorithm

Exploiting the symmetry of $\mathbf{P}(k)$, the computational complexity of the RLS algorithm, expressed as the number of complex multiplications per output sample, is given by

$$\text{CC}_{\text{RLS}} = 2N^2 + 4N \tag{3.208}$$

For a number of $(K - N + 1)$ output samples, the direct method (3.171) requires instead

$$\text{CC}_{\text{DIR}} = N^2 + N + \frac{N^3}{K - N + 1} \tag{3.209}$$

We note that, if $K \gg N$, the direct method is more convenient. In any case the RLS solution has other advantages:

1. It can be numerically more stable than the direct method.
2. It provides an estimate of the coefficients at each step and not only at the end of the data sequence.
3. For $\lambda < 1$ and $1/(1-\lambda)$ much less than the time interval it takes for the input samples to change statistics, the algorithm is capable of "tracking" the changes.

Example of application: the predictor

With reference to the AR(2) process considered in Section 3.1.5, convergence curves for the RLS algorithm are plotted in Figure 3.20 for a single realization and for the mean (estimated over 500 realizations) of the coefficients and of the squared estimation error, for $\lambda = 1$. We note that a different scale is used for the abscissa as compared to the LMS method; in fact the RLS algorithm converges in a number of iterations of the order of N.

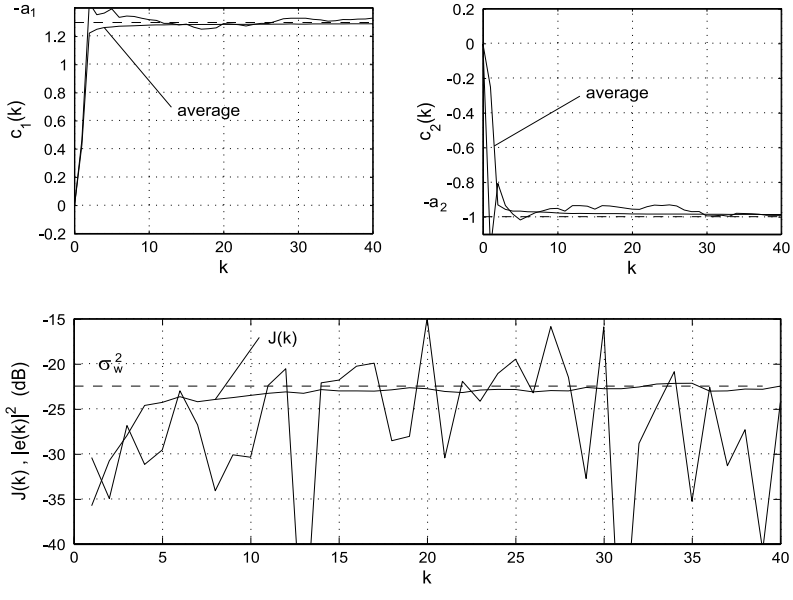


Figure 3.20. Convergence curves for the predictor of order $N = 2$, obtained by the RLS algorithm.

3.3 Fast recursive algorithms

As observed in the previous section, the RLS algorithm has the disadvantage of requiring $(2N^2 + 4N)$ multiplications per iteration. Therefore we will list a few *fast* algorithms, whose computational complexity increases linearly with N, the number of dimensions of the coefficient vector $\mathbf{c}$.

1. *Algorithms for transversal filters.* The *fast Kalman algorithm* has the same speed of convergence as the RLS, but with a computational complexity comparable to that of the LMS algorithm. Exploiting some properties of the correlation matrix $\mathbf{\Phi}(k)$, Falconer and Ljung [14] have shown that the recursive equation (3.193) requires only $10(2N + 1)$ multiplications. Cioffi and Kailath [15], with their *fast transversal filter* (FTF), have further reduced the number of multiplications to $7(2N + 1)$.

   The implementation of these algorithms still remains relatively simple; their weak point resides in the sensitivity of the operations to round off errors in the various coefficients and signals. As a consequence the fast algorithms may become numerically unstable.

2. *Algorithms for lattice filters.* There are versions of the RLS algorithm for *lattice* structures that in the literature are called *recursive least squares lattice* (LSL) that have, in addition to a lower computational complexity than the standard RLS form, strong and weak points similar to those already discussed in the case of the LMS algorithm for lattice structures [12, 16].

Table 3.2 Comparison of three adaptive algorithms in terms of computational complexity.

cost function	algorithm	multiplications	divisions	additions subtractions
MSE	LMS	$2N+1$	0	$2N$
LS	RLS	$2N^2+7N+5$	N^2+4N+3	$2N^2+6N+4$
	FTF	$7(2N+1)$	4	$6(2N+1)$

3. *Algorithms for filters based on systolic structures.* A particular structure is the *QR decomposition-based LSL*. The name comes from the use of an orthogonal triangularization process, usually known as QR decomposition, that leads to a systolic-type structure with the following characteristics:

   - high speed of convergence;
   - numerical stability, owing to the QR decomposition and lattice structure;
   - a very efficient and modular structure, which does not require the *a priori* knowledge of the filter order and is suitable for implementation in *very large-scale integration* (VLSI) technology.

   For further study on the subject we refer the reader to [17, 18, 19, 20, 21, 22].

3.3.1 Comparison of the various algorithms

In practice the choice of an algorithm must be made bearing in mind some fundamental aspects:

- computational complexity;
- performance in terms of speed of convergence, error in steady state, and tracking capabilities under non-stationary conditions;
- robustness, that is good performance achieved in the presence of a large eigenvalue spread and finite-precision arithmetic [5, 23].

Regarding the computational complexity per output sample, a brief comparison among LMS, RLS and FTF is given in Table 3.2. Although the FTF method exhibits a lower computational complexity than the RLS method, its implementation is rather laborious, therefore it is rarely used.

3.4 Block adaptive algorithms in the frequency domain

In this section some algorithms are examined that transform the input signal, for example from the time to the frequency domain, before adaptive filtering. With respect to the LMS algorithm, this approach may exhibit: a) lower computational complexity, or b) improved

convergence properties of the adaptive process. We will first consider some adaptive algorithms in the frequency domain that offer some advantages from the standpoint of computational complexity [24, 25, 26, 27].

3.4.1 Block LMS algorithm in the frequency domain: the basic scheme

The basic scheme includes a filter that performs the equivalent operation of a *circular convolution* in the frequency domain. As illustrated in Figure 3.21, the method operates over blocks of N samples. The instant at which a block is processed is $k = nN$, where n is an integer number. Each input block is transformed using the DFT (see Section 1.4). The samples of the transformed sequence are denoted by $\{X_i(nN)\}$, $i = 0, 1, \ldots, N-1$. We indicate with $\{D_i(nN)\}$ and $\{Y_i(nN)\}$, $i = 0, 1, \ldots, N-1$, respectively, the DFT of the desired output and of the adaptive filter output. Defining $E_i(nN) = D_i(nN) - Y_i(nN)$, the LMS adaptation algorithm is expressed as:

$$C_i((n+1)N) = C_i(nN) + \mu E_i(nN) X_i^*(nN) \qquad i = 0, 1, \ldots, N-1 \tag{3.210}$$

In the following, lower case letters will be used to indicate sequences in the time domain, while upper case letters will denote sequences in the frequency domain.

Computational complexity of the block LMS algorithm via FFT

We consider the computational complexity of the scheme of Figure 3.21 for *N-sample real input vectors*. The algorithm requires three N-point FFTs and $2N$ complex multiplications to update $\{C_i\}$ and compute $\{Y_i\}$. As for real data the complexity of an N-point FFT in

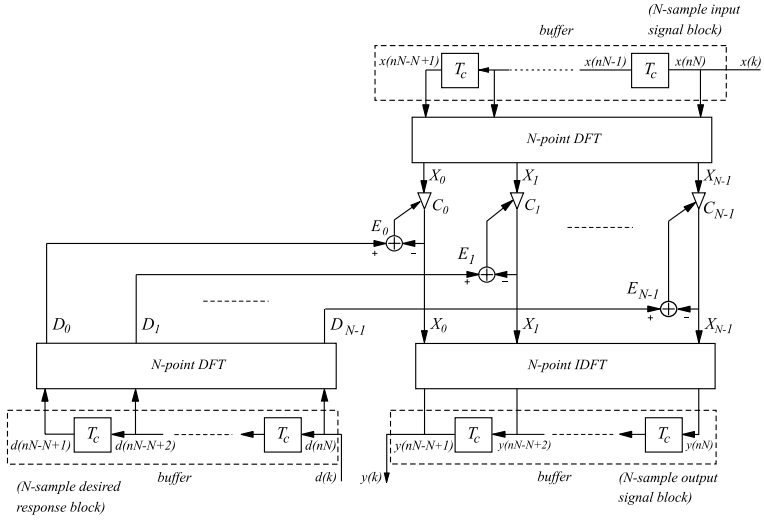


Figure 3.21. Adaptive transversal filter in the frequency domain.

Table 3.3 Comparison between the computational complexity of the LMS algorithm via FFT and the standard LMS for various values of the filter length N.

N	CC_{LMS_f}/CC_{LMS_t}
16	0.41
64	0.15
1024	0.015

terms of complex multiplications is given by

$$\begin{aligned} N\text{-point FFT of } N \text{ real samples} &= \frac{N}{2}\text{-point FFT} + \frac{N}{2} \\ &= \left(\frac{N}{4}\log_2\frac{N}{2} - \frac{N}{2}\right) + \frac{N}{2} \end{aligned} \tag{3.211}$$

then the algorithm requires a number of complex multiplications per output sample equal to

$$CC_{LMS_f} = 3\left(\frac{1}{4}\log_2\frac{N}{2}\right) + 1 \tag{3.212}$$

using the fact that $\{Y_i\}$ and $\{C_i\}$, $i = 0, 1, \ldots, N-1$ are Hermitian sequences. As each complex multiplication requires four real multiplications, the complexity in terms of real multiplications per output sample becomes

$$CC_{LMS_f} = 4\left(\frac{3}{4}\log_2\frac{N}{2} + 1\right) \tag{3.213}$$

We note that the complexity in terms of real multiplications per output sample of the standard LMS algorithm is

$$CC_{LMS_t} = 2N + 1 \simeq 2N \tag{3.214}$$

A comparison between the computational complexity of the LMS algorithm via FFT and the standard LMS algorithm is given in Table 3.3. We note that the advantage of the LMS algorithm via FFT is non negligible even for small values of N.

However, as the product between DFTs of two time sequences is equivalent to a circular convolution, the direct application of the scheme of Figure 3.21 is appropriate only if the relation between y and x is a circular convolution rather than a linear convolution.

3.4.2 Block LMS algorithm in the frequency domain: the FLMS algorithm

We consider a block LMS adaptive algorithm in the time domain, for blocks of N input samples. Let us define:

1. input vector at instant k

$$\mathbf{x}^T(k) = [x(k), x(k-1), \ldots, x(k-N+1)] \tag{3.215}$$

2. coefficient vector at instant nN

$$\mathbf{c}^T(nN) = [c_0(nN), c_1(nN), \ldots, c_{N-1}(nN)] \tag{3.216}$$

3. filter output signal at instant $nN + i$

$$y(nN+i) = \mathbf{c}^T(nN)\mathbf{x}(nN+i) \tag{3.217}$$

4. error at instant $nN + i$

$$e(nN+i) = d(nN+i) - y(nN+i) \qquad i = 0, 1, \ldots N-1 \tag{3.218}$$

The equation for updating the coefficients according to the block LMS algorithm is given by

$$\mathbf{c}((n+1)N) = \mathbf{c}(nN) + \mu \sum_{i=0}^{N-1} e(nN+i)\mathbf{x}^*(nN+i) \tag{3.219}$$

As in the case of the standard LMS algorithm, the updating term is the estimate of the gradient at instant nN, $\nabla(nN)$. The above equations can be efficiently implemented in the frequency domain by the *overlap-save* technique (see (1.112)). Assuming L-point blocks, where for example $L = 2N$, we define[10]

$$\mathbf{C'}^T(nN) = \mathrm{DFT}[\mathbf{c}^T(nN), \underbrace{0, \ldots, 0}_{N \text{ zeros}}] \tag{3.220}$$

$$\mathbf{X}'(nN) = \mathrm{diag}\left\{\mathrm{DFT}[\underbrace{x(nN-N), \ldots, x(nN-1)}_{\text{block } n-1}, \underbrace{x(nN), \ldots, x(nN+N-1)}_{\text{block } n}]\right\} \tag{3.221}$$

and

$$\mathbf{Y}'(nN) = \mathbf{X}'(nN)\mathbf{C}'(nN) \tag{3.222}$$

then the filter output at instants $k = nN, nN+1, \ldots, nN+N-1$, is given by

$$\mathbf{y}(nN) = \begin{bmatrix} y(nN) \\ y(nN+1) \\ \vdots \\ y(nN+N-1) \end{bmatrix} = \text{last } N \text{ elements of } \mathrm{DFT}^{-1}[\mathbf{Y}'(nN)] \tag{3.223}$$

[10] The superscript $'$ denotes a vector of $2N$ elements.

We give now the equations to update the coefficients in the frequency domain. Let us consider the m-th component of the gradient,

$$[\nabla(nN)]_m = \sum_{i=0}^{N-1} e(nN+i)x^*(nN+i-m) \qquad m = 0, 1, \ldots, N-1 \tag{3.224}$$

This component is given by the correlation between the error sequence $\{e(k)\}$ and input $\{x(k)\}$, which is also equal to the convolution between $e(k)$ and $x^*(-k)$. Let

$$\mathbf{E}'^T(nN) = \text{DFT}[\underbrace{0, \ldots, 0}_{N \text{ zeros}}, \underbrace{d(nN) - y(nN), \ldots, d(nN+N-1) - y(nN+N-1)}_{\text{errors in block } n}] \tag{3.225}$$

then

$$\nabla(nN) = \text{first } N \text{ elements of } \text{DFT}^{-1}[\mathbf{X}'^*(nN)\mathbf{E}'(nN)] \tag{3.226}$$

In the frequency domain, the adaptation equation (3.219) becomes

$$\mathbf{C}'((n+1)N) = \mathbf{C}'(nN) + \mu\text{DFT}\begin{bmatrix} \nabla(nN) \\ \mathbf{0} \end{bmatrix} \tag{3.227}$$

where $\mathbf{0}$ is the null vector with N elements.

In summary, if $\mathbf{0}_{N\times N}$ is the $N \times N$ all zero matrix, $\mathbf{I}_{N\times N}$ the $N \times N$ identity matrix, and $\mathbf{F}$ the $2N \times 2N$ DFT matrix, then the following equations define the fast LMS (FLMS):

$$\mathbf{d}'^T(nN) = [\mathbf{0}^T, d(nN), \ldots, d(nN+N-1)] \tag{3.228}$$

$$\mathbf{y}'(nN) = \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{0}_{N\times N} \\ \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \end{bmatrix} \mathbf{F}^{-1}[\mathbf{X}'(nN)\mathbf{C}'(nN)] \tag{3.229}$$

$$\mathbf{E}'(nN) = \mathbf{F}[\mathbf{d}'(nN) - \mathbf{y}'(nN)] \tag{3.230}$$

$$\mathbf{C}'((n+1)N) = \mathbf{C}'(nN) + \mu\mathbf{F}\begin{bmatrix} \mathbf{I}_{N\times N} & \mathbf{0}_{N\times N} \\ \mathbf{0}_{N\times N} & \mathbf{0}_{N\times N} \end{bmatrix} \mathbf{F}^{-1}[\mathbf{X}'^*(nN)\mathbf{E}'(nN)] \tag{3.231}$$

The implementation of the FLMS algorithm is illustrated in Figure 3.22.

Computational complexity of the FLMS algorithm

For N output samples we have to evaluate five $2N$-point FFTs and $4N$ complex multiplications. For real input samples, referring to the scheme in Figure 3.22, the complexity in terms of real multiplications per output sample is given by

$$\text{CC}_{\text{FLMS}} = 10\log_2 N + 8 \tag{3.232}$$

A comparison between the computational complexity of the FLMS algorithm and the standard LMS is given in Table 3.4.

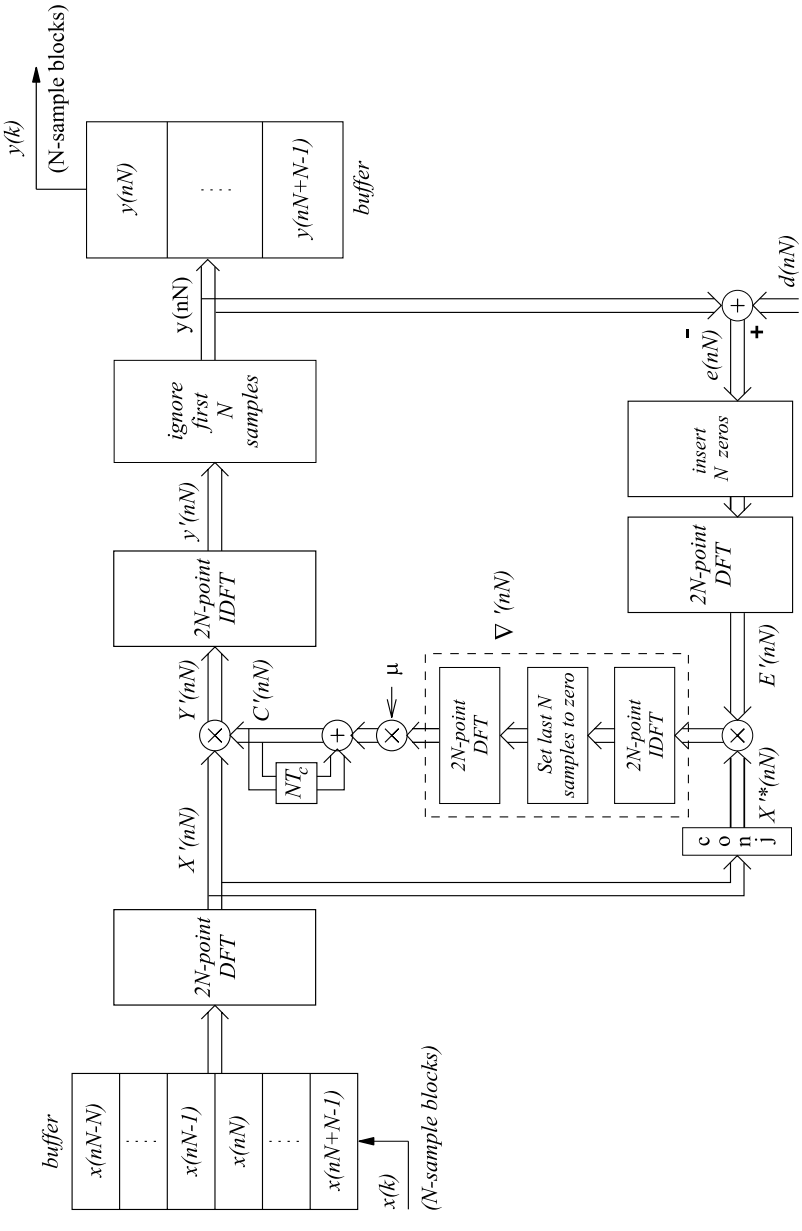


Figure 3.22. Implementation of the FLMS algorithm.

Table 3.4 Computational complexity comparison between FLMS and LMS.

N	CC_{FLMS}/CC_{LMS}
16	1.5
32	0.85
64	0.53
1024	0.05

Convergence in the mean of the coefficients for the FLMS algorithm

Observing (3.217) and (3.218), and taking the expectation of both members of the adaptation equation (3.219), we get

$$\begin{aligned} E[\mathbf{c}((n+1)N)] &= E[\mathbf{c}(nN)] + \mu N(\mathbf{p} - \mathbf{R}E[\mathbf{c}(nN)]) \\ &= (\mathbf{I} - \mu N\mathbf{R})E[\mathbf{c}(nN)] + \mu N\mathbf{p} \end{aligned} \tag{3.233}$$

where, as usual, $\mathbf{R} = E[\mathbf{x}^*(k)\mathbf{x}^T(k)]$ and $\mathbf{p} = \mathbf{r}_{d\mathbf{x}} = E[d(k)\mathbf{x}^*(k)]$.

Recalling the analysis of the convergence of the *steepest-descent* algorithm of Section 3.1.1, we have

$$\lim_{n\to\infty} E[\mathbf{c}((n+1)N)] = \mathbf{R}^{-1}\mathbf{p} \tag{3.234}$$

for $0 < \mu < 2/(N\lambda_{max})$, where λ_{max} is the maximum eigenvalue of $\mathbf{R}$.

From these equations we can conclude:

1. The FLMS algorithm converges in the mean to the same solution of the LMS, however, μ must be smaller by a factor N in order to guarantee stability.
2. The time constant for the convergence of the i-th mode (for $\mu \ll 1$) is

$$\begin{aligned} \tau_i &= \frac{1}{\mu\lambda_i N} \qquad \text{blocks} \\ &= \frac{1}{\mu\lambda_i} \qquad \text{samples} \end{aligned} \tag{3.235}$$

   equal to that of the LMS algorithm.
3. For $\mu \ll 2/N\lambda_{max}$, it can be seen that the misadjustment is equal to that of the LMS algorithm:

$$\text{MSD} = \frac{\mu}{2}\,\text{tr}[\mathbf{R}] = \frac{\mu}{2}N r_x(0) \tag{3.236}$$

3.5 LMS algorithm in a transformed domain

We consider now some adaptive algorithms in the frequency domain that offer some advantages in terms of speed of convergence [28].

3.5.1 Basic scheme

Referring to Figure 3.23, we define the following quantities.

1. *Input vector at instant k*

$$\mathbf{x}^T(k) = [x(k), x(k-1), \ldots, x(k-N+1)] \tag{3.237}$$

with correlation matrix $\mathbf{R}_\mathbf{x} = E[\mathbf{x}^*(k)\mathbf{x}^T(k)]$.

2. *Transformed vector*

$$\mathbf{z}^T(k) = [z_0(k), z_1(k), \ldots, z_{N-1}(k)] \tag{3.238}$$

In general,

$$\mathbf{z}(k) = \mathbf{G}\mathbf{x}(k) \tag{3.239}$$

where $\mathbf{G}$ is a unitary matrix of rank N:

$$\mathbf{G}^{-1} = \mathbf{G}^H \tag{3.240}$$

3. *Coefficient vector at instant k*

$$\mathbf{c}^T(k) = [c_0(k), c_1(k), \ldots, c_{N-1}(k)] \tag{3.241}$$

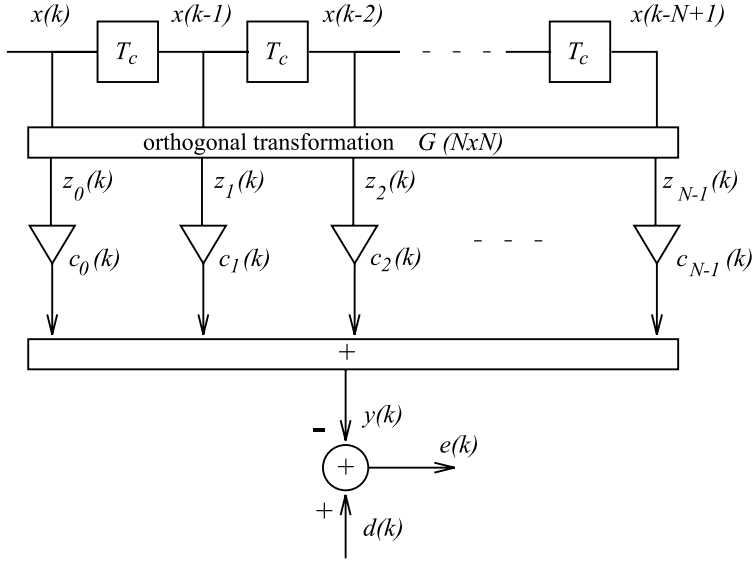


Figure 3.23. General scheme for a LMS algorithm in a transformed domain.

4. *Filter output signal*

$$y(k) = \mathbf{z}^T(k)\mathbf{c}(k) = \mathbf{c}^T(k)\mathbf{z}(k) \tag{3.242}$$

5. *Estimation error*

$$e(k) = d(k) - y(k) \tag{3.243}$$

6. *Equation for updating the coefficients*, LMS type:

$$c_i(k+1) = c_i(k) + \mu_i e(k) z_i^*(k) \qquad i = 0, 1, \ldots, N-1 \tag{3.244}$$

where

$$\mu_i = \frac{\tilde{\mu}}{E[|z_i(k)|^2]} \tag{3.245}$$

We note that each component of the adaptation gain vector has been normalized using the statistical power of the corresponding component of the transformed input vector. The various powers can be estimated, e.g., by considering a small window of input samples or recursively. Let

$$\mathbf{\Lambda}_N = \text{diag}\{E[|z_0(k)|^2], E[|z_1(k)|^2], \ldots, E[|z_{N-1}(k)|^2]\} \tag{3.246}$$

Then (3.244) can be written in vector notation as

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \tilde{\mu} e(k) \mathbf{\Lambda}_N^{-1} \mathbf{z}^*(k) \tag{3.247}$$

We find that, for a suitable choice of $\tilde{\mu}$,

$$\lim_{k \to \infty} \mathbf{c}(k+1) = \mathbf{c}_{opt} = \mathbf{R}_{\mathbf{z}}^{-1} \mathtt{r}_{d\mathbf{z}} \tag{3.248}$$

where

$$\mathbf{R}_{\mathbf{z}} = E[\mathbf{z}^*(k)\mathbf{z}^T(k)] = E[\mathbf{G}^*\mathbf{x}^*(k)\mathbf{x}^T(k)\mathbf{G}^T] = \mathbf{G}^*\mathbf{R}_{\mathbf{x}}\mathbf{G}^T \tag{3.249}$$

and

$$\mathtt{r}_{d\mathbf{z}} = E[d(k)\mathbf{z}^*(k)] = \mathbf{G}^* E[d(k)\mathbf{x}^*(k)] = \mathbf{G}^* \mathtt{r}_{d\mathbf{x}} \tag{3.250}$$

Then

$$\begin{aligned} \mathbf{c}_{opt} &= (\mathbf{G}^*\mathbf{R}_{\mathbf{x}}\mathbf{G})^{-1}\mathbf{G}^*\mathtt{r}_{d\mathbf{x}} \\ &= \mathbf{G}^{-1}\mathbf{R}_{\mathbf{x}}^{-1}\mathbf{G}^{*-1}\mathbf{G}^*\mathtt{r}_{d\mathbf{x}} \\ &= \mathbf{G}^H\mathbf{R}_{\mathbf{x}}^{-1}\mathtt{r}_{d\mathbf{x}} \\ &= \mathbf{G}^H(\mathbf{R}_{\mathbf{x}}^{-1}\mathtt{r}_{d\mathbf{x}}) \end{aligned} \tag{3.251}$$

where $\mathbf{R}_{\mathbf{x}}^{-1}\mathtt{r}_{d\mathbf{x}}$ is the optimum Wiener solution without transformation.

On the speed of convergence

The speed of convergence depends on the eigenvalue spread of the matrix $\mathbf{R_z}$. If $\mathbf{R_z}$ is diagonal, then the eigenvalue spread of $\mathbf{\Lambda}_N^{-1}\mathbf{R_z}$ is equal to one. Consequently, a transformation with these characteristics exhibits the best convergence properties. In this case the adaptation algorithm reduces to N independent scalar adaptation algorithms in the transformed domain, and the N modes of convergence do not influence each other. Common choices for $\mathbf{G}$ are the following:

1. Karhunen-Loève transform (KLT). The KLT depends on $\mathbf{R_x}$, and consequently is difficult to evaluate in real time.
2. Lower triangular matrix transformation, used in lattice filters.
3. DFT and discrete cosine transform (DCT). They reduce the number of computations to evaluate $\mathbf{z}(k)$ in (3.239) from $O(N^2)$ to $O(N \log_2 N)$. Moreover, recalling the definition (1.376) of the eigenvalue spread, these two transformations, for the normalization $\mathbf{\Lambda}_N^{-1}$, whiten the signal x by operating on the different sub-bands; the resulting signal, with reduced spectral variations, is used for the adaptation process.

3.5.2 Normalized FLMS algorithm

The convergence of the FLMS algorithm can be improved by dividing each component of the vector $[\mathbf{X}'^*(nN)\mathbf{E}'(nN)]$ in (3.231) by the power of the respective component of $\mathbf{X}'(nN)$. Consequently the adaptation gain μ is adjusted to the various modes. This procedure, however, requires that the components of $\mathbf{X}'(nN)$ are indeed uncorrelated.

3.5.3 LMS algorithm in the frequency domain

In this case

$$\mathbf{G} = \mathbf{F} \tag{3.252}$$

$N \times N$ DFT matrix. Then

$$z_i(k) = \sum_{m=0}^{N-1} x(k-m) e^{-j2\pi \frac{mi}{N}} \qquad i = 0, 1, \ldots, N-1 \tag{3.253}$$

or, in a simpler recursive form,

$$z_i(k) = z_i(k-1) \exp\left(-j2\pi \frac{i}{N}\right) + x(k) - x(k-N) \tag{3.254}$$

The filters are of passband *comb* type, implemented by either 1) FFT with parallel input or 2) recursively with serial input to implement equations (3.254), as illustrated in Figure 3.24. In both cases the computational complexity to evaluate the output sample $y(k)$ is $O(N \log_2 N)$.

Observation 3.3

- A filter bank can be more effective in separating the various subchannels in frequency, even if more costly from the point of view of the computational complexity.

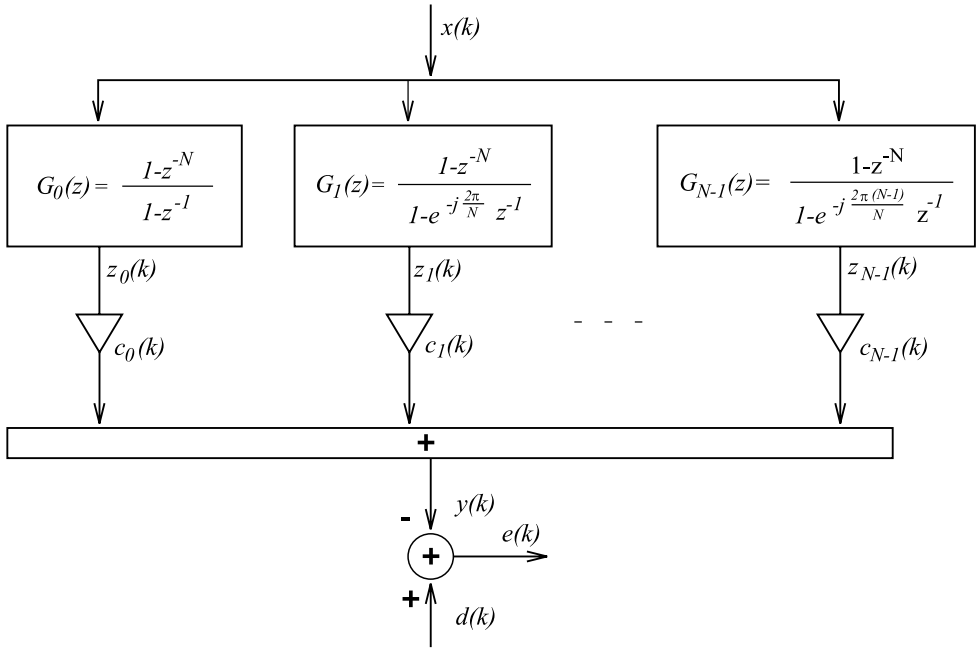


Figure 3.24. Adaptive filter in the frequency domain.

- There are versions of the algorithm where each output $z_i(k)$ is decimated, with the aim of reducing the number of operations.
- If $\{x(k)\}$ and $\{d(k)\}$ are real-valued signals, the filter coefficients satisfy the Hermitian property:

$$c_i(k) = c^*_{N-1-i}(k) \qquad i = 0, 1, \ldots, \frac{N-1}{2} \tag{3.255}$$

3.5.4 LMS algorithm in the DCT domain

The LMS algorithm in the DCT domain is obtained by filtering the input by the filter bank of Figure 3.24, where the i-th filter has impulse response and transfer function given by, respectively,

$$g_i(k) = \cos \frac{\pi(2k+1)i}{2N} \qquad k = 0, 1, \ldots, N-1 \tag{3.256}$$

and

$$G_i(z) = \mathcal{Z}[g_i(k)] = \frac{(1-z^{-1})(1-(-1)^i z^{-N}) \cos\left(\dfrac{\pi}{2N} i\right)}{1 - 2\cos\left(\dfrac{\pi}{N}\right) z^{-1} + z^{-2}} \tag{3.257}$$

Correspondingly, we have

$$z_0(k) = \frac{\sqrt{2}}{N} \sum_{m=0}^{N-1} x(k-m) \qquad i = 0 \tag{3.258}$$

$$z_i(k) = \frac{2}{N} \sum_{m=0}^{N-1} x(k-m) \cos \frac{\pi(2m+1)i}{2N} \qquad i = 1, 2, \ldots, N-1 \tag{3.259}$$

Ignoring the gain factor $\cos((\pi/2N)i)$, that can be included in the coefficient c_i, even the filtering operation determined by $G_i(z)$ can be implemented recursively [12]. We note that, if all the signals are real, the scheme can be implemented by using real arithmetic.

3.5.5 General observations

- Orthogonalization algorithms are useful if the input has a large eigenvalue spread and fast adaptation is required.
- If the signals exhibit time-varying statistical parameters, usually these methods do not offer any advantage over the standard LMS algorithm.
- In general, they require larger computational complexity than the standard LMS.

3.6 Examples of application

We give now some examples of applications of the algorithms investigated in this chapter [1, 25, 29, 30].

3.6.1 System identification

We want to determine the relation between the input x and the output z of the system illustrated in Figure 3.25. We note that the observation d is affected by additive noise w, having zero mean and variance σ_w^2, assumed statistically independent of x.

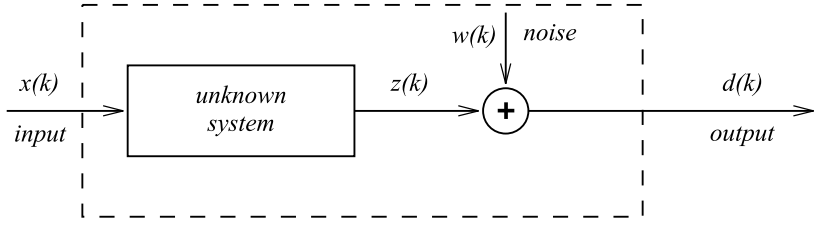


Figure 3.25. System model in which we want to identify the relation between x and z.

Linear case

Assuming the system between $z(k)$ and $x(k)$ can be modelled as a FIR filter, the experiment illustrated in Figure 3.26 can be adopted to estimate the filter impulse response.

Using an input x, known to both systems, we determine the output of the transversal filter c with N coefficients

$$y(k) = \sum_{i=0}^{N-1} c_i(k)x(k-i) = \mathbf{c}^T(k)\mathbf{x}(k) \tag{3.260}$$

and the estimation error

$$e(k) = d(k) - y(k) \tag{3.261}$$

The LMS adaptation equation follows,

$$\mathbf{c}(k+1) = \mathbf{c}(k) + \mu e(k)\mathbf{x}^*(k) \tag{3.262}$$

We analyze the specific case of an unknown linear FIR system whose impulse response has N_h coefficients. Assuming $N \geq N_h$, we introduce the vector $\mathbf{h}$ with N components,

$$\mathbf{h}^T = [h_0, h_1, \ldots, h_{N_h-1}, 0, \ldots, 0] \tag{3.263}$$

In this case,

$$\begin{aligned} d(k) &= h_0x(k) + h_1x(k-1) + \cdots + h_{N_h-1}x(k-(N_h-1)) + w(k) \\ &= \mathbf{h}^T\mathbf{x}(k) + w(k) \end{aligned} \tag{3.264}$$

For $N \geq N_h$, and assuming the input $\mathbf{x}$ is white noise[11] with statistical power $\mathtt{r}_x(0)$, we get

$$\mathbf{R} = E[\mathbf{x}^*(k)\mathbf{x}^T(k)] = \mathtt{r}_x(0)\mathbf{I} \tag{3.265}$$

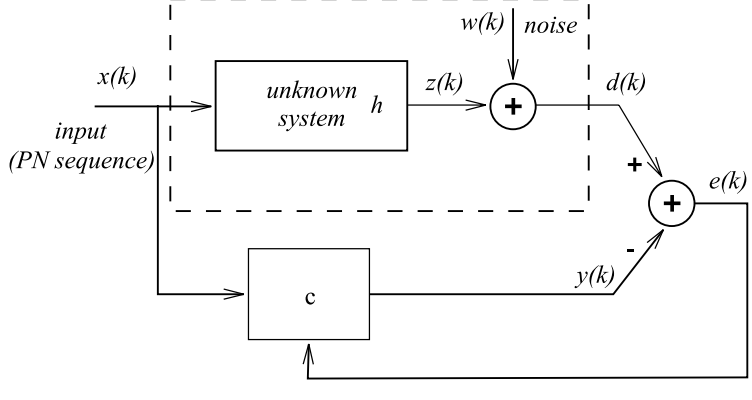


Figure 3.26. Adaptive scheme to estimate the impulse response of the unknown system.

[11] Typically x is generated by repeating a PN sequence of length $L > N_h$ (see Appendix 3.A).

and

$$\mathbf{p} = E[d(k)\mathbf{x}^*(k)] = r_x(0)\mathbf{h} \tag{3.266}$$

Then the Wiener–Hopf solution to the system identification problem is given by

$$\mathbf{c}_{opt} = \mathbf{R}^{-1}\mathbf{p} = \mathbf{h} \tag{3.267}$$

and

$$J_{min} = \sigma_w^2 \tag{3.268}$$

From (3.267) we see that the noise w does not affect the solution $\mathbf{c}_{opt}$, consequently the expectation of (3.262) for $k \to \infty$ (equal to $\mathbf{c}_{opt}$) is also not affected by w. Anyway, as seen in Section 3.1.3, the noise influences the convergence process and the solution obtained by the adaptive LMS algorithm. The larger the power of the noise, the smaller μ must be so that $\mathbf{c}(k)$ approaches $E[\mathbf{c}(k)]$. In any case $J(\infty) \neq 0$.

On the other hand, if $N < N_h$ then $\mathbf{c}_{opt}$ in (3.267) coincides with the first N coefficients of h, and

$$J_{min} = \sigma_w^2 + r_x(0)\,\|\mathbf{\Delta h}(\infty)\|^2 \tag{3.269}$$

where $\mathbf{\Delta h}(\infty)$ represents the residual error vector,

$$\mathbf{\Delta h}(\infty) = [0, \ldots, 0, -h_N, -h_{N+1}, \ldots, -h_{N_h-1}]^T \tag{3.270}$$

As the input x is white, the convergence behavior of the LMS algorithm (3.262) is easily determined. Let γ be defined as in (3.79):

$$\gamma = 1 + r_x(0)(\mu^2 N r_x(0) - 2\mu)$$

Let $\mathbf{\Delta c}(k) = \mathbf{c}(k) - \mathbf{c}_{opt}$; then we get

$$J(k) = E[|e(k)|^2] = J_{min} + r_x(0)E[\|\mathbf{\Delta c}(k)\|^2] \tag{3.271}$$

where

$$E[\|\mathbf{\Delta c}(k)\|^2] = \gamma^k\, E[\|\mathbf{\Delta c}(0)\|^2] + \mu^2 N\, r_x(0)\, J_{min}\, \frac{1-\gamma^k}{1-\gamma} \qquad k \geq 0 \tag{3.272}$$

The result (3.272) is obtained by (3.70) and the following assumptions:

1. $\mathbf{\Delta c}(k)$ is statistically independent of $\mathbf{x}(k)$;
2. $e_{min}(k)$ is orthogonal to $\mathbf{x}(k)$;
3. the approximation $\mathbf{x}^T(k)\,\mathbf{x}^*(k) \simeq N\, r_x(0)$ holds.

Indeed, (3.272) is an extension of (3.78). At convergence, for $\mu\, r_x(0) \ll 1$, it results in

$$E[\|\mathbf{\Delta c}(\infty)\|^2] = \mu \frac{N}{2} J_{min} \tag{3.273}$$

and

$$J(\infty) = J_{min}\left(1 + \mu\, r_x(0)\frac{N}{2}\right) \tag{3.274}$$

A faster convergence and a more accurate estimate, for fixed μ, are obtained by choosing a smaller value of N; this, however, may increase the residual estimation error (3.269).

Example 3.6.1

Consider an unknown system whose impulse response, given in Table 1.4 on page 26 as h_1, has energy equal to 1.06. The noise is additive, white, and Gaussian with statistical power $\sigma_w^2 = 0.01$. Identification via standard LMS and RLS adaptive algorithms is obtained using as input a maximal-length PN sequence of length $L = 31$ and unit power, $M_x = 1$. For a filter with $N = 5$ coefficients, the convergence curves of the mean-square error (estimated over 500 realizations) are shown in Figure 3.27. For the LMS algorithm, $\mu = 0.1$ is chosen, which leads to a misadjustment equal to MSD = 0.26.

As discussed in Appendix 3.B, as index of the estimate quality we adopt the ratio:

$$\Lambda_n = \frac{\sigma_w^2}{E[\|\mathbf{\Delta h}\|^2]} \tag{3.275}$$

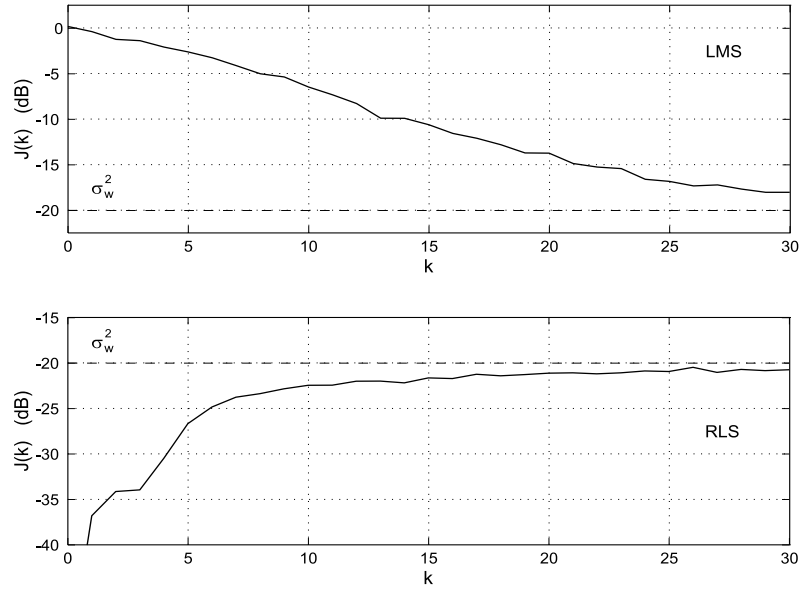


Figure 3.27. Convergence curves of the mean-square error for system identification using LMS and RLS.

where $\mathbf{\Delta h} = \mathbf{c} - \mathbf{h}$ is the estimate error vector. At convergence, that is for $k = 30$ in our example, it results:

$$\Lambda_n = \begin{cases} 3.9 & \text{for LMS} \\ 7.8 & \text{for RLS} \end{cases} \tag{3.276}$$

We note that, even if the input signal is white, the RLS algorithm usually yields a better estimate than the LMS. However, for systems with a large noise power and/or slow time-varying impulse responses, the two methods tend to give the same performance in terms of speed of convergence and error in steady state. As a result it is usually preferable to adopt the LMS algorithm, as it leads to easier implementation.

Finite alphabet case

Assume a more general, non-linear relation between $z(k)$ and $x(k)$, given by

$$z(k) = g[x(k), x(k-1), x(k-2)] = g(\mathbf{x}(k)) \tag{3.277}$$

where $x(i) \in \mathcal{A}$, finite alphabet with M elements. Then $z(k)$ assumes values in an alphabet with at most M^3 values, which can be identified by a table or *random-access memory* (RAM) method, as illustrated in Figure 3.28. The cost function to be minimized is expressed as

$$E[|e(k)|^2] = E[|d(k) - \hat{g}(\mathbf{x}(k))|^2] \tag{3.278}$$

and the gradient estimate is given by

$$\nabla_{\hat{g}} |e(k)|^2 = -2e(k) \tag{3.279}$$

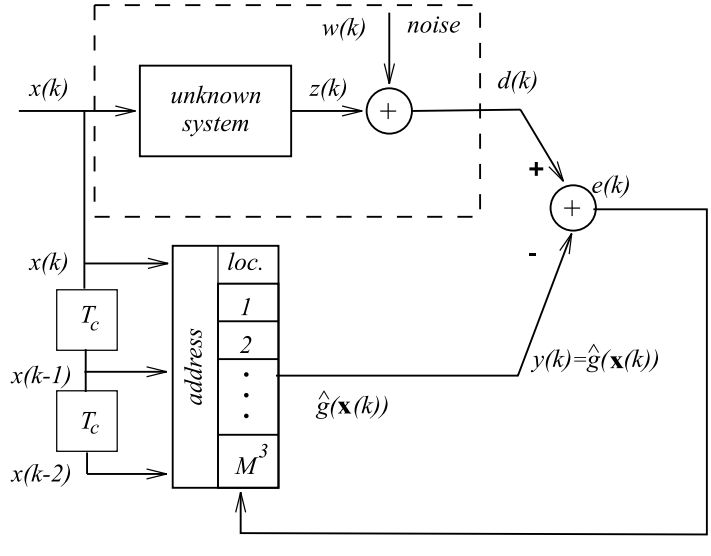


Figure 3.28. Adaptive scheme to estimate the input--output relation of a system.

Therefore the LMS adaptation equation becomes

$$\hat{g}(\mathbf{x}(k)) = \hat{g}(\mathbf{x}(k)) + \mu e(k) \tag{3.280}$$

In other words, the input vector $\mathbf{x}(k)$ identifies a particular RAM location whose content is updated by adding a term proportional to the error. In the absence of noise, if the RAM is initialized to zero, the content of a memory location can be immediately identified by looking at the output. In practice, however, it is necessary to access each memory location several times to average out the noise. We note that, if the sequence $\{x(k)\}$ is i.i.d., $\mathbf{x}(k)$ selects in the average each RAM location the same number of times.

An alternative method consists of setting $y(k) = 0$ during the entire time interval devoted to system identification, and to update the RAM with the values of $\{d(k)\}$, according to the equation

$$\hat{g}(\mathbf{x}(k)) = \hat{g}(\mathbf{x}(k)) + d(k) \qquad k = 0, 1, \ldots \tag{3.281}$$

To complete the identification process, the value at each RAM location is scaled by the number of updates that have taken place for that location. This is equivalent to considering

$$\hat{g}(\mathbf{x}) = E[g(\mathbf{x}) + w] \tag{3.282}$$

We note that this method is a block version of the LMS algorithm with block length equal to the input sequence, where the RAM is initialized to zero, so that $e(k) = d(k)$, and μ is given by the relative frequency of each address.

Observation 3.4
In this section and in Appendix 3.B, the observation d and the input x are determined on the same time domain with sampling period T_c. Often, however, the input is determined on the domain with sampling period T_c, and the system output signal is determined on T_c/F_0. Using the polyphase representation (see Section 1.A.9) of d, it is convenient to represent the estimate of h determined on T_c/F_0 as F_0 estimates determined on T_c.

3.6.2 Adaptive cancellation of interfering signals

With reference to Figure 3.29, we consider two sensors:

1. *Primary input*, consisting of the desired signal s corrupted by additive noise w_0,

$$d(k) = s(k) + w_0(k) \qquad \text{with } s \perp w_0 \tag{3.283}$$

2. *Reference input*, consisting of the noise signal w_1, with $s \perp w_1$. We assume that w_0 and w_1 are in general correlated.

w_1 is filtered by an adaptive filter with coefficients $\{c_i\}$, $i = 0, 1, \ldots, N-1$, so that the filter output, given by

$$y(k) = \sum_{i=0}^{N-1} c_i(k) w_1(k-i) \tag{3.284}$$

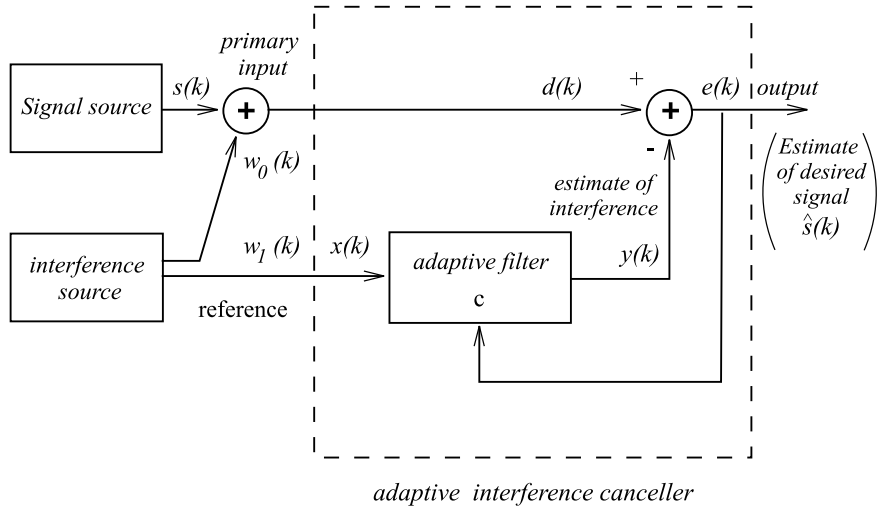


Figure 3.29. General configuration of an interference canceller.

is the most accurate replica of $w_0(k)$. Defining the error

$$e(k) = d(k) - y(k) = s(k) + w_0(k) - y(k) \tag{3.285}$$

the cost function, assuming real-valued signals and recalling that s is orthogonal to the noise signals, is given by

$$J = E[e^2(k)] = E[s^2(k)] + E[(w_0(k) - y(k))^2] \tag{3.286}$$

We have two cases.

1. w_1 and w_0 are correlated:

$$\min_{\mathbf{c}} J = \mathrm{r}_s(0) + \min_{\mathbf{c}} E[(w_0(k) - y(k))^2] = \mathrm{r}_s(0) \tag{3.287}$$

   for $y(k) = w_0(k)$. In this case $e(k) = s(k)$.

2. w_1 and w_0 are uncorrelated:

$$\begin{aligned} \min_{\mathbf{c}} J &= E[(s(k) + w_0(k))^2] + \min_{\mathbf{c}} E[y^2(k)] \\ &= E[(s(k) + w_0(k))^2] \end{aligned} \tag{3.288}$$

   for $y(k) = 0$. In this case $e(k) = d(k)$ and the noise w_0 is not cancelled.

General solution

With reference to Figure 3.30, for a general input x to the adaptive filter, the Wiener–Hopf solution in the z-transform domain is given by (see (2.50))

$$C_{opt}(z) = \frac{P_{dx}(z)}{P_x(z)} \tag{3.289}$$

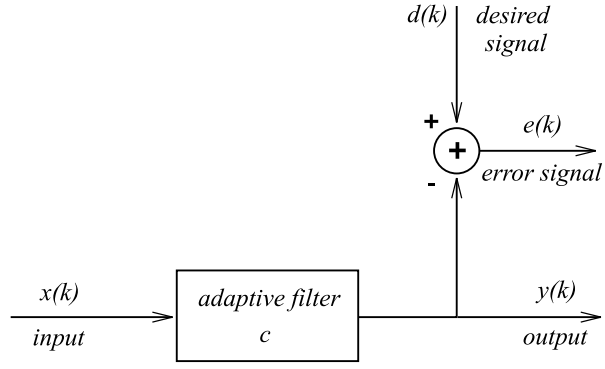


Figure 3.30. Block diagram of an adaptive cancellation scheme.

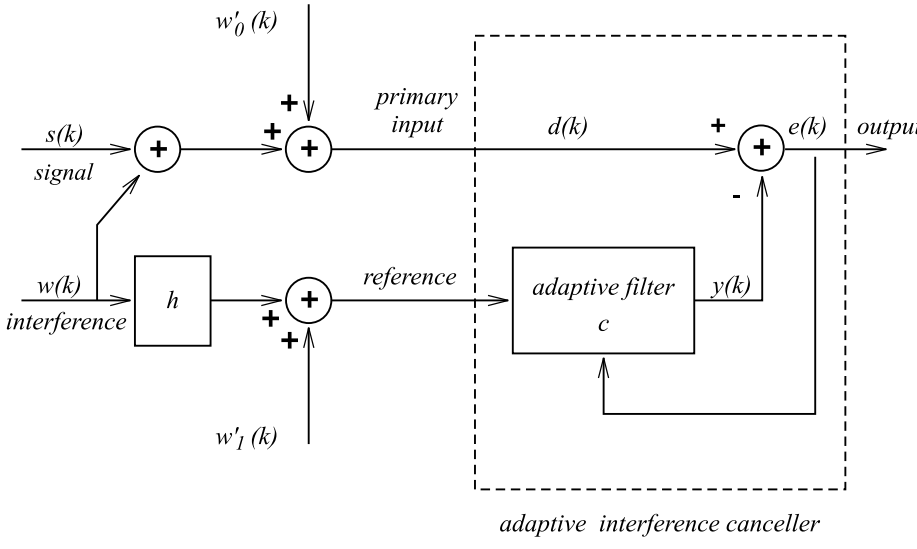


Figure 3.31. Specific configuration of an interference canceller.

Adopting for d and x the model of Figure 3.31, in which w'_0 and w'_1 are additive noise signals uncorrelated with w and s, and using Table 1.3, (3.289) becomes

$$C_{opt}(z) = \frac{P_w(z)H^*(1/z^*)}{P_{w'_1}(z) + P_w(z)H(z)H^*(1/z^*)} \tag{3.290}$$

If $w'_1 \equiv 0$, (3.290) becomes

$$C_{opt}(z) = \frac{1}{H(z)} \tag{3.291}$$

3.6.3 Cancellation of a sinusoidal interferer with known frequency

Let

$$d(k) = s(k) + A\cos(2\pi f_0 k T_c + \varphi_0) \tag{3.292}$$

where s is the desired signal, and the sinusoidal term is the interferer. As shown in Figure 3.32, we take as reference signals

$$x_1(k) = B\cos(2\pi f_0 k T_c + \varphi) \tag{3.293}$$

and

$$x_2(k) = B\sin(2\pi f_0 k T_c + \varphi) \tag{3.294}$$

The adaptation equations of the LMS algorithm are

$$c_1(k+1) = c_1(k) + \mu e(k) x_1(k) \tag{3.295}$$

$$c_2(k+1) = c_2(k) + \mu e(k) x_2(k) \tag{3.296}$$

At convergence, the two coefficients c_1 and c_2 change the amplitude and phase of the reference signal to cancel the interfering tone. The relation between d and output e corresponds to a *notch* filter as illustrated in Figure 3.33.

It is easy to see that x_2 is obtained from x_1 via a Hilbert filter (see Figure 1.28). We note that in this case x_2 can be obtained as a delayed version of x_1.

3.6.4 Disturbance cancellation for speech signals

With reference to Figure 3.34, the primary signal is a speech waveform affected by interference signals such as echoes and/or environmental disturbances. The reference signal

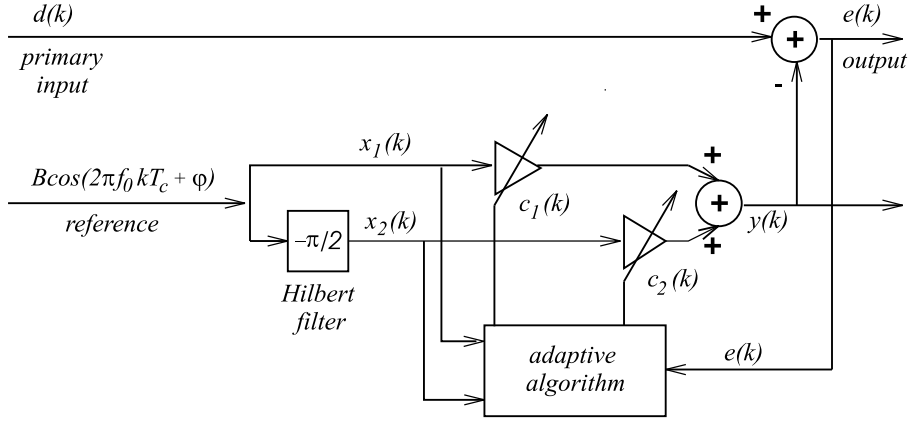


Figure 3.32. Configuration to cancel a sinusoidal interferer of known frequency.

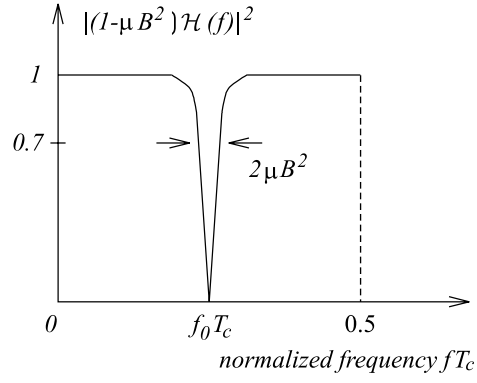


Figure 3.33. Frequency response of a notch filter.

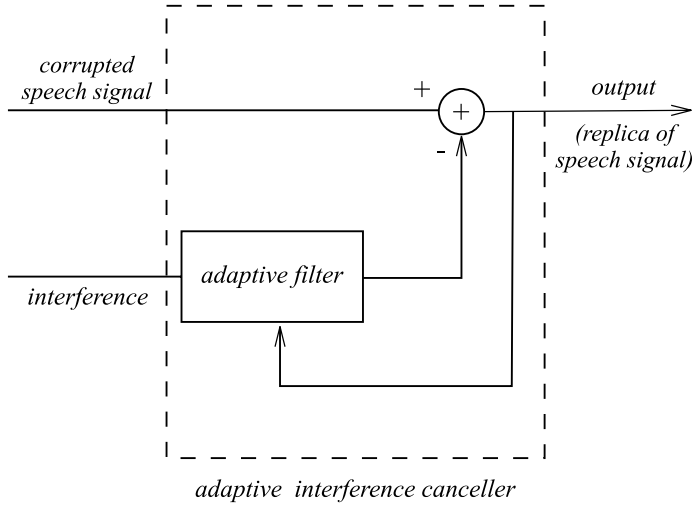


Figure 3.34. Disturbance cancellation for speech signals.

consists of a replica of the disturbances. At convergence, the adaptive filter output will attempt to subtract the interference signal, which is correlated to the reference signal, from the primary signal. The output signal is a replica of the speech waveform, obtained by removing to the best possible extent the disturbances from the input signal.

3.6.5 Echo cancellation in subscriber loops

With reference to the simplified scheme of Figure 3.35, the speech signal of user A is transmitted over a transmission line consisting of a pair of wires (*local loop*) [31] to the central office A, where the signals in the two directions of transmission, i.e. the signal transmitted by user A and the signal received from user B, are separated by a device called

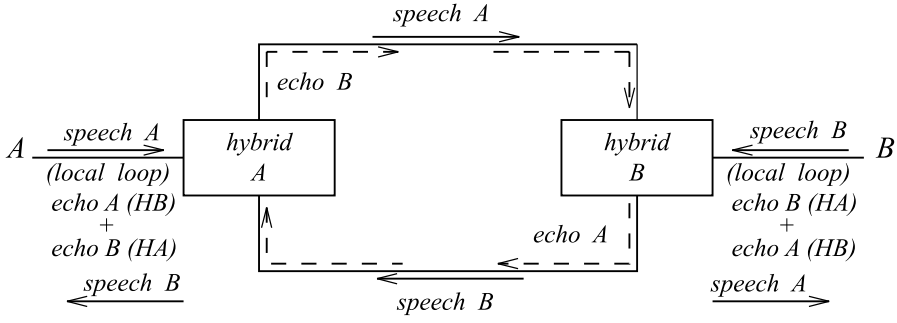


Figure 3.35. Transmission between two users in the public network.

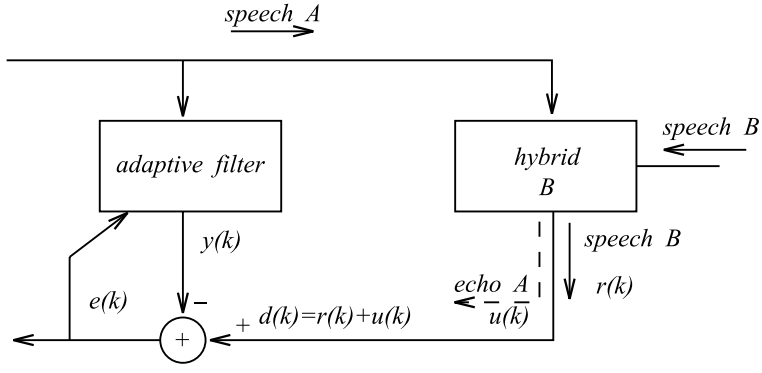


Figure 3.36. Configuration to remove the echo of signal A caused by the hybrid B.

hybrid. A similar situation takes place at the central office B, with the roles of the signals A and B reversed. Because of impedance mismatch, the hybrids give origin to echo signals that are added to the desired speech signals. For speech waveforms, the echo of signal A that is generated at the hybrid A can be ignored because it is not perceived by the human ear. The case for digital transmission is different, as will be discussed in Chapter 16. A method to remove echo signals is illustrated in Figure 3.36, where y is a replica of the echo. At convergence, e will consist of the speech signal B only.

3.6.6 Adaptive antenna arrays

In radio systems, to equalize the desired signal and remove interference, it is convenient to use several sensors, i.e. an antenna array, with the task of filtering signals in space, discriminating them through their angle of arrival. The signals of the array are then equalized to compensate for linear distortion introduced by the radio channel. A general scheme for wideband signals is illustrated in Figure 3.37. For narrowband signals, it is sufficient to substitute for each sensor the filter with a single complex-valued coefficient [32, 33] (see Section 8.18).

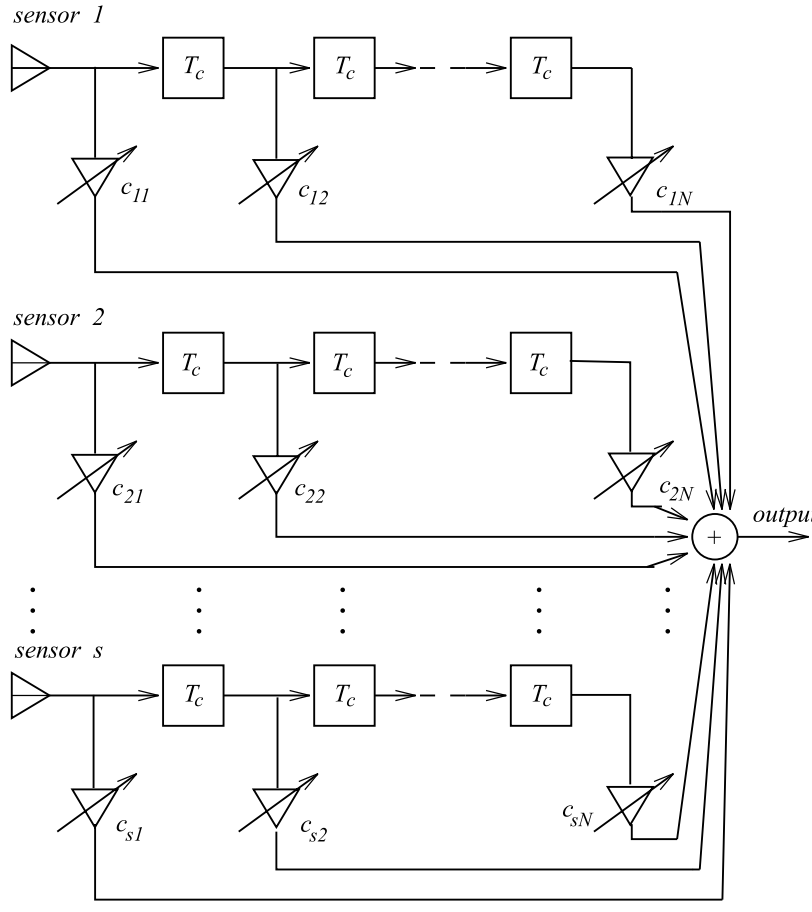


Figure 3.37. Antenna array to filter and equalize wideband radio signals.

3.6.7 Cancellation of a periodic interfering signal

For the cancellation of a periodic interfering signal, we can use the scheme of Figure 3.38, where:

- we note the absence of an external reference signal; the reference signal is generated by delaying the primary input;
- a delay $\Delta = DT_c$, where D is an integer, is needed to decorrelate the desired component of the *primary signal* from that of the *reference signal*, otherwise part of the desired signal would also be cancelled.

On the other hand, to cancel a wideband interferer from a periodic signal it is sufficient to take the output of the adaptive filter (see Figure 3.39).

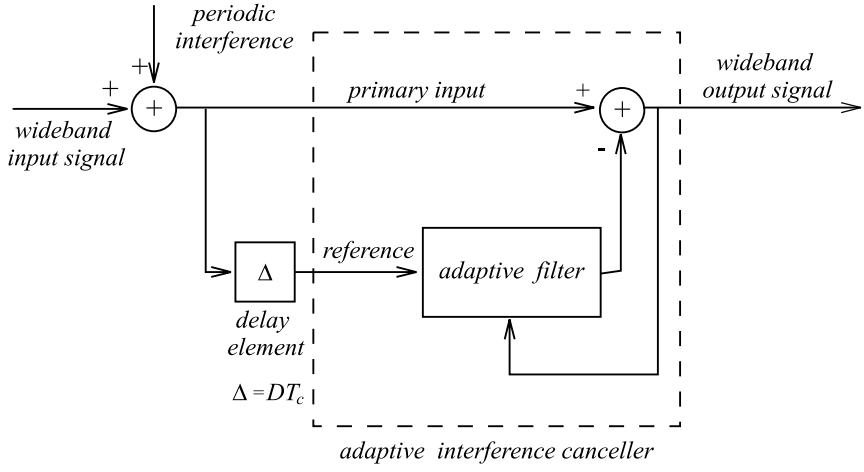


Figure 3.38. Scheme to remove a periodic interferer from a wideband desired signal.

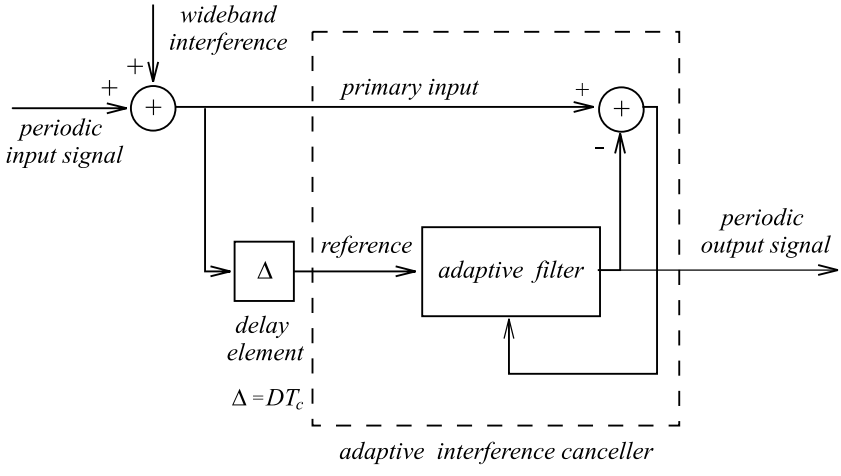


Figure 3.39. Scheme to remove a wideband interferer from a periodic desired signal.

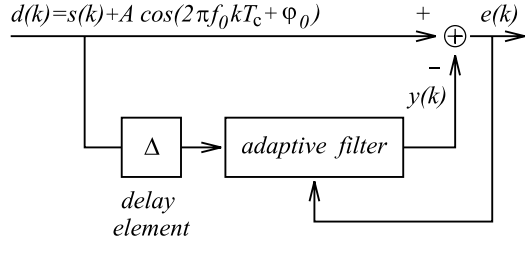


Figure 3.40. Scheme to remove a sinusoidal interferer from a wideband signal.

Note that in both schemes the adaptive filter acts as a predictor.

Exploiting the general concept described above, an alternative scheme to that of Figure 3.32 is illustrated in Figure 3.40, where the knowledge of the frequency of the interfering signal is not required. In general, for $D > 1$ the scheme of Figure 3.40 requires many more than two coefficients, therefore it has a higher implementation complexity than that of the scheme of Figure 3.32. However, if the wideband signal can be modeled as white noise, then $D = 1$; hence, observing (1.555), for a sinusoidal interferer a second-order predictor is sufficient.

Bibliography

[1] J. R. Treichler, C. R. Johnson Jr., and M. G. Larimore, *Theory and design of adaptive filters*. New York: John Wiley & Sons, 1987.

[2] J. J. Shynk, "Adaptive IIR filtering", *IEEE ASSP Magazine*, vol. 6, pp. 4–21, Apr. 1989.

[3] G. Ungerboeck, "Theory on the speed of convergence in adaptive equalizers for digital communication", *IBM Journal of Research and Development*, vol. 16, pp. 546–555, Nov. 1972.

[4] G. H. Golub and C. F. van Loan, *Matrix computations*. Baltimore and London: The Johns Hopkins University Press, 2nd ed., 1989.

[5] S. H. Ardalan and S. T. Alexander, "Fixed-point round-off error analysis of the exponentially windowed RLS algorithm for time varying systems", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 35, pp. 770–783, June 1987.

[6] E. Eweda, "Comparison of RLS, LMS and sign algorithms for tracking randomly time varying channels", *IEEE Trans. on Signal Processing*, vol. 42, pp. 2937–2944, Nov. 1994.

[7] W. A. Gardner, "Nonstationary learning characteristics of the LMS algorithm", *IEEE Trans. on Circuits and Systems*, vol. 34, pp. 1199–1207, Oct. 1987.

[8] V. Solo, "The limiting behavior of LMS", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 37, pp. 1909–1922, Dec. 1989.

[9] S. Haykin, *Neural networks: a comprehensive foundation*. New York: Macmillan Publishing Company, 1994.

[10] S. C. Douglas, "A family of normalized LMS algorithms", *IEEE Signal Processing Letters*, vol. 1, pp. 49–51, Mar. 1994.

[11] B. Porat and T. Kailath, "Normalized lattice algorithms for least-squares FIR system identification", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 31, pp. 122–128, Feb. 1983.

[12] S. Haykin, *Adaptive filter theory*. Englewood Cliffs, NJ: Prentice-Hall, 3rd ed., 1996.

[13] J. R. Zeidler, "Performance analysis of LMS adaptive prediction filters", *IEEE Proceedings*, vol. 78, pp. 1781–1806, Dec. 1990.

[14] D. Falconer and L. Ljung, "Application of fast Kalman estimation to adaptive equalization", *IEEE Trans. on Communications*, vol. 26, pp. 1439–1446, Oct. 1978.

[15] J. M. Cioffi and T. Kailath, "Fast, recursive-least-squares transversal filter for adaptive filtering", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 32, pp. 304–337, Apr. 1984.

[16] M. L. Honig and D. G. Messerschmitt, *Adaptive filters: structures, algorithms and applications*. Boston, MA: Kluwer Academic Publishers, 1984.

[17] J. M. Cioffi, "High speed systolic implementation of fast QR adaptive filters", in *Proc. ICASSP*, pp. 1584–1588, 1988.

[18] F. Ling, D. Manolakis, and J. G. Proakis, "Numerically robust least-squares lattice-ladder algorithm with direct updating of the reflection coefficients", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 34, pp. 837–845, Aug. 1986.

[19] P. A. Regalia, "Numerical stability properties of a QR-based fast least squares algorithm", *IEEE Trans. on Signal Processing*, vol. 41, pp. 2096–2109, June 1993.

[20] S. T. Alexander and A. L. Ghirnikar, "A method for recursive least-squares filtering based upon an inverse QR decomposition", *IEEE Trans. on Signal Processing*, vol. 41, pp. 20–30, Jan. 1993.

[21] J. M. Cioffi, "The fast adaptive rotor's RLS algorithm", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 38, pp. 631–653, Apr. 1990.

[22] Z.-S. Liu, "QR methods of $O(N)$ complexity in adaptive parameter estimation", *IEEE Trans. on Signal Processing*, vol. 43, pp. 720–729, Mar. 1995.

[23] J. A. Bucklew, T. G. Kurtz, and W. A. Sethares, "Weak convergence and local stability properties of fixed step size recursive algorithms", *IEEE Trans. on Information Theory*, vol. 39, pp. 966–978, May 1993.

[24] B. Widrow, "Fundamental relations between the LMS algorithm and the DFT", *IEEE Trans. on Circuits and Systems*, vol. 34, pp. 814–820, July 1987.

[25] C. F. N. Cowan and P. M. Grant, *Adaptive filters*. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[26] N. J. Bershad and P. L. Feintuch, "A normalized frequency domain LMS adaptive algorithm", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 34, pp. 452–461, June 1986.

[27] R. P. Bitmead and B. D. O. Anderson, "Adaptive frequency sampling filters", *IEEE Trans. on Circuits and Systems*, vol. 28, pp. 524–543, June 1981.

[28] D. F. Marshall, W. K. Jenkins, and J. J. Murphy, "The use of orthogonal transforms for improving performance of adaptive filters", *IEEE Trans. on Circuits and Systems*, vol. 36, pp. 474–484, Apr. 1989.

[29] B. Widrow and S. D. Stearns, *Adaptive signal processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[30] O. Macchi, *Adaptive processing: the LMS approach with applications in transmission*. New York: John Wiley & Sons, 1995.

[31] D. G. Messerschmitt, "Echo cancellation in speech and data transmission", *IEEE Journal on Selected Areas in Communications*, vol. 2, pp. 283–297, Mar. 1984.

[32] I. J. Gupta and A. A. Ksienski, "Adaptive antenna array for weak interfering signals", *IEEE Trans. on Antennas Propag.*, vol. 34, pp. 420–426, Mar. 1986.

[33] L. L. Horowitz and K. D. Senne, "Performance advantage of complex LMS for controlling narrow-band adaptive array", *IEEE Trans. on Circuits and Systems*, vol. 28, pp. 562–576, June 1981.

[34] D. G. Messerschmitt and E. A. Lee, *Digital communication*. Boston, MA: Kluwer Academic Publishers, 2nd ed., 1994.

[35] S. W. Golomb, *Shift register sequences*. San Francisco: Holden-Day, 1967.

[36] P. Fan and M. Darnell, *Sequence design for communications applications*. Taunton: Research Studies Press, 1996.

[37] D. C. Chu, "Polyphase codes with good periodic correlation properties", *IEEE Trans. on Information Theory*, vol. 18, pp. 531–532, July 1972.

[38] R. L. Frank and S. A. Zadoff, "Phase shift pulse codes with good periodic correlation properties", *IRE Trans. on Information Theory*, vol. 8, pp. 381–382, Oct. 1962.

[39] A. Milewsky, "Periodic sequences with optimal properties for channel estimation and fast start-up equalization", *IBM Journal of Research and Development*, vol. 27, pp. 426–431, Sept. 1983.

[40] R. L. Peterson, R. E. Ziemer, and D. E. Borth, *Introduction to spread spectrum communications*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[41] R. Gold, "Optimal binary sequences for spread spectrum multiplexing", *IEEE Trans. on Information Theory*, vol. 13, pp. 619–621, Oct. 1967.

[42] R. Gold, "Maximal recursive sequences with 3-valued recursive cross-correlation functions", *IEEE Trans. on Information Theory*, vol. 14, pp. 154–155, Jan. 1968.

[43] S. L. Marple Jr., "Efficient least squares FIR system identification", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 29, pp. 62–73, Feb. 1981.

[44] J. I. Nagumo and A. Noda, "A learning method for system identification", *IEEE Trans. on Automatic Control*, vol. 12, pp. 282–287, June 1967.

[45] S. N. Crozier, D. D. Falconer, and S. A. Mahmoud, "Least sum of squared errors (LSSE) channel estimation", *IEE Proceedings-F*, vol. 138, pp. 371–378, Aug. 1991.

[46] N. Benvenuto, "Distortion analysis on measuring the impulse response of a system using a cross-correlation method", *AT&T Bell Laboratories Technical Journal*, vol. 63, pp. 2171–2192, Dec. 1984.

Appendix 3.A *PN sequences*

In this Appendix we introduce three classes of deterministic periodic sequences having spectral characteristics similar to those of a white noise signal, hence the name *pseudo-noise* (PN) sequences.

Maximal-length sequences

Maximal-length sequences are binary PN sequences, also called *r-sequences*, that are generated recursively, e.g., using a *shift-register* (see page 877), and have period equal to $L = 2^r - 1$. Let $\{p(\ell)\}$, $\ell = 0, 1, \ldots, L-1$, $p(\ell) \in \{0, 1\}$, be the values assumed by the sequence in a period. It can be shown that the maximal-length sequences enjoy the following properties [34, 35].

- Every non-zero sequence of r bits appears exactly once in each period; therefore all binary sequences of r bits are generated, except the all zero sequence.
- The number of bits equal to "1" in a period is 2^{r-1}, and the number of bits equal to "0" is $2^{r-1} - 1$.
- A *subsequence* is intended here as a set of consecutive bits of the *r-sequence*. The relative frequency of any *non-zero* subsequence of length $i \leq r$ is

$$\frac{2^{r-i}}{2^r - 1} \simeq 2^{-i} \tag{3.297}$$

  and the relative frequency of a subsequence of length $i < r$ with *all bits equal to zero* is

$$\frac{2^{r-i} - 1}{2^r - 1} \simeq 2^{-i} \tag{3.298}$$

  In both formulae the approximation is valid for a sufficiently large r.
- The sum of two r-sequences, which are generated by the same shift-register, but with different initial conditions, is still an r-sequence.
- The *linear span*, that determines the predictability of a sequence, is equal to r [36]. In other words, the elements of a sequence can be determined by any $2r$ consecutive elements of the sequence itself, while the remaining elements can be produced by a recursive algorithm (see, e.g., the Berlekamp-Massey algorithm on page 891).

A practical example is given in Figure 3.41 for a sequence with $L = 15$ ($r = 4$), which is generated by the recursive equation

$$p(\ell) = p(\ell - 3) \oplus p(\ell - 4) \tag{3.299}$$

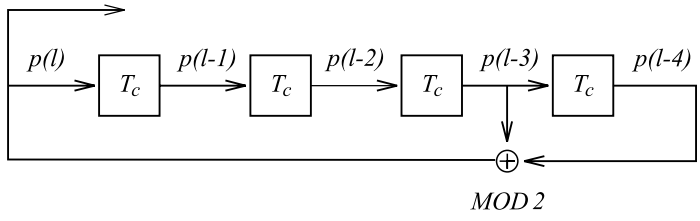


Figure 3.41. Generation of a PN sequence with period $L = 15$.

where $\oplus$ denotes *modulo 2 sum*. Assuming initial conditions $p(-1) = p(-2) = p(-3) = p(-4) = 1$, applying (3.299) we obtain the sequence

$$\underbrace{0}_{p(0)}\ \underbrace{0}_{p(1)}\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 1\ 1\ \underbrace{1}_{p(L-1)}\ \ldots \tag{3.300}$$

Obviously, the all zero initial condition must be avoided. To generate sequences with a larger period L we refer to Table 3.5. The above properties make an r-sequence, even if deterministic and periodic, appear as a random i.i.d. sequence from the point of view of the relative frequency of subsequences of bits. It turns out that an r-sequence appears as random i.i.d. also from the point of view of the autocorrelation function. In fact, mapping "0" to "-1" and "1" to "$+1$", we get the following correlation properties.

1. *Mean*

$$\frac{1}{L}\sum_{\ell=0}^{L-1} p(\ell) = \frac{1}{L} \tag{3.301}$$

2. *Correlation* (periodic of period L)

$$\mathfrak{r}_p(n) = \frac{1}{L}\sum_{\ell=0}^{L-1} p(\ell)p^*(\ell-n)_{mod\,L} = \begin{cases} 1 & \text{for } (n)_{mod\,L} = 0 \\ -\dfrac{1}{L} & \text{otherwise} \end{cases} \tag{3.302}$$

3. *Spectral density* (periodic of period L)

$$\mathcal{P}_p\left(m\frac{1}{LT_c}\right) = T_c\sum_{n=0}^{L-1} \mathfrak{r}_p(n)e^{-j2\pi m\frac{1}{LT_c}nT_c} = \begin{cases} T_c\dfrac{1}{L} & \text{for } (m)_{mod\,L} = 0 \\ T_c\left(1+\dfrac{1}{L}\right) & \text{otherwise} \end{cases} \tag{3.303}$$

We note that, with the exception of the values assumed for $(m)_{mod\,L} = 0$, the spectral density of maximal length sequences is constant.

Table 3.5 Recursive equations to generate PN sequences of length $L = 2^r - 1$, for different values of r.

r	Period $L = 2^r - 1$
1	$p(\ell) = p(\ell - 1)$
2	$p(\ell) = p(\ell - 1) \oplus p(\ell - 2)$
3	$p(\ell) = p(\ell - 2) \oplus p(\ell - 3)$
4	$p(\ell) = p(\ell - 3) \oplus p(\ell - 4)$
5	$p(\ell) = p(\ell - 3) \oplus p(\ell - 5)$
6	$p(\ell) = p(\ell - 5) \oplus p(\ell - 6)$
7	$p(\ell) = p(\ell - 6) \oplus p(\ell - 7)$
8	$p(\ell) = p(\ell - 2) \oplus p(\ell - 3) \oplus p(\ell - 4) \oplus p(\ell - 8)$
9	$p(\ell) = p(\ell - 5) \oplus p(\ell - 9)$
10	$p(\ell) = p(\ell - 7) \oplus p(\ell - 10)$
11	$p(\ell) = p(\ell - 9) \oplus p(\ell - 11)$
12	$p(\ell) = p(\ell - 2) \oplus p(\ell - 10) \oplus p(\ell - 11) \oplus p(\ell - 12)$
13	$p(\ell) = p(\ell - 1) \oplus p(\ell - 11) \oplus p(\ell - 12) \oplus p(\ell - 13)$
14	$p(\ell) = p(\ell - 2) \oplus p(\ell - 12) \oplus p(\ell - 13) \oplus p(\ell - 14)$
15	$p(\ell) = p(\ell - 14) \oplus p(\ell - 15)$
16	$p(\ell) = p(\ell - 11) \oplus p(\ell - 13) \oplus p(\ell - 14) \oplus p(\ell - 16)$
17	$p(\ell) = p(\ell - 14) \oplus p(\ell - 17)$
18	$p(\ell) = p(\ell - 11) \oplus p(\ell - 18)$
19	$p(\ell) = p(\ell - 14) \oplus p(\ell - 17) \oplus p(\ell - 18) \oplus p(\ell - 19)$
20	$p(\ell) = p(\ell - 17) \oplus p(\ell - 20)$

CAZAC sequences

The *constant amplitude zero autocorrelation* (CAZAC) sequences are complex-valued PN sequences with constant amplitude (assuming values on the unit circle) and autocorrelation function $\mathtt{r}_p(n)$ equal to zero for $(n)_{mod\, L} \neq 0$. Because of these characteristics they are also called *polyphase sequences* [37, 38, 39]. Let L and M be two integer numbers that are relatively prime. The CAZAC sequences are defined as,

$$\text{for } L \text{ even} \qquad p(\ell) = e^{j\frac{M\pi\ell^2}{L}} \qquad \ell = 0, 1, \ldots, L-1 \tag{3.304}$$

$$\text{for } L \text{ odd} \qquad p(\ell) = e^{j\frac{M\pi\ell(\ell+1)}{L}} \qquad \ell = 0, 1, \ldots, L-1 \tag{3.305}$$

It can be shown that, in both cases, these sequences have the following properties.

1. *Mean*

$$\frac{1}{L}\sum_{\ell=0}^{L-1} p(\ell) = 0 \tag{3.306}$$

2. *Correlation*

$$\mathtt{r}_p(n) = \begin{cases} 1 & \text{for } (n)_{mod\, L} = 0 \\ 0 & \text{otherwise} \end{cases} \tag{3.307}$$

3. *Spectral density*

$$\mathcal{P}_p\left(m\frac{1}{LT_c}\right) = T_c \tag{3.308}$$

Gold sequences

In a large number of applications, as for example in *spread-spectrum* systems with code-division multiple access (see Chapter 10), *sets of sequences* having one or both of the following properties [40] are required.

- Each sequence of the set must be easily distinguishable from *its own time shifted versions*.
- Each sequence of the set must be easily distinguishable from *any other sequence* of the set and from its time-shifted versions.

An important class of periodic binary sequences that satisfy these properties, or, in other words, that have good autocorrelation and cross-correlation characteristics, is the set of Gold sequences [41, 42].

Construction of pairs of preferred r-sequences. In general the cross-correlation sequence (CCS) between two r-sequences may assume three, four or maybe even a greater number of values. We show now the construction of a pair of r-sequences, called *preferred r-sequences* [36], whose CCS assumes only three values. Let $a = \{a(\ell)\}$ be an r-sequence with period $L = 2^r - 1$. We define now another r-sequence of length $L = 2^r - 1$ obtained from the sequence a by decimation by a factor M, that is:

$$b = \{b(\ell)\} = \{a(M\ell)_{mod\, L}\} \tag{3.309}$$

We make the following assumptions.

- $r_{mod\, 4} \neq 0$, that is r must be odd or equal to odd multiples of 2, i.e. $r_{mod\, 4} = 2$.
- The factor M satisfies one of the following properties:

$$M = 2^k + 1 \quad \text{or} \quad M = 2^{2k} - 2^k + 1 \qquad k \text{ integer} \tag{3.310}$$

- For k determined as in the (3.310), defining g.c.d.(r, k) as the greatest common divisor of r and k, let

$$e = \text{g.c.d.}(r, k) = \begin{cases} 1 & r \text{ odd} \\ 2 & r_{mod\,4} = 2 \end{cases} \tag{3.311}$$

Then the CCS between the two r-sequences a and b assumes only three values [35, 36]:

$$\begin{aligned} \mathtt{r}_{ab}(n) &= \frac{1}{L}\sum_{\ell=0}^{L-1} a(\ell)b^*(\ell - n)_{mod\,L} \\ &= \frac{1}{L}\begin{cases} -1 + 2^{\frac{r+e}{2}} & (\text{value assumed } 2^{r-e-1} + 2^{\frac{r+e-2}{2}} \text{ times}) \\ -1 & (\text{value assumed } 2^r - 2^{r-e} - 1 \text{ times}) \\ -1 - 2^{\frac{r+e}{2}} & (\text{value assumed } 2^{r-e-1} - 2^{\frac{r+e-2}{2}} \text{ times}) \end{cases} \end{aligned} \tag{3.312}$$

Example 3.A.1 (Construction of a pair of preferred r-sequences)
Let the following r-sequence of period $L = 2^5 - 1 = 31$ be given:

$$\{a(\ell)\} = (0000100101100111110001101110101) \tag{3.313}$$

As $r = 5$ and $r_{mod\,4} = 1$, we take $k = 1$. Therefore $e = \text{g.c.d.}(r, k) = \text{g.c.d.}(5, 1) = 1$ and $M = 2^k + 1 = 2^1 + 1 = 3$. The sequence $\{b(\ell)\}$ obtained by decimation of the sequence $\{a(\ell)\}$ is then given by

$$\{b(\ell)\} = \{a(3\ell)_{mod\,L}\} = (0001010110100001100100111110111) \tag{3.314}$$

The CCS between the two sequences, assuming "0" is mapped to "−1", is:

$$\begin{aligned} \{\mathtt{r}_{ab}(n)\} = \frac{1}{31}(&7, 7, -1, -1, -1, -9, 7, -9, 7, 7, -1, -1, 7, 7, -1, 7, -1, \\ &-1, -9, -1, -1, -1, -1, -9, -1, 7, -1, -9, -9, 7, -1) \end{aligned} \tag{3.315}$$

We note that, if we had chosen $k = 2$, then $e = \text{g.c.d.}(5, 2) = 1$ and $M = 2^2 + 1 = 5$, or else $M = 2^{2\cdot 2} - 2^2 + 1 = 13$.

Construction of a set of Gold sequences. A set of Gold sequences can be constructed from any pair $\{a(\ell)\}$ and $\{b(\ell)\}$ of preferred r-sequences of period $L = 2^r - 1$. We define the set of sequences:

$$G(a, b) = \{a, b, a \oplus b, a \oplus Zb, a \oplus Z^2 b, \ldots, a \oplus Z^{L-1} b\} \tag{3.316}$$

where Z is the *shift* operator that cyclically shifts a sequence to the left by a position. The set (3.316) contains $L + 2 = 2^r + 1$ sequences of length $L = 2^r - 1$ and is called the *set of Gold sequences*. It can be proved [41, 42] that, for the two sequences $\{a'(\ell)\}$ and $\{b'(\ell)\}$

belonging to the set $G(a, b)$, the CCS as well as the ACS, with the exception of zero lag, assume only three values:

$$\mathtt{r}_{a'b'}(n) = \frac{1}{L}\begin{cases} -1 & -1-2^{\frac{r+1}{2}} & -1+2^{\frac{r+1}{2}} & r \text{ odd} \\ -1 & -1-2^{\frac{r+2}{2}} & -1+2^{\frac{r+2}{2}} & r_{mod\,4}=2 \end{cases} \tag{3.317}$$

Clearly, the ACS of a Gold sequence no longer has the characteristics of an r-sequence, as is seen in the next example.

Example 3.A.2 (Gold sequence properties)
Let $r = 5$, hence $L = 2^5 - 1 = 31$. From Example 3.A.1, the two sequences (3.313) and (3.314) are a pair of preferred r-sequences, from which it is possible to generate the whole set of Gold sequences. For example we calculate the ACS of $\{a(\ell)\}$ and $\{b'(\ell)\} = \{a(\ell) \oplus b(\ell-2)\} = a \oplus Z^2 b$, and the CCS between $\{a(\ell)\}$ and $\{b'(\ell)\}$:

$$\begin{aligned}\{a(\ell)\}=(&-1,-1,-1,-1,1,-1,-1,1,-1,1,1,-1,-1,1,1,1,1,1,-1,\\ &-1,-1,1,1,-1,1,1,1,-1,1,-1,1)\end{aligned} \tag{3.318}$$

$$\begin{aligned}\{b'(\ell)\}=(&-1,-1,1,-1,-1,-1,1,-1,-1,-1,1,-1,-1,1,-1,-1,1,\\ &1,1,-1,-1,-1,-1,1,-1,-1,1,1,-1,1,1)\end{aligned} \tag{3.319}$$

$$\begin{aligned}\{\mathtt{r}_a(n)\}=\frac{1}{31}(&31,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,\\ &-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1)\end{aligned} \tag{3.320}$$

$$\begin{aligned}\{\mathtt{r}_{b'}(n)\}=\frac{1}{31}(&31,-1,-9,7,7,-9,-1,7,-1,-9,7,7,-1-1,7,-1,-1,7,\\ &-1,-1,7,7,-9,-1,7,-1,9,7,7,-9,-1)\end{aligned} \tag{3.321}$$

$$\begin{aligned}\{\mathtt{r}_{ab'}(n)\}=\frac{1}{31}(&-1,7,7,7,-1,-1,-1,-1,-1,7,-9,-1,-1,7,-1,-9,7,\\ &-1-9,7,7,-9,-1,7,-1,-9,-1,-1,-1,-9,-1)\end{aligned} \tag{3.322}$$

Appendix 3.B *Identification of a FIR system by PN sequences*

3.B.1 Correlation method

With reference to (3.264), which describes the relation between input and output of an unknown system with impulse response $\{h_i\}$, $i = 0, 1, \ldots, N-1$, we take as an input signal *white noise* with statistical power $\mathtt{r}_x(0)$. To estimate the impulse response of a linear system, we observe that the cross-correlation between d and x is then proportional, with a factor $\mathtt{r}_x(0)$, to the impulse response $\{h_i\}$. In fact, we have

$$\mathtt{r}_{dx}(n) = \mathtt{r}_{zx}(n) = \mathtt{r}_x * h(n) = \mathtt{r}_x(0) h_n \tag{3.323}$$

In practice, instead of noise a PN sequence with period L, $\{p(i)\}$, $i = 0, 1, \ldots, L-1$, is used as input. We recall that the autocorrelation of a PN sequence is also periodic with period L and is given by (see Appendix 3.A):

$$\mathtt{r}_p(n) \begin{cases} = 1 & n = 0, L, 2L, \ldots \\ \simeq 0 & n = 1, \ldots, L-1, L+1, \ldots \end{cases} \tag{3.324}$$

Moreover, we recall that if the input to a time-invariant filter is periodic with period L, the output will also be periodic with period L. To estimate the impulse response $\{h_i\}$, $i = 0, 1, \ldots, N-1$, we consider the scheme illustrated in Figure 3.42, where we choose $L \geq N$, and an input sequence x with length of at least $(L+1)N$ samples, obtained by repeating $\{p(i)\}$; in other words, $x(k) = p(k)_{mod\, L}$. We assume a delay $m \in \{0, 1, \ldots, N-1\}$, a rectangular window $g_{Rc}(k) = (1/L)\mathtt{w}_L(k)$, and that the system is started at instant $k = 0$. For $k \geq (N-1) + (L-1)$, the output $v(k)$ is given by

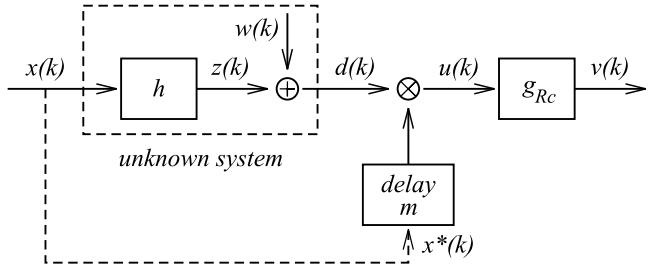


Figure 3.42. Correlation method to estimate the impulse response of an unknown system.

$$
\begin{aligned}
v(k) &= \sum_{\ell=0}^{L-1} \frac{1}{L}\, u(k-\ell) = \sum_{\ell=0}^{L-1} \frac{1}{L}\, d(k-\ell)\, p^*(k-\ell-m)_{mod\, L} \\
&= \sum_{\ell=0}^{L-1} \frac{1}{L} \left[\sum_{i=0}^{N-1} h_i\, p(k-\ell-i)_{mod\, L}\; p^*(k-\ell-m)_{mod\, L} \right. \\
&\quad \left. + w(k-\ell)\, p^*(k-\ell-m)_{mod\, L} \right] \qquad (3.325) \\
&= \sum_{i=0}^{N-1} h_i\, \frac{1}{L} \sum_{\ell=0}^{L-1} p(k-\ell-i)_{mod\, L}\; p^*(k-\ell-m)_{mod\, L} \\
&\quad + \frac{1}{L} \sum_{\ell=0}^{L-1} w(k-\ell)\, p^*(k-\ell-m)_{mod\, L}
\end{aligned}
$$

As

$$
\frac{1}{L} \sum_{\ell=0}^{L-1} p(k-\ell-i)_{mod\, L}\; p^*(k-\ell-m)_{mod\, L} = \mathtt{r}_p(m-i)_{mod\, L} \qquad (3.326)
$$

(3.325) becomes

$$
v(k) = \sum_{i=0}^{N-1} h_i\, \mathtt{r}_p(m-i)_{mod\, L} + \frac{1}{L} \sum_{\ell=0}^{L-1} w(k-\ell)\, p^*(k-\ell-m)_{mod\, L} \qquad (3.327)
$$

If $L \gg 1$, the second term on the right-hand side of (3.327) can be ignored, hence observing (3.324) we get

$$
v(k) \simeq h_m \qquad (3.328)
$$

Mean and variance of the estimate of h_m given by (3.327) are obtained as follows.

1. *Mean*

$$
E[v(k)] = \sum_{i=0}^{N-1} h_i\, \mathtt{r}_p(m-i)_{mod\, L} \qquad (3.329)
$$

assuming w has zero mean.

2. *Variance*

$$
\mathrm{var}[v(k)] = \mathrm{var}\left[\frac{1}{L} \sum_{\ell=0}^{L-1} w(k-\ell)\, p^*(k-\ell-m)_{mod\, L} \right] \simeq \frac{\sigma_w^2}{L} \qquad (3.330)
$$

assuming w white and $|p(\ell)| \leq 1$.

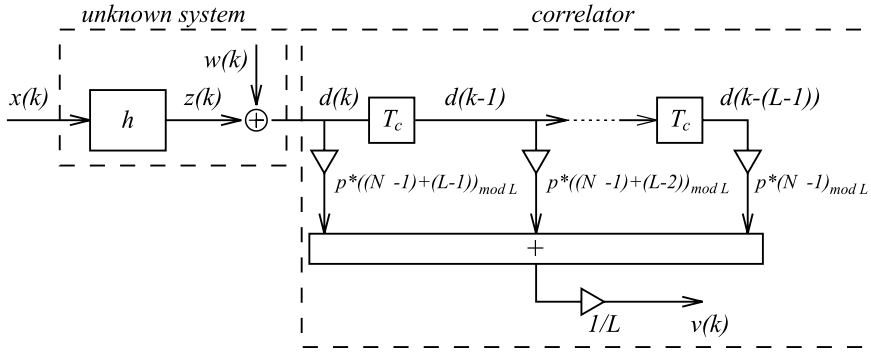


Figure 3.43. Correlation method via correlator to estimate the impulse response of an unknown system.

Using the scheme of Figure 3.42, varying m from 0 to $N-1$ it is possible to get an estimate of the samples of the impulse response of the unknown system $\{h_i\}$ at the output of the filter g_{Rc}. However, this scheme has two disadvantages:

1. it requires a very long computation time (NL);
2. it requires synchronization between the two PN sequences, at transmitter and receiver.

Both problems can be resolved by memorizing, after a transient equal to $N-1$ instants, L consecutive output samples $\{d(k)\}$ in a buffer and computing the correlation off-line:

$$\hat{r}_{dx}(m) = \frac{1}{L} \sum_{k=(N-1)}^{(N-1)+(L-1)} d(k)\, p^*(k-m)_{mod\,L} \simeq h_m \qquad m = 0, 1, \ldots, N-1 \quad (3.331)$$

An alternative scheme is represented in Figure 3.43: with steps analogous to those of the preceding scheme, we get

$$v(k) = \frac{1}{L} \sum_{\ell=0}^{L-1} d(k-(L-1)+\ell)\, p^*(\ell+(N-1))_{mod\,L} \simeq h_{(k-(N-1)-(L-1))_{mod\,L}} \quad (3.332)$$

After a transient of $(N-1)+(L-1)$ samples, from (3.332) we get

$$\hat{h}_i = v(i+(N-1)+(L-1)) \qquad i = 0, 1, \ldots, N-1 \quad (3.333)$$

In other words, the samples at the correlator output from instant $k = (N-1)+(L-1)$ to $2(N-1)+(L-1)$, give an estimate of the samples of the impulse response of the unknown system $\{h_i\}$.

Signal-to-estimation error ratio

Let $\mathbf{h}^T = [h_0, h_1, \ldots, h_{N-1}]$ be the filter coefficients to be estimated and $\hat{\mathbf{h}}^T = [\hat{h}_0, \hat{h}_1, \ldots, \hat{h}_{N-1}]$ those estimated. Let $\mathbf{\Delta h}$ be the estimation error vector

$$\mathbf{\Delta h} = \hat{\mathbf{h}} - \mathbf{h}$$

The quality of the estimate is measured by the signal-to-estimation error ratio

$$\Lambda_e = \frac{\|\mathbf{h}\|^2}{E[\|\mathbf{\Delta h}\|^2]} \tag{3.334}$$

On one hand, we have to take into consideration the noise present in the observed system and measured by (see Figure 3.42):

$$\Lambda = \frac{\mathtt{M}_x \|\mathbf{h}\|^2}{\sigma_w^2} \tag{3.335}$$

where $\mathtt{M}_x$ is the statistical power of the input signal. In our case $\mathtt{M}_x = 1$. Finally, we refer to the normalized ratio

$$\Lambda_n = \frac{\Lambda_e}{\Lambda} = \frac{\sigma_w^2}{\mathtt{M}_x \, E[\|\mathbf{\Delta h}\|^2]} \tag{3.336}$$

We note that if we indicate with $\hat{d}(k)$ the output of the identified system,

$$\hat{d}(k) = \sum_{i=0}^{N-1} \hat{h}_i \, x(k-i) \tag{3.337}$$

the fact that $\hat{\mathbf{h}} \neq \mathbf{h}$ causes $\hat{d}(k) \neq z(k)$, with an error given by

$$z(k) - \hat{d}(k) = \sum_{i=0}^{N-1} (h_i - \hat{h}_i) \, x(k-i) \tag{3.338}$$

having variance $\mathtt{M}_x \, E[\|\mathbf{\Delta h}\|^2]$ for a white noise input. As a consequence, (3.336) measures the ratio between the variance of the additive noise of the observed system and the variance of the error at the output of the identified system. From (3.338) we note that the difference

$$d(k) - \hat{d}(k) = (z(k) - \hat{d}(k)) + w(k) \tag{3.339}$$

consists of two terms, one due to the estimation error and the other due to the noise of the system.

3.B.2 Methods in the frequency domain

System identification in the absence of noise

In the absence of noise ($w = 0$), the output signal of the unknown system, represented in Figure 3.42, is

$$z(k) = \sum_{n=0}^{L-1} x(k-n)_{mod\, L} \, h_n \tag{3.340}$$

where a PN sequence of period $L = N$, equal to the length of the impulse response to be estimated, is assumed as input signal x. Let us consider the vector $\mathbf{z}^T = [z(k),$

$z(k+1), \ldots, z(k+(L-1))]$, and a circulant matrix $\mathbf{M}$ whose first row is $[x(k)_{mod\,L}$, $x(k-1)_{mod\,L}, \ldots, x(k-(L-1))_{mod\,L}]$. After an initial transient of $L-1$ samples, using the output samples $\{z(L-1), \ldots, z(2(L-1))\}$ we obtain a system of L linear equations in L unknowns, which in matrix notation can be written as

$$\mathbf{z} = \mathbf{M}\mathbf{h} \tag{3.341}$$

assuming $k = L-1$ in the definition of $\mathbf{z}$ and $\mathbf{M}$. The system of equations (3.341) admits a unique solution if and only if the matrix $\mathbf{M}$ is non-singular. Because the input sequence is periodic, the system (3.341) can be solved very efficiently, from a computational complexity point of view, in the frequency domain, rather than inverting the matrix $\mathbf{M}$. Being $\mathbf{M}$ circulant, the product in (3.341) can be substituted by the circular convolution (see (1.105))

$$z(k) = x \overset{L}{\otimes} h(k) \qquad k = L-1, \ldots, 2(L-1) \tag{3.342}$$

Letting $\mathcal{Z}_m = \mathrm{DFT}[z(k)]$, $\mathcal{X}_m = \mathrm{DFT}[x(k)]$, and $\mathcal{H}_m = \mathrm{DFT}[h_k]$, (3.342) can be rewritten in terms of the discrete Fourier transforms as

$$\mathcal{Z}_m = \mathcal{X}_m \mathcal{H}_m \qquad m = 0, \ldots, L-1 \tag{3.343}$$

from which we get

$$h_k = \mathrm{DFT}^{-1}\left[\frac{\mathcal{Z}_m}{\mathcal{X}_m}\right] \qquad k = 0, \ldots, L-1 \tag{3.344}$$

or, setting $s(k) = \mathrm{DFT}^{-1}[1/\mathcal{X}_m]$,

$$h_k = s \overset{L}{\otimes} z(k) \tag{3.345}$$

System identification in the presence of noise

Substituting in (3.345) the expression of the output signal obtained in the presence of noise, $d(k) = z(k) + w(k)$, for $k = L-1, \ldots, 2(L-1)$, the estimate of the coefficients of the unknown system is given by

$$\hat{h}_k = s \overset{L}{\otimes} d(k) = s \overset{L}{\otimes} z(k) + s \overset{L}{\otimes} w(k) = h_k + s \overset{L}{\otimes} w(k) \tag{3.346}$$

Assuming that w is zero-mean white noise with power σ_w^2, mean and variance of the estimate (3.346) are obtained as follows.

1. *Mean*

$$E[\hat{\mathbf{h}}] = \mathbf{h} \tag{3.347}$$

2. *Variance*

$$E[\|\boldsymbol{\Delta}\mathbf{h}\|^2] = E\left[\sum_{k=0}^{L-1} |\hat{h}_k - h_k|^2\right] = L\,E[|\hat{h}_k - h_k|^2] = L\,\sigma_w^2 \sum_{i=0}^{L-1} |s(i)|^2 \tag{3.348}$$

Using the Parseval theorem

$$\sum_{i=0}^{L-1} |s(i)|^2 = \frac{1}{L}\sum_{j=0}^{L-1} |\mathcal{S}_j|^2 = \frac{1}{L}\sum_{j=0}^{L-1} \frac{1}{|\mathcal{X}_j|^2} \tag{3.349}$$

it is possible to particularize (3.336) for PN *maximal-length* and CAZAC sequences. In the first case, from (3.303), it turns out

$$\begin{aligned} \mathcal{X}_0 &= 1 \\ |\mathcal{X}_1|^2 = |\mathcal{X}_2|^2 &= \cdots = |\mathcal{X}_{L-1}|^2 = L+1 \end{aligned} \tag{3.350}$$

hence, observing (3.348), (3.336) becomes

$$\Lambda_n = \frac{L+1}{2L} \tag{3.351}$$

For CAZAC sequences, from (3.308), we have that all terms $|\mathcal{X}_j|^2$ are equal

$$|\mathcal{X}_j|^2 = L \qquad j = 0, 1, \ldots, L-1 \tag{3.352}$$

and the minimum of (3.348) is equal to σ_w^2, therefore $\Lambda_n = 1$. In other words, if L is large, CAZAC sequences yield 3 dB improvement with respect to the maximal-length sequences. Although this method is very simple, it has the disadvantage that, in the best case, it gives an estimate with variance equal to the noise variance of the original system.

3.B.3 The LS method

With reference to the system of Figure 3.42, letting $\mathbf{x}^T(k) = [x(k), x(k-1), \ldots, x(k-(N-1))]$, the noisy output of the unknown system can be written as

$$d(k) = \mathbf{h}^T\mathbf{x}(k) + w(k) \qquad k = (N-1), \ldots, (N-1)+(L-1) \tag{3.353}$$

From (3.353) we see that the observation of L samples of the received signal requires the transmission of $L_{TS} = L + N - 1$ symbols of the training sequence $\{x(0), x(1), \ldots, x((N-1)+(L-1))\}$. The unknown system can be identified using the LS criterion [43, 44, 45]. For a certain estimate $\hat{\mathbf{h}}$ of the unknown system, the sum of squared errors at the output is given by

$$\mathcal{E} = \sum_{k=N-1}^{N-1+L-1} |d(k) - \hat{d}(k)|^2 \tag{3.354}$$

where, from (3.337),

$$\hat{d}(k) = \hat{\mathbf{h}}^T\mathbf{x}(k) \tag{3.355}$$

As for the analysis of Section 2.3, we introduce the following quantities.

1. *Energy of the desired signal*

$$\mathcal{E}_d = \sum_{k=N-1}^{N-1+L-1} |d(k)|^2 \tag{3.356}$$

2. *Correlation matrix of the input signal*

$$\mathbf{\Phi} = [\Phi(i,n)] \qquad i,n = 0,\ldots,N-1 \tag{3.357}$$

where

$$\Phi(i,n) = \sum_{k=N-1}^{N-1+L-1} x^*(k-i)\,x(k-n) \tag{3.358}$$

3. *Cross-correlation vector*

$$\boldsymbol{\vartheta}^T = [\vartheta(0),\ldots,\vartheta(N-1)] \tag{3.359}$$

where

$$\vartheta(n) = \sum_{k=N-1}^{N-1+L-1} d(k)\,x^*(k-n) \tag{3.360}$$

Then the cost function (3.354) becomes

$$\mathcal{E} = \mathcal{E}_d - \hat{\mathbf{h}}^H\boldsymbol{\vartheta} - \boldsymbol{\vartheta}^H\hat{\mathbf{h}} + \hat{\mathbf{h}}^H\mathbf{\Phi}\hat{\mathbf{h}} \tag{3.361}$$

As the matrix $\mathbf{\Phi}$ is determined by a suitably chosen training sequence, we can assume that $\mathbf{\Phi}$ is positive definite and therefore the inverse exists. The solution to the LS problem yields

$$\hat{\mathbf{h}}_{ls} = \mathbf{\Phi}^{-1}\boldsymbol{\vartheta} \tag{3.362}$$

with a corresponding error equal to

$$\mathcal{E}_{min} = \mathcal{E}_d - \boldsymbol{\vartheta}^H\hat{\mathbf{h}}_{ls} \tag{3.363}$$

We observe that the matrix $\mathbf{\Phi}^{-1}$ in the (3.362) can be pre-computed and memorized, because it depends only on the training sequence. In some applications it is useful to estimate the variance of the noise signal w that, observing (3.339), for $\hat{\mathbf{h}} \simeq \mathbf{h}$ can be assumed equal to

$$(\hat{\sigma}_w^2) = \frac{1}{L}\,\mathcal{E}_{min} \tag{3.364}$$

Formulation using the data matrix

From the general analysis given on page 152, we recall the following definitions.

1. *$L \times N$ observation matrix*

$$\boldsymbol{\mathcal{I}} = \begin{bmatrix} x(N-1) & \dots & x(0) \\ \vdots & \ddots & \vdots \\ x((N-1)+(L-1)) & \dots & x(L-1) \end{bmatrix} \tag{3.365}$$

2. *Desired sample vector*

$$\boldsymbol{o}^T = [d(N-1), \dots, d((N-1)+(L-1))] \tag{3.366}$$

where $d(k)$ is given by (3.353).

Observing (2.139), (2.131), and (2.160), we have

$$\boldsymbol{\Phi} = \boldsymbol{\mathcal{I}}^H \boldsymbol{\mathcal{I}} \qquad \boldsymbol{\vartheta} = \boldsymbol{\mathcal{I}}^H \boldsymbol{o} \tag{3.367}$$

and

$$\hat{\mathbf{h}}_{ls} = (\boldsymbol{\mathcal{I}}^H \boldsymbol{\mathcal{I}})^{-1} \boldsymbol{\mathcal{I}}^H \boldsymbol{o} \tag{3.368}$$

which coincides with (3.362).

We note the introduction of the new symbols $\boldsymbol{\mathcal{I}}$ and $\boldsymbol{o}$, in relation to an alternative LMMSE estimation method, which will be given in Section 3.B.4.

Computation of the signal-to-estimation error ratio

We now evaluate the performance of the LS method for the estimation of $\mathbf{h}$. From (3.360), (3.359) can be rewritten as

$$\boldsymbol{\vartheta} = \sum_{k=N-1}^{N-1+L-1} d(k)\, \mathbf{x}^*(k) \tag{3.369}$$

Substituting (3.353) in (3.369), and letting

$$\boldsymbol{\xi} = \sum_{k=N-1}^{N-1+L-1} w(k)\, \mathbf{x}^*(k) \tag{3.370}$$

observing (3.357), we obtain the relation

$$\boldsymbol{\vartheta} = \boldsymbol{\Phi}\mathbf{h} + \boldsymbol{\xi} \tag{3.371}$$

Consequently, substituting (3.371) in (3.362), the estimation error vector can be expressed as

$$\boldsymbol{\Delta}\mathbf{h} = \boldsymbol{\Phi}^{-1}\boldsymbol{\xi} \tag{3.372}$$

If w is zero-mean white noise with variance σ_w^2, $\boldsymbol{\xi}^*$ is a zero-mean random vector with correlation matrix

$$\mathbf{R}_{\boldsymbol{\xi}} = E[\boldsymbol{\xi}^* \boldsymbol{\xi}^T] = \sigma_w^2 \boldsymbol{\Phi}^* \tag{3.373}$$

Therefore, $\boldsymbol{\Delta h}$ has mean zero and correlation matrix

$$\mathbf{R}_{\boldsymbol{\Delta h}} = \sigma_w^2 (\boldsymbol{\Phi}^*)^{-1} \tag{3.374}$$

In particular,

$$E[\|\boldsymbol{\Delta h}\|^2] = \sigma_w^2 \operatorname{tr}[(\boldsymbol{\Phi}^*)^{-1}] \tag{3.375}$$

and, from (3.336), we get

$$\Lambda_n = (\operatorname{tr}[\boldsymbol{\Phi}^{-1}])^{-1} \tag{3.376}$$

Using as training sequence a CAZAC sequence, the matrix $\boldsymbol{\Phi}$ is diagonal,

$$\boldsymbol{\Phi} = L\,\mathbf{I} \tag{3.377}$$

where $\mathbf{I}$ is the $N \times N$ identity matrix. The elements on the diagonal of $\boldsymbol{\Phi}^{-1}$ are equal to $1/L$, and (3.376) yields

$$\Lambda_n = \frac{L}{N} \tag{3.378}$$

The (3.378) gives a good indication of the relation between the number of observations L, the number of system coefficients N, and Λ_n. For example, doubling the length of the training sequence, Λ_n also doubles. Now, using as training sequence a maximal-length sequence of periodicity L, and indicating with $\mathbf{1}_{N\times N}$ the matrix with all elements equal to 1, the correlation matrix $\boldsymbol{\Phi}$ can be written as

$$\boldsymbol{\Phi} = (L+1)\mathbf{I} - \mathbf{1}_{N\times N} \tag{3.379}$$

From (3.379) the inverse is given by

$$\boldsymbol{\Phi}^{-1} = \frac{1}{L+1}\left(\mathbf{I} + \frac{\mathbf{1}_{N\times N}}{L+1-N}\right) \tag{3.380}$$

which, substituted in (3.376), yields

$$\Lambda_n = \frac{(L+1)(L+1-N)}{N(L+2-N)} \tag{3.381}$$

In Figure 3.44 the behavior of Λ_n is represented as a function of N, for CAZAC sequences (solid line) and for maximal-length sequences (dotted-dashed line), with parameter L. We make the following observations.

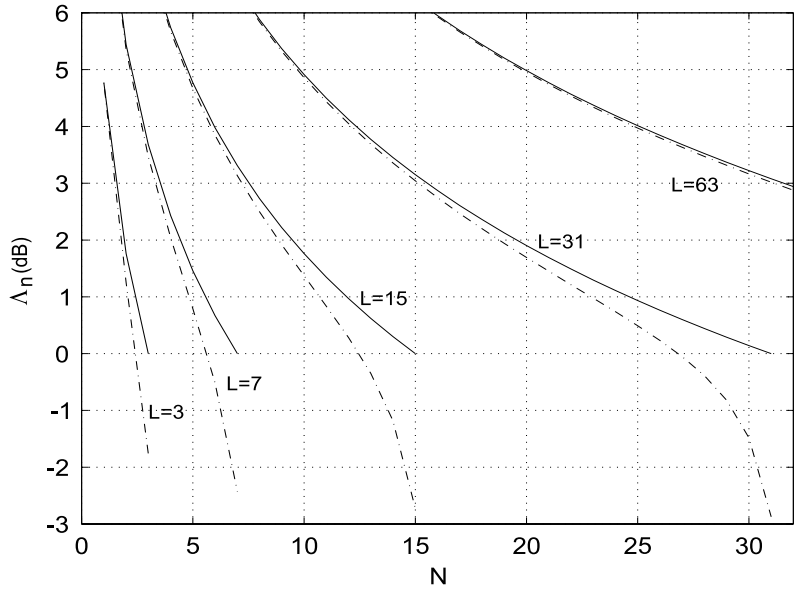


Figure 3.44. Λ_n vs. N for CAZAC sequences (solid line) and maximal-length sequences (dotted-dashed line), for various values of L.

- For a given N, choosing $L \gg N$, the two sequences yield approximately the same Λ_n. The worst case is obtained for $L = N$; for example, for $L = 15$ the maximal-length sequence yields a value of Λ_n that is about 3 dB lower than the upper bound (3.378). We note that the frequency method operates for $L = N$.

- For a given value of L, because of the presence of the noise w, the estimate of the coefficients becomes worse if the number of coefficients N is larger than the number of coefficients of the system N_h. On the other hand, if N is smaller than N_h, the estimation error may assume large values (see (3.270)).

- For sparse systems, where the number of coefficients may be large, but only a few of them are non-zero, the estimate is usually very noisy. Therefore, after obtaining the estimate, it is necessary to set to zero all coefficients whose amplitude is below a certain threshold.

- If the correlation method (3.331) is adopted, we get

$$\hat{\mathbf{h}} = \frac{1}{L}\,\boldsymbol{\vartheta}$$

where $\boldsymbol{\vartheta}$ is given by (3.359). Observing (3.371), we get

$$\boldsymbol{\Delta}\mathbf{h} = \left(\frac{1}{L}\,\boldsymbol{\Phi} - \mathbf{I}\right)\mathbf{h} + \frac{1}{L}\,\boldsymbol{\xi} \tag{3.382}$$

Consequently the estimate is affected by a BIAS term equal to $((1/L)\,\mathbf{\Phi} - \mathbf{I})\mathbf{h}$, and has a covariance matrix equal to $(1/L^2)\,\mathbf{R}_{\boldsymbol{\xi}}$. In particular, using (3.373), it turns out

$$E[\|\mathbf{\Delta h}\|^2] = \left\| \left(\frac{1}{L}\,\mathbf{\Phi} - \mathbf{I} \right) \mathbf{h} \right\|^2 + \frac{\sigma_w^2}{L^2}\,\mathrm{tr}[\mathbf{\Phi}] \tag{3.383}$$

and

$$\Lambda_n = \frac{1}{\dfrac{1}{L^2}\,\mathrm{tr}[\mathbf{\Phi}] + \left\| \left(\dfrac{1}{L}\,\mathbf{\Phi} - \mathbf{I} \right) \mathbf{h} \right\|^2 \dfrac{1}{\sigma_w^2}} \tag{3.384}$$

Using a CAZAC sequence, from (3.377) the second term of the denominator in (3.384) vanishes, and Λ_n is given by (3.378). In fact, for a CAZAC sequence, as (3.324) is strictly true and $\mathbf{\Phi}^{-1}$ is diagonal, the LS method (3.362) coincides with the correlation method (3.331).

Using instead a maximal-length sequence, from (3.379) we get

$$\left\| \left(\frac{1}{L}\,\mathbf{\Phi} - \mathbf{I} \right) \mathbf{h} \right\|^2 = \frac{1}{L^2}\,\|(\mathbf{1} - \mathbf{I})\,\mathbf{h}\|^2 = \frac{1}{L^2} \sum_{i=0}^{N-1} |h_i - \mathcal{H}(0)|^2 \tag{3.385}$$

$$= \frac{1}{L^2}(\|\mathbf{h}\|^2 + (N-2)\,|\mathcal{H}(0)|^2)$$

where $\mathcal{H}(0) = \sum_{i=0}^{N-1} h_i$. Moreover, we have

$$\mathrm{tr}[\mathbf{\Phi}] = NL \tag{3.386}$$

hence

$$\Lambda_n = \frac{L}{N + \dfrac{1}{L}\left[\Lambda + (N-2)\,\dfrac{|\mathcal{H}(0)|^2}{\sigma_w^2} \right]} \tag{3.387}$$

where Λ is defined in (3.335).

We observe that using the correlation method, we obtain the same values Λ_n (3.381) as the LS method, if L is large enough to satisfy the condition

$$\Lambda + (N-2)\,\frac{|\mathcal{H}(0)|^2}{\sigma_w^2} < L$$

3.B.4 The LMMSE method

We refer to the system model of Figure 3.42. Let us assume that w and h are statistically independent random processes, whose second-order statistic is known.

For a known input sequence x, we desire to estimate h using the LMMSE method given in the Appendix 2.A, from the observation of the noisy output sequence d, which now will be denoted as o.

We note that in the LS method the observation was the transmitted signal x, while the desired signal was given by the system noisy output d. The observation is now given by d and the desired signal is the system impulse response h. Consequently, some caution is needed to apply (2.229) to the problem under investigation. Recalling the definition (3.366) of the observation vector $\boldsymbol{o}$, and (2.229), the LMMSE estimator is given by

$$\hat{\mathbf{h}}_{LMMSE} = (\mathbf{R}_{\boldsymbol{o}}^{-1}\mathbf{R}_{\boldsymbol{o}\mathbf{h}})^T\boldsymbol{o}\,, \tag{3.388}$$

where we have assumed $E[\boldsymbol{o}] = \mathbf{0}$ and $E[\mathbf{h}] = \mathbf{0}$. Consequently, (3.388) provides an estimate only of the random (i.e. the non-deterministic) component of the channel impulse response.

Now, from (3.353), letting

$$\mathbf{w}^T = [w(N-1), \ldots, w((N-1)+(L-1))] \tag{3.389}$$

be a random vector with noise components, we can write

$$\boldsymbol{o} = \boldsymbol{\mathcal{I}}\mathbf{h} + \mathbf{w} \tag{3.390}$$

Assuming that the sequence $\{x(k)\}$ is known, we have

$$\mathbf{R}_{\boldsymbol{o}} = E[\boldsymbol{o}^*\boldsymbol{o}^T] = \boldsymbol{\mathcal{I}}^*\mathbf{R}_{\mathbf{h}}\boldsymbol{\mathcal{I}}^T + \mathbf{R}_{\mathbf{w}} \tag{3.391}$$

and

$$\mathbf{R}_{\boldsymbol{o}\mathbf{h}} = E[\boldsymbol{o}^*\mathbf{h}^T] = \boldsymbol{\mathcal{I}}^*\mathbf{R}_{\mathbf{h}} \tag{3.392}$$

Then (3.388) becomes

$$\hat{\mathbf{h}}_{LMMSE} = [(\boldsymbol{\mathcal{I}}^*\mathbf{R}_{\mathbf{h}}\boldsymbol{\mathcal{I}}^T + \mathbf{R}_{\mathbf{w}})^{-1}\boldsymbol{\mathcal{I}}^*\mathbf{R}_{\mathbf{h}}]^T\boldsymbol{o} \tag{3.393}$$

Using the matrix inversion lemma (3.176), (3.393) can be rewritten as

$$\hat{\mathbf{h}}_{LMMSE} = [(\mathbf{R}_{\mathbf{h}}^*)^{-1} + \boldsymbol{\mathcal{I}}^H(\mathbf{R}_{\mathbf{w}}^*)^{-1}\boldsymbol{\mathcal{I}}]^{-1}\boldsymbol{\mathcal{I}}^H(\mathbf{R}_{\mathbf{w}}^*)^{-1}\boldsymbol{o} \tag{3.394}$$

If $\mathbf{R}_{\mathbf{w}} = \sigma_w^2\mathbf{I}$, we have

$$\hat{\mathbf{h}}_{LMMSE} = [\sigma_w^2(\mathbf{R}_{\mathbf{h}}^*)^{-1} + \boldsymbol{\mathcal{I}}^H\boldsymbol{\mathcal{I}}]^{-1}\boldsymbol{\mathcal{I}}^H\boldsymbol{o} \tag{3.395}$$

We note that with respect to the LS method (3.368), the LMMSE method (3.395) introduces a weighting of the components given by $\boldsymbol{\vartheta} = \boldsymbol{\mathcal{I}}^H\boldsymbol{o}$, which depends on the ratio between the noise variance and the variance of $\mathbf{h}$. If the variance of the components of $\mathbf{h}$ is large, then $\mathbf{R}_{\mathbf{h}}$ is also large and likely $\mathbf{R}_{\mathbf{h}}^{-1}$ can be neglected in (3.395).

We conclude by recalling that $\mathbf{R}_{\mathbf{h}}$ is diagonal for a WSSUS radio channel model (see (4.221)), and the components of $\mathbf{h}$ are derived by the *power delay profile*.

For an analysis of the estimation error we can refer to (2.233), which uses the error vector $\mathbf{\Delta h} = \hat{\mathbf{h}}_{LMMSE} - \mathbf{h}$ having a correlation matrix

$$\mathbf{R}_{\Delta\mathbf{h}} = \{[(\mathbf{R}_{\mathbf{h}}^*)^{-1} + \boldsymbol{\mathcal{I}}^H(\mathbf{R}_{\mathbf{w}}^*)^{-1}\boldsymbol{\mathcal{I}}]^*\}^{-1} \tag{3.396}$$

If $\mathbf{R}_{\mathbf{w}} = \sigma_w^2\mathbf{I}$, we get

$$\mathbf{R}_{\Delta\mathbf{h}} = \sigma_w^2\{[\sigma_w^2(\mathbf{R}_{\mathbf{h}}^*)^{-1} + \boldsymbol{\mathcal{I}}^H\boldsymbol{\mathcal{I}}]^*\}^{-1} \tag{3.397}$$

Moreover, in general

$$E[\|\mathbf{\Delta h}\|^2] = \mathrm{tr}[\mathbf{R}_{\Delta\mathbf{h}}] \tag{3.398}$$

This result can be compared with that of the LS method given by (3.375).

3.B.5 Identification of a continuous-time system

In the case of continuous-time systems, the scheme of Figure 3.42 can be modified to that of Figure 3.45 [46], where the noise is neglected. A PN sequence of period L, repeated several times, is used to modulate in amplitude the pulse

$$g(t) = \mathrm{w}_{T_c}(t) = \mathrm{rect}\left(\frac{t - T_c/2}{T_c}\right) \tag{3.399}$$

The modulated output signal x is therefore given by

$$x(t) = \sum_{i=0}^{+\infty} p(i)_{mod\,L}\, g(t - iT_c) \tag{3.400}$$

The autocorrelation of x, periodic function of period LT_c, is expressed by

$$\mathrm{r}_x(t) = \frac{1}{LT_c}\int_{-\frac{LT_c}{2}}^{+\frac{LT_c}{2}} x(\eta)x^*(\eta - t)d\eta \tag{3.401}$$

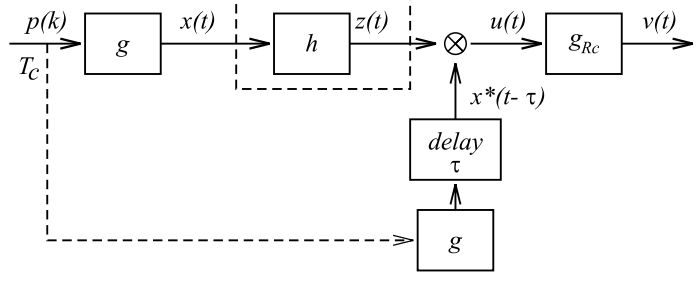


Figure 3.45. Basic scheme to measure the impulse response of an unknown system.

As g has finite support of length T_c, $\mathtt{r}_x(t)$ has a simple expression given by

$$\mathtt{r}_x(t) = \frac{1}{T_c} \sum_{\ell=0}^{L-1} \mathtt{r}_p(\ell) \mathtt{r}_g(t - \ell T_c) \qquad 0 \le t \le LT_c \tag{3.402}$$

where, in the case of $g(t)$ given by (3.399), we have

$$\mathtt{r}_g(t) = \int_0^{T_c} g(\eta) g(\eta - t) d\eta = T_c \left(1 - \frac{|t|}{T_c}\right) \text{rect}\left(\frac{t}{2T_c}\right) \tag{3.403}$$

Substituting (3.403) in (3.402) and assuming a maximal-length PN sequence, with $\mathtt{r}_p(0) = 1$ and $\mathtt{r}_p(\ell) = -1/L$ for $\ell = 1, \ldots, L-1$, we obtain

$$\mathtt{r}_x(t) = -\frac{1}{L} + \left(1 + \frac{1}{L}\right)\left(1 - \frac{|t|}{T_c}\right) \text{rect}\left(\frac{t}{2T_c}\right) \qquad |t| \le \frac{LT_c}{2} \tag{3.404}$$

as shown in Figure 3.46 for $L = 8$. If the output z of the unknown system to be identified is multiplied by a delayed version of the input, $x^*(t - \tau)$, and the result is filtered by an ideal integrator between 0 and LT_c with impulse response

$$g_{Rc}(t) = \frac{1}{LT_c} \mathtt{w}_{LT_c}(t) = \frac{1}{LT_c} \text{rect}\left(\frac{t - LT_c/2}{LT_c}\right) \tag{3.405}$$

we obtain

$$\begin{aligned} v(t) &= \frac{1}{LT_c} \int_{t-LT_c}^{t} u(\eta) d\eta \\ &= \frac{1}{LT_c} \int_{t-LT_c}^{t} \int_0^{+\infty} [h(\xi) x(\eta - \xi) d\xi] x^*(\eta - \tau) d\eta \\ &= \int_0^{+\infty} h(\xi) \mathtt{r}_x(\tau - \xi) d\xi = h^* \mathtt{r}_x(\tau) = v_\tau \end{aligned} \tag{3.406}$$

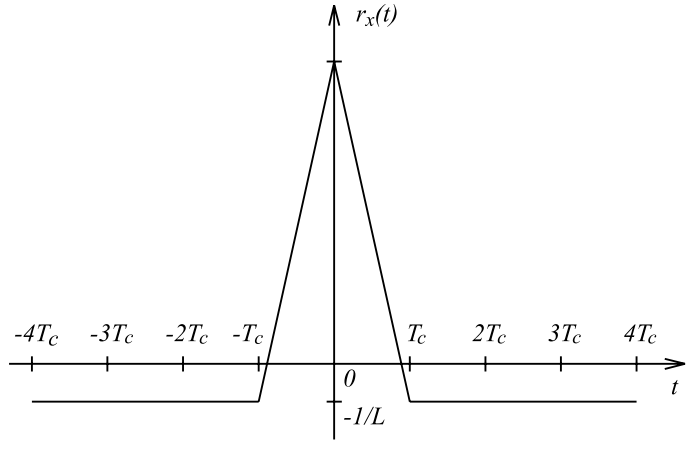


Figure 3.46. Autocorrelation function of $x(t)$.

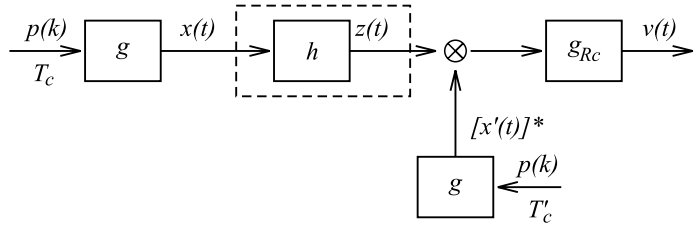


Figure 3.47. Sliding window method to measure the impulse response of an unknown system.

Therefore the output assumes a constant value v_τ equal to the convolution between the unknown system h and the autocorrelation of x evaluated in τ. Assuming $1/T_c$ is larger than the maximum frequency of the spectral components of h, and L is sufficiently large, the output v_τ is approximately proportional to $h(\tau)$. The scheme represented in Figure 3.47 is an alternative to that of Figure 3.45, of simpler implementation because it does not require synchronization of the two PN sequences at transmitter and receiver. In this latter scheme, the output z of the unknown system is multiplied by a PN sequence having the same characteristics of the transmitted sequence, but a different clock frequency $f'_0 = 1/T'_c$, related to the clock frequency $f_0 = 1/T_c$ of the transmitter by the relation

$$f'_0 = f_0 \left(1 - \frac{1}{K}\right) \tag{3.407}$$

where K is a parameter of the system. We consider the function

$$\mathtt{r}_{x'x}(\tau) = \frac{1}{LT_c} \int_0^{LT_c} [x'(\eta)]^* x(\eta - \tau)\, d\eta \tag{3.408}$$

where τ is the delay at time $t = 0$ between the two sequences. As time elapses, the delay between the two sequence diminishes of the quantity $(t/T'_c)(T'_c - T_c) = t/K$, so that

$$\frac{1}{LT_c} \int_{t-LT_c}^{t} [x'(\eta)]^* x(\eta - \tau)\, d\eta \simeq \mathtt{r}_{x'x}\left(\tau - \frac{t - LT_c}{K}\right) \qquad \text{for } t \geq LT_c \tag{3.409}$$

If K is sufficiently large, we can assume that

$$\mathtt{r}_{x'x}(\tau) \simeq \mathtt{r}_x(\tau) \tag{3.410}$$

At the output of the filter g_{Rc}, given by (3.405), therefore we have

$$\begin{aligned} v(t) &= \frac{1}{LT_c} \int_{t-LT_c}^{t} \int_0^{+\infty} [h(\xi) x(\eta - \xi)\, d\xi][x'(\eta)]^*\, d\eta \\ &= \int_0^{+\infty} h(\xi) \mathtt{r}_{x'x}\left(\xi - \frac{t - LT_c}{K}\right) d\xi \\ &\simeq \int_0^{+\infty} h(\xi) \mathtt{r}_x\left(\frac{t - LT_c}{K} - \xi\right) d\xi \end{aligned} \tag{3.411}$$

or, with the substitution $t' = (t - LT_c)/K$,

$$y(Kt' + LT_c) \simeq \int_0^{+\infty} h(\xi) r_x(t' - \xi)\, d\xi \tag{3.412}$$

where the integral in (3.412) coincides with the integral in (3.406). If K is sufficiently large (an increase of K clearly requires a greater precision and hence a greater cost of the frequency synthesizer to generate f_0'), it can be shown that the approximations in (3.409) and (3.410) are valid. Therefore the systems in Figure 3.47 and in Figure 3.45 are equivalent.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 4

Transmission media

The first two sections of this chapter introduce several parameters that are associated with the electrical characteristics of electronic devices. The fundamental properties of various transmission media will be discussed in the remaining sections.

4.1 Electrical characterization of a transmission system

Simplified scheme of a transmission system

We consider a *message source*, which could be for example a machine that generates speech signals and/or sequences of symbols. To be able to convey the information represented by these messages to a *user* situated at a certain distance from the source, a transmission system can be configured as illustrated in Figure 4.1. The *transmission medium* may consist, e.g., of one or more of the following media: twisted-pair cable, coaxial cable, radio link, waveguide, or optical fiber. The *transmitter* is a device that converts the source message into a signal that can physically propagate through the transmission medium. The task of the *receiver* is to yield an accurate replica of the original message. An *intermediate device* may compensate for attenuation and/or disturbances introduced by the medium. It may consist of a simple amplifier or, more generally, of a repeater, or even—for data signals – of a *regenerator* that attempts to restore the original source message. The word *channel* is often used in practice to indicate in abstract terms a transmission medium that allows the propagation of a signal.

From an electrical point of view, many components (e.g., cables, amplifiers, filters, ...) of a transmission system may be interpreted as a cascade of *2-port linear networks*. A 2-port network is a device that transfers a signal v_i from a *source* to a *load*, as shown in Figure 4.2a, where Z_i denotes the source impedance, Z_L the load impedance, and v_1 and v_L are, respectively, the input and output signals of the 2-port network, that are expressed as voltage signals. If Z_1 and Z_2 are respectively the input and output impedances of the 2-port network and v_o is the open-circuit voltage at the output, we obtain the equivalent electrical scheme of Figure 4.2b, where we identify input and output two-terminal devices. The corresponding abstract model is illustrated in Figure 4.2c, where the signal v_i produces v_1 through the source voltage divider, v_1 gives origin to v_o according to the 2-port network transfer characteristics, and the signal v_L is obtained from v_o through the load voltage divider.

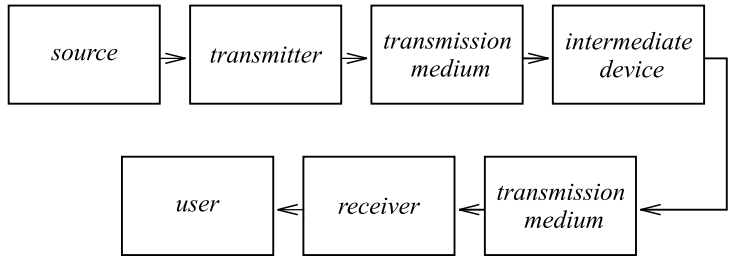
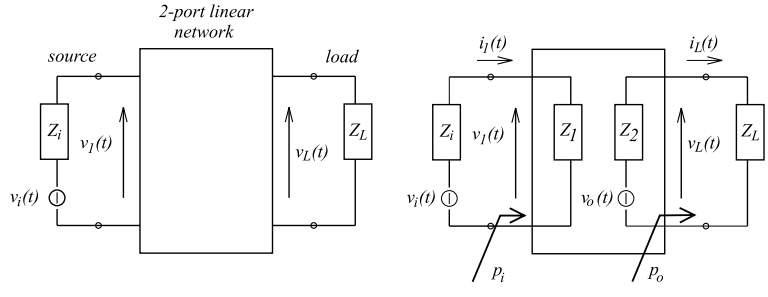
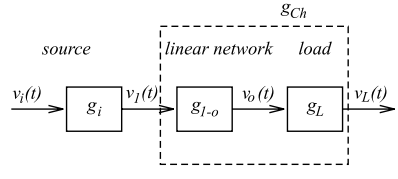


Figure 4.1. Simplified scheme of a transmission system.



(a) Connection of a source to a load through a 2-port linear network

(b) Equivalent electrical circuit



(c) Equivalent model

Figure 4.2. Connection of a source to a load through a 2-port linear network.

In the cascade of several 2-port networks, the frequency response $\mathcal{G}_{Ch}(f)$ of each network is given by the ratio V_L/V_1. Therefore, with reference to Figure 4.2c, we have

$$\mathcal{G}_{Ch}(f) = \mathcal{G}_{1-o}(f)\mathcal{G}_L(f) \tag{4.1}$$

where $\mathcal{G}_{1-o}(f)$ and $\mathcal{G}_L(f)$ denote the frequency responses of the 2-port network and of the load, respectively. We note that in some cases the frequency response could be defined as V_L/I_1, I_L/V_1, or I_L/I_1. For these cases, the expression of $\mathcal{G}_{Ch}(f)$ will be different from (4.1).

To analyze the characteristics of the network of Figure 4.2a, however, we will refer to the study of two-terminal devices.

Characterization of an active device

We consider the two-terminal device of Figure 4.3 that consists of a generator with an open-circuit voltage v_b, modelled as a random WSS process with *statistical power spectral density* $\mathcal{P}_{v_b}$ (V^2/Hz), and an impedance Z_b. The device is connected to a load with impedance Z_c.

If v and i are, respectively, the voltage and the current at the load, the *average power* (W) transferred to the load is defined by the relation:[1]

$$\mathtt{P} = \lim_{t\to\infty} \frac{1}{t} \int_{-t/2}^{t/2} v(\tau)\, i(\tau)\, d\tau = E[v(t)\, i(t)] \tag{4.3}$$

assuming that $v(t)i(t)$ is an ergodic process in mean (see (1.442)). If $\mathcal{P}_{vi}$ is the cross-spectral density between v and i, we have that:

$$\mathtt{P} = \int_{-\infty}^{+\infty} \mathcal{P}_{vi}(f)\, df = \int_{-\infty}^{+\infty} Re[\mathcal{P}_{vi}(f)]\, df \tag{4.4}$$

In fact, from (1.230), the cross-correlation $\mathtt{r}_{vi}$ is a real function. Hence $\mathcal{P}_{vi}$ is Hermitian with even real part and odd imaginary part.

Definition 4.1
The function

$$\mathtt{p}(f) = Re[\mathcal{P}_{vi}(f)] \text{ (W/Hz)} \tag{4.5}$$

is called *average power density* transferred to the load and expresses the average power per unit of frequency.

We now obtain $\mathcal{P}_{vi}$ in terms of $\mathcal{P}_{v_b}$ using the method shown on page 49. Being

$$I = V_b \frac{1}{Z_b + Z_c} \qquad V = V_b \frac{Z_c}{Z_b + Z_c} \tag{4.6}$$

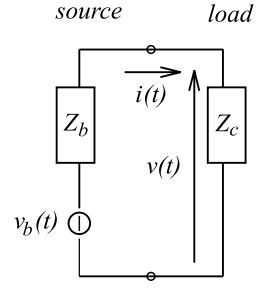


Figure 4.3. Active two-terminal device (voltage generator) connected to a load.

[1] In propagation theory [1], P is sometimes defined as

$$\mathtt{P} = \frac{1}{2} \lim_{t\to\infty} \frac{1}{t} \int_{-t/2}^{t/2} v(\tau)\, i(\tau)\, d\tau \tag{4.2}$$

then

$$VI^* = V_b V_b^* \frac{Z_c}{|Z_b + Z_c|^2} \tag{4.7}$$

hence

$$\mathcal{P}_{vi}(f) = \mathcal{P}_{v_b}(f) \frac{Z_c}{|Z_b + Z_c|^2} \tag{4.8}$$

and

$$\mathrm{p}(f) = \mathcal{P}_{v_b}(f) \frac{R_c}{|Z_b + Z_c|^2} \qquad R_c = Re[Z_c] \tag{4.9}$$

In general, if v is the voltage at the load impedance Z_c, the following relation holds

$$\mathrm{p}(f) = \mathcal{P}_v(f) \frac{R_c}{|Z_c|^2} \tag{4.10}$$

Definition 4.2
The *available power* per unit of frequency of an active two-terminal device is defined as the maximum of (4.9) with respect to Z_c and is obtained for $Z_c = Z_b^*$:

$$\mathrm{p}_d(f) = \frac{\mathcal{P}_{v_b}(f)}{4R_b} \qquad R_b = Re[Z_b] \tag{4.11}$$

We note that (4.11) is a parameter of the active device that expresses the maximum power per unit of frequency that can be delivered to the load.

Active two-terminal device as a current generator. For the circuit of Figure 4.4, that consists of a current source with admittance $Y_b = G_b + jB_b$ and a load with $Y_c = Y_b^*$, the available power per unit of frequency is given by

$$\mathrm{p}_d(f) = \frac{\mathcal{P}_{i_b}(f)}{4G_b} \tag{4.12}$$

where $G_b = Re[Y_b]$, and $\mathcal{P}_{i_b}$ (A^2/Hz) is the PSD of the signal i_b.

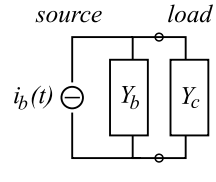


Figure 4.4. Active two-terminal device (current generator) connected to a load.

We have a simple relation between the circuit of Figure 4.4 and that of Figure 4.3 for $Z_b = R_b + jX_b$; from Norton theorem we get

$$Y_b = \frac{1}{Z_b} = \frac{R_b}{|Z_b|^2} - j\frac{X_b}{|Z_b|^2} \tag{4.13}$$

and

$$I_b = V_b \frac{1}{Z_b} \tag{4.14}$$

Conditions for the absence of signal distortion

We now return to the circuit of Figure 4.3 and consider the relation between the load voltage v and the source voltage v_b; in this case the frequency response is given by

$$\mathcal{G}(f) = \frac{V(f)}{V_b(f)} = \frac{Z_c(f)}{Z_c(f) + Z_b(f)} \qquad f \in \mathcal{B} \tag{4.15}$$

where $\mathcal{B}$ is the passband of v_b (see Definitions 1.10 on page 29 and 1.11 on page 46). According to Heaviside conditions (1.144), the voltage v_b is transferred to the load without distortion if

$$Z_b(f) = K_1 Z_c(f) \qquad f \in \mathcal{B} \tag{4.16}$$

where K_1 is a constant. In a connection between a source and a load, the conditions for the absence of signal distortion, as well as for maximum transfer of power to the load, are verified in the following two cases.

1. For a *broadband* signal v_b, regarding the impedances as *complex-valued functions* of the frequency within the passband $\mathcal{B}$, the only way to verify the conditions $Z_c(f) = Z_b^*(f)$ and (4.16) is that both the load impedance (R_c) and the source impedance (R_b) are purely resistive. Note that for $R_b = R_c$ also the condition for maximum transfer of power is satisfied.

2. For a *narrowband* signal v_b, for which the impedances are regarded as *complex-valued constants* within the passband $\mathcal{B}$, the condition for maximum transfer of power $Z_c = Z_b^*$ is easily verified. Distortion is not a problem because $\mathcal{G}(f)$ is a complex constant in the passband of v_b, and the phase term is equivalent to a delay.

Characterization of a 2-port network

With reference to the circuit of Figure 4.2, the frequency responses of the various blocks are given by

$$\mathcal{G}_i(f) = \frac{Z_1(f)}{Z_i(f) + Z_1(f)} \tag{4.17}$$

$$\mathcal{G}_L(f) = \frac{Z_L(f)}{Z_2(f) + Z_L(f)} \tag{4.18}$$

and

$$\mathcal{G}_{1-o}(f) = \frac{V_o(f)}{V_1(f)} \tag{4.19}$$

The conditions for the absence of distortion between v_i and v_L are verified if

$$\mathcal{G}(f) = \frac{V_L}{V_i} = \mathcal{G}_i(f)\mathcal{G}_{1-o}(f)\mathcal{G}_L(f) \tag{4.20}$$

is a constant in the passband of v_i. Note that also in this case the presence of a constant delay factor is possible.

Let $\mathrm{p}_i(f)$ and $\mathrm{p}_o(f)$ be the average power densities of source and load, respectively:

$$\mathrm{p}_i(f) = \mathcal{P}_{v_1}(f)\frac{R_1}{|Z_1|^2} = \mathcal{P}_{v_i}(f)\frac{R_1}{|Z_1 + Z_i|^2} \tag{4.21}$$

and

$$\mathrm{p}_o(f) = \mathcal{P}_{v_L}(f)\frac{R_L}{|Z_L|^2} = \mathcal{P}_{v_o}(f)\frac{R_L}{|Z_L + Z_2|^2} \tag{4.22}$$

Definition 4.3
The *network power gain* is defined as the ratio

$$\mathrm{g}(f) = \frac{\mathrm{p}_o(f)}{\mathrm{p}_i(f)} \tag{4.23}$$

Using the expressions of the various frequency responses, and observing (see Figure 4.2c)

$$\mathcal{P}_{v_L}(f) = |\mathcal{G}_{Ch}(f)|^2\mathcal{P}_{v_1}(f) \tag{4.24}$$

we get

$$\mathrm{g}(f) = \mathcal{P}_{v_L}(f)\frac{R_L}{|Z_L|^2}\frac{1}{\mathcal{P}_{v_1}(f)}\frac{|Z_1|^2}{R_1} = \frac{R_L}{R_1}\frac{|Z_1|^2}{|Z_L|^2}|\mathcal{G}_{Ch}(f)|^2 \tag{4.25}$$

In the presence of match for maximum transfer of power only at the source, we introduce the notion of *transducer gain*, defined as

$$\mathrm{g}_t(f) = \frac{\mathrm{p}_o(f)}{\mathrm{p}_{i,d}(f)} \tag{4.26}$$

where $\mathrm{p}_{i,d}(f)$ is the available power per unit of frequency at the source.

Definition 4.4
A 2-port network is said to be *perfectly matched* if the conditions for maximum transfer of power are established at the source as well as at the load:

$$Z_1 = Z_i^* \quad \text{and} \quad Z_L = Z_2^* \tag{4.27}$$

In this case the powers become *available powers*:

$$\bullet \textit{at the source} \qquad \mathtt{p}_{i,d}(f) = \mathcal{P}_{v_i}(f)\frac{1}{4R_i} \tag{4.28}$$

$$\bullet \textit{at the load} \qquad \mathtt{p}_{o,d}(f) = \mathcal{P}_{v_o}(f)\frac{1}{4R_2} \tag{4.29}$$

Definition 4.5
The *available power gain* is defined as the ratio between $\mathtt{p}_{o,d}(f)$ and $\mathtt{p}_{i,d}(f)$,

$$\mathtt{g}_d(f) = \frac{\mathtt{p}_{o,d}(f)}{\mathtt{p}_{i,d}(f)} \tag{4.30}$$

In particular for $Z_1 = Z_2$, that is when input and output network impedances coincide, from (4.25) we get

$$\mathtt{g}_d(f) = |\mathcal{G}_{Ch}(f)|^2 \tag{4.31}$$

If in the passband $\mathcal{B}$ of v_i we have $\mathtt{g}_d > 1$ the network is said to be *active*. If instead $\mathtt{g}_d < 1$ then the network is *passive*; in this case, we speak of *available attenuation of the network*:

$$\mathtt{a}_d = \frac{1}{\mathtt{g}_d} \tag{4.32}$$

In dB,

$$(\mathtt{g}_d)_{dB} = 10\log_{10}\mathtt{g}_d \tag{4.33}$$

and

$$(\mathtt{a}_d)_{dB} = -(\mathtt{g}_d)_{dB} \tag{4.34}$$

Definition 4.6
Apart from a possible delay, for an *ideal distortionless network* with power gain (attenuation) $\mathtt{g}_d$ ($\mathtt{a}_d$), we will assume that the frequency response of the network is

$$\mathcal{G}_{Ch}(f) = \mathcal{G}_o \quad \text{constant} \qquad f \in \mathcal{B} \tag{4.35}$$

Consequently, the impulse response is given by

$$g_{Ch}(t) = \mathcal{G}_o\delta(t) \tag{4.36}$$

where, observing (4.31),

$$\mathcal{G}_o = \sqrt{\mathtt{g}_d} = \frac{1}{\sqrt{\mathtt{a}_d}} \tag{4.37}$$

We note that, in case the conditions leading to (4.31) are not verified, the relation between $\mathcal{G}_o$ and $\mathtt{g}_d$ is more complicated (see (4.25)).

Measurement of signal power

Typically $\mathtt{g}_d$ and $\mathtt{a}_d$ are expressed in dB; the power $\mathtt{P}$ is expressed in W, mW (10^{-3} W), or pW (10^{-12} W), or in dBW, dBm, or dBrn:

$$(\mathtt{P})_{dBW} = 10\log_{10} \qquad (\mathtt{P} \text{ in W}) \tag{4.38}$$

$$(\mathtt{P})_{dBm} = 10\log_{10} \qquad (\mathtt{P} \text{ in mW}) \tag{4.39}$$

$$(\mathtt{P})_{dBrn} = 10\log_{10} \qquad (\mathtt{P} \text{ in pW}) \tag{4.40}$$

Some relations are

$$(\mathtt{P})_{dBW} = (\mathtt{P})_{dBrn} - 120 = (\mathtt{P})_{dBm} - 30 \tag{4.41}$$

Example 4.1.1
For $\mathtt{P} = 0.5$ W we have $(\mathtt{P})_{dBW} = -3$ dBW, and $(\mathtt{P})_{dBm} = 27$ dBm.

With reference to (4.37), we note that $(\mathcal{G}_o)_{dB} = (\mathtt{g}_d)_{dB}$. In fact, as $\mathcal{G}_o$ denotes a ratio of voltages, it follows that

$$(\mathcal{G}_o)_{dB} = 20\log_{10}\mathcal{G}_o = 10\log_{10}\mathtt{g}_d = (\mathtt{g}_d)_{dB} \tag{4.42}$$

For telephone signals, a further power unit is given by dBrnc, which expresses the power in dBrn of a signal filtered according to the mask given in Figure 4.5 [2]. The filter reflects the perception of the human ear and is known as *C-message weighting*.

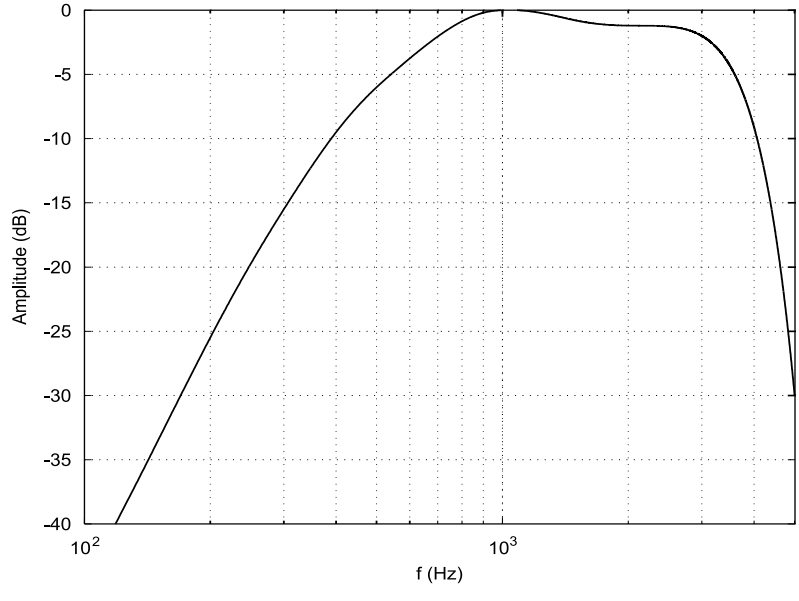


Figure 4.5. Frequency weighting known as C-message weighting. [© *1982 Bell Telephone Laboratories. Reproduced with permission of Lucent Technologies, Inc./Bell Labs.*]

4.2 Noise generated by electrical devices and networks

Various noise and disturbance signals are added to the desired signal at different points of a transmission system. In addition to interference caused by electromagnetic coupling between various system elements and noise coming from the surrounding environment, there is also noise generated by the transmission devices themselves. Such noise is very important because it determines the limits of the system. We will analyze two types of noise generated by transmission devices: *thermal noise* and *shot noise*.

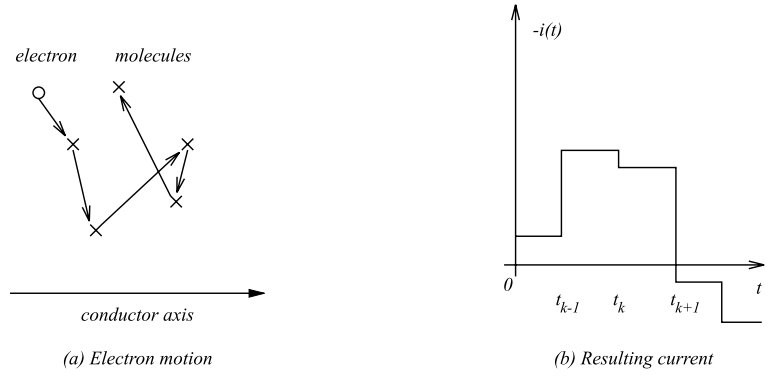
Thermal noise

Thermal noise is a phenomenon associated with Brownian or random motion of electrons in a conductor. As each electron carries a unit charge, its motion between collisions with atoms produces a short impulse of current. Actually, if we represent the motion of an electron within a conductor in a two-dimensional plane, the typical behavior is represented in Figure 4.6a where the changes in the direction of the electron motion are determined by random collisions with atoms at the set of instants $\{t_k\}$. Between two consecutive collisions the electron produces a current that is proportional to the projection of the velocity onto the axis of the conductor. For example, the behavior of instantaneous current for the path of Figure 4.6a is illustrated in Figure 4.6b. Although the average value (DC component) is zero, the large number of electrons and collisions gives origin to a measurable alternating component. If a current flows through the conductor, an orderly motion is superimposed on the disorderly motion of electrons; the sources of the two motions do not interact with each other. For a conductor of resistance R, at an absolute temperature of $\mathtt{T}$ Kelvin, the power spectral density of the open circuit voltage w at the conductor terminals is given by

$$\mathcal{P}_w(f) = 2\mathtt{k}\mathtt{T}R\gamma(f) \tag{4.43}$$

where $\mathtt{k} = 1.3805 \cdot 10^{-23}$ J/K is the Boltzmann constant and

$$\gamma(f) = \frac{hf}{\mathtt{k}\mathtt{T}}\left(e^{\frac{hf}{\mathtt{k}\mathtt{T}}} - 1\right)^{-1} \tag{4.44}$$



(a) Electron motion (b) Resulting current

Figure 4.6. Representation of electron motion and current produced by the motion.

where $h = 6.6262 \cdot 10^{-34}$ Js is the Planck constant. We note that, for $f \ll \mathtt{kT}/h = 6 \cdot 10^{12}$ Hz (at room temperature $\mathtt{T} = 290$ K), we get $\gamma(f) \simeq 1$. Therefore the PSD of w is approximately white, i.e.

$$\mathcal{P}_w(f) = 2\mathtt{kT}R \tag{4.45}$$

We adopt the electrical model of Figure 4.7, where a conductor is modelled as a noiseless device having in series a generator of noise voltage w.[2] Because at each instant the noise voltage $w(t)$ is due to the superposition of several current pulses, a suitable model for the *amplitude* distribution of $w(t)$ is the *Gaussian* distribution with zero mean. Note that the variance is very large, because of the wide support of $\mathcal{P}_w$. In the case of a linear two-terminal device with impedance $Z = R + jX$ at absolute temperature T, the spectral density of the open circuit voltage w is still given by (4.43), where $R = Re[Z]$. In other words, only the resistive component of the impedance gives origin to thermal noise. Let us consider the scheme of Figure 4.8, where a noisy impedance $Z = R + jX$ is matched to the load for maximum transfer of power. Observing (4.11), the available noise power per

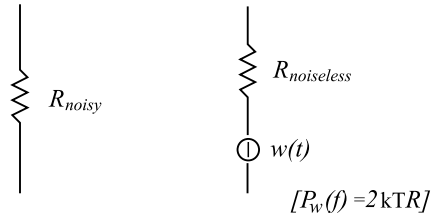
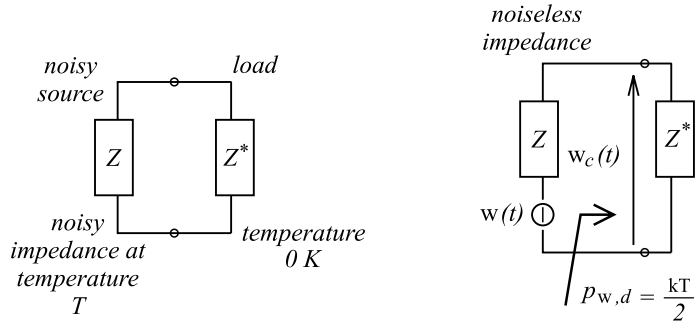


Figure 4.7. Electrical model of a noisy conductor.



(a) Electrical circuit (b) Equivalent scheme

Figure 4.8. Electrical circuit to measure the available source noise power to the load.

[2] An equivalent model assumes a noiseless conductor in parallel to a generator of noise current $j(t)$ with PSD $\mathcal{P}_j(f) = 2\mathtt{kT}\frac{1}{R}\gamma(f)$.

unit of frequency is given by

$$\mathtt{p}_{w,d}(f) = \frac{\mathtt{kT}}{2} \text{ (W/Hz)} \tag{4.46}$$

At room temperature $\mathtt{T} = 290$ K, $\mathtt{p}_{w,d}(f) = 2 \cdot 10^{-21}$ (W/Hz) and

$$(\mathtt{p}_{w,d}(f))_{dBm} = -177 \text{ (dBm/Hz)} \tag{4.47}$$

If the circuit of Figure 4.8 has a bandwidth B, the power delivered to the load is equal to

$$\mathtt{P}_w = \frac{\mathtt{kT}}{2} 2B = \mathtt{kT}B \text{ (W)} \tag{4.48}$$

We note that a noisy impedance produces an open circuit voltage w with a root mean-square (rms) value equal to

$$\sigma_w = \sqrt{\mathcal{P}_w \, 2B} = \sqrt{\mathtt{p}_{w,d} \, 4R2B} = \sqrt{\mathtt{kT} \, 4RB} \text{ (V)} \tag{4.49}$$

We also note from (4.48) that the total available power of a thermal noise source is proportional to the product of the system bandwidth and the absolute temperature of the source. For $\mathtt{T} = 290$ K,

$$(\mathtt{P}_w)_{dBm} = -174 + 10 \log_{10} B \text{ (dBm)} \tag{4.50}$$

Shot noise

Most devices are affected by shot noise, which is due to the discrete nature of electron flow: also in this case the noise represents the instantaneous random deviation of current or voltage from the average value. Shot noise, expressed as a current signal, can also be modelled as Gaussian noise with a constant PSD given by

$$\mathcal{P}_{i_{shot}}(f) = eI \text{ (A}^2\text{/Hz)} \tag{4.51}$$

where $e = 1.6 \cdot 10^{-19}$ C is the electron charge and I is the average current that flows through the device; in this case it is convenient to use the electrical model of Figure 4.4.

Noise in diodes and transistors

Models are given in the literature to describe the different noise sources in electronic devices. Specifically, in [2] shot noise is evaluated for a junction diode and shot and thermal noise for a transistor. In any case, the total output noise power of a device is usually not described by $\mathtt{p}$, but rather by an equivalent function called *noise temperature*.

Noise temperature of a two-terminal device

Let $\mathtt{p}_{w,d}$ be the available power per unit of frequency, due to the presence of noise in a device. The noise temperature is defined as

$$\mathtt{T}_w(f) = \frac{\mathtt{p}_{w,d}(f)}{\mathtt{k}/2} \tag{4.52}$$

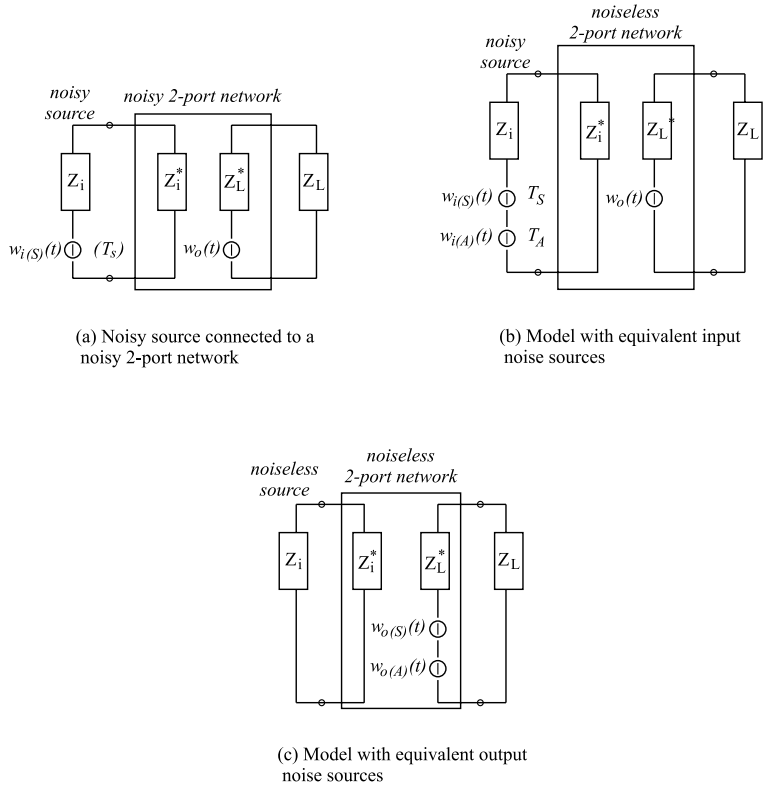
In other words, $\mathtt{T}_w$ represents the absolute temperature that a thermal noise source should have in order to produce the same available noise power as the device. This concept can be extended and applied to the output of an amplifier or an antenna, expressing the noise power in terms of *effective noise temperature*. We note that if a device at absolute temperature $\mathtt{T}$ contains more than one noise source, then $\mathtt{T}_w > \mathtt{T}$.

Noise temperature of a 2-port network

We will consider the circuit of Figure 4.9a, where both the source impedance and the load impedance are matched for maximum transfer of power. Assuming that the source, with noise temperature $\mathtt{T}_S$, generates a noise voltage $w_{i(S)}$, and $\mathtt{g}_d$ is the available power gain of the 2-port network defined in (4.30), the noise voltage generated at the network output because of the presence of $w_{i(S)}$ is equal to $w_o = w_{o(S)}$, with available power at the load given by

$$\mathtt{p}_{w_{o(S)}}(f) = \mathtt{g}_d(f)\frac{\mathtt{k}\mathtt{T}_S}{2} \tag{4.53}$$

If in addition to the source, the network also introduces noise, which if measured at the output is equal to $w_{o(A)}$, with available power $\mathtt{p}_{w_{o(A)}}$, we will have a total output noise



(a) Noisy source connected to a noisy 2-port network

(b) Model with equivalent input noise sources

(c) Model with equivalent output noise sources

Figure 4.9. Noise source connected to a noisy 2-port network: three equivalent models.

signal given by $w_o = w_{o(S)} + w_{o(A)}$. Assuming the two noise signals $w_{o(S)}$ and $w_{o(A)}$ are uncorrelated, the available power at the load will be equal to the sum of the two powers, i.e.

$$\mathrm{p}_{w_o}(f) = \mathrm{g}_d(f)\frac{\mathrm{k}\mathrm{T}_S}{2} + \mathrm{p}_{w_{o(A)}}(f) \tag{4.54}$$

Definition 4.7
The *effective noise temperature* T_A of the 2-port network is defined as

$$\mathrm{T}_A(f) = \frac{p_{w_{o(A)}}(f)}{\mathrm{g}_d(f)\frac{\mathrm{k}}{2}} \tag{4.55}$$

and denotes the temperature of a thermal noise source connected to a 2-port noiseless network that produces the same output noise power. Then (4.54) becomes:[3]

$$\mathrm{p}_{w_o}(f) = \mathrm{g}_d(f)\frac{\mathrm{k}}{2}[\mathrm{T}_S + \mathrm{T}_A] \tag{4.56}$$

Definition 4.8
The *effective input temperature* of a system consisting of a source connected to a 2-port network is

$$\mathrm{T}_{w_i} = \mathrm{T}_S + \mathrm{T}_A \tag{4.57}$$

Definition 4.9
The *effective output temperature* of a system consisting of a source connected to a 2-port network is

$$\mathrm{T}_{w_o} = \mathrm{g}_d(f)\mathrm{T}_{w_i} \tag{4.58}$$

Then

$$\mathrm{p}_{w_o}(f) = \frac{\mathrm{k}}{2}\mathrm{T}_{w_o} \tag{4.59}$$

Equivalent-noise models

By the previous considerations, we introduce the equivalent circuits illustrated in Figure 4.9b and 4.9c. In particular the scheme of Figure 4.9b assumes the network to be noiseless and an equivalent noise source is considered at the input. The scheme of Figure 4.9c, on the other hand, considers all noise sources at the output. The effects on the load for the three schemes of Figure 4.9 are the same.

[3] To simplify the notation we have omitted indicating the dependency on frequency of all noise temperatures. Note that the dependency on frequency of T_A and T_S is determined by $\mathrm{g}_d(f)$, $\mathrm{p}_{w_{o(A)}}(f)$, and $\mathrm{p}_{w_{i(S)}}(f)$.

Noise figure of a 2-port network

Usually the noise of a 2-port network is not directly characterized through $\mathtt{T}_A$, but through the *noise figure* $\mathtt{F}$. Recognizing that $\mathtt{p}_{w_{o(A)}}$ does not depend on $\mathtt{T}_S$, as the source and network noise signals are generated by uncorrelated phenomena, leads to the following experiment. We set the source at a noise temperature equal to the room temperature: $\mathtt{T}_{S^0} = \mathtt{T}_0 = 290$ K. This is obtained by disconnecting the source and setting as an input to the 2-port network an impedance Z_i equal to the source impedance; now the noise $w_{i(S^0)}$ will only be thermal noise with noise temperature equal to $\mathtt{T}_0$. The noise figure is given by the ratio between the available power at the load due to the total noise power $\mathtt{p}_{w_o} = \mathtt{p}_{w_{o(A)}} + \mathtt{p}_{w_{o(S^0)}}$ and that due only to the source $\mathtt{p}_{w_{o(S^0)}}$[4]

$$\begin{aligned} \mathtt{F}(f) &= \frac{\mathtt{p}_{w_o}(f)}{\mathtt{p}_{w_{o(S^0)}}(f)} \\ &= 1 + \frac{\mathtt{p}_{w_{o(A)}}(f)}{\mathtt{p}_{w_{o(S^0)}}(f)} \end{aligned} \tag{4.61}$$

Being $\mathtt{p}_{w_{o(S^0)}}(f) = \mathtt{g}_d(f)\frac{\mathtt{kT}_0}{2}$, and substituting for $\mathtt{p}_{w_{o(A)}}$ the expression (4.55), we obtain the important relation

$$\mathtt{F}(f) = 1 + \frac{\mathtt{T}_A}{\mathtt{T}_0} \tag{4.62}$$

We note that $\mathtt{F}$ is always greater than 1 and it equals 1 in the ideal case of a noiseless 2-port network. Moreover, $\mathtt{F}$ is a parameter of the network and does not depend on the noise temperature of the source to which it is connected. From (4.61) the noise power of the 2-port network can be expressed as

$$\mathtt{p}_{w_{o(A)}}(f) = \frac{\mathtt{k}}{2}(\mathtt{F} - 1)\mathtt{T}_0\mathtt{g}_d \tag{4.63}$$

From the above considerations we deduce that to describe the noise of an *active* 2-port network, we must assign the gain $\mathtt{g}_d$ and the noise figure $\mathtt{F}$ (or equivalently the noise temperature $\mathtt{T}_A$). We now see that for a *passive* network at temperature $\mathtt{T}_0$, it is sufficient to assign only one of the two parameters. Let us consider a passive network at temperature $\mathtt{T}_0$, as for example a transmission line, for which $\mathtt{g}_d < 1$. To determine the noise figure let us assume as source an impedance, which is matched to the network for maximum transfer of power, at temperature $\mathtt{T}_0$. Applying Thevenin theorem to the network output, the system is equivalent to a two-terminal device with impedance Z_2 at temperature $\mathtt{T}_0$.

---

[4] Given an electrical circuit, a useful relation to determine $\mathtt{F}$, equivalent to (4.61), that employs the PSDs of the output noise signals, is given by (see (4.9))

$$\mathtt{F}(f) = \frac{\mathcal{P}_{w_o}(f)}{\mathcal{P}_{w_{o(S^0)}}(f)} = 1 + \frac{\mathcal{P}_{w_{o(A)}}(f)}{\mathcal{P}_{w_{o(S^0)}}(f)} \tag{4.60}$$

Assuming the load is matched for maximum transfer of power, i.e. $Z_2 = Z_L^*$, from (4.46) at the output we have $\mathrm{p}_{w_0}(f) = (\mathrm{kT}_0/2)$. On the other hand, $\mathrm{p}_{w_{i(S^0)}}(f) = (\mathrm{kT}_0/2)$, and $\mathrm{p}_{w_{0(S^0)}}(f) = \mathrm{g}_d \mathrm{p}_{w_{i(S^0)}}$.

Hence from the first of (4.61) we have

$$\mathrm{F}(f) = \frac{1}{\mathrm{g}_d} = \mathrm{a}_d \tag{4.64}$$

where a_d is the power attenuation of the network. Note that also in this case, given F, we can determine the effective noise temperature of the network, T_A, according to (4.62). Summarizing, in a connection between a source and a 2-port network, the effective input temperature of the system can be expressed as

$$\mathrm{T}_{w_i} = \mathrm{T}_S + \mathrm{T}_A = \mathrm{T}_S + (\mathrm{F} - 1)\mathrm{T}_0 \tag{4.65}$$

and the available noise power at the load is given by

$$\mathrm{p}_{w_0}(f) = \mathrm{g}_d(f)\frac{\mathrm{kT}_{w_i}}{2} \tag{4.66}$$

Example 4.2.1
Let us consider the configuration of Figure 4.10a, where an antenna with noise temperature T_S is connected to a pre-amplifier with available power gain g and noise figure F. An electrical model of the connection is given in Figure 4.10b, where the antenna is modelled as a resistance with noise temperature T_S. If the impedances of the two devices are matched, T_{w_i} is given by (4.65).

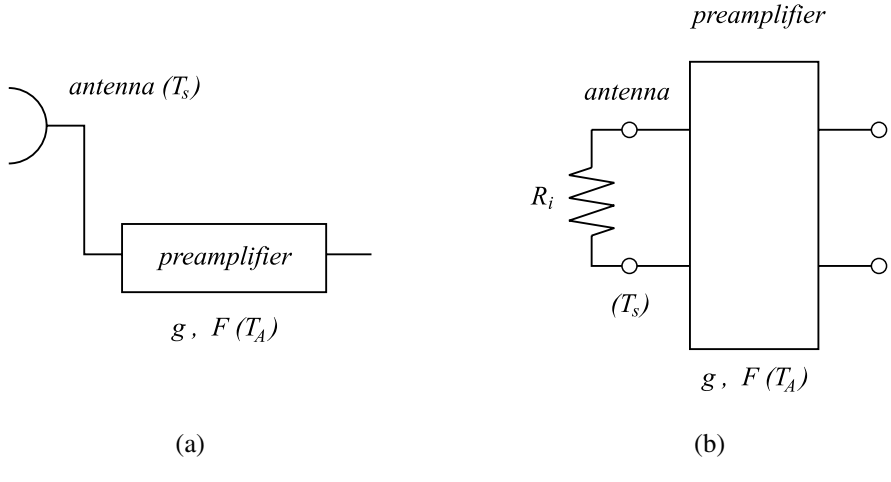


Figure 4.10. Antenna-preamplifier configuration and electrical model.

Cascade of 2-port networks

As shown in Figure 4.11, we consider the cascade of two 2-port networks A_1 and A_2, with available power gains $\mathtt{g}_1$ and $\mathtt{g}_2$ and noise figures $\mathtt{F}_1$ and $\mathtt{F}_2$, respectively. Assuming the impedances are matched for maximum transfer of power between different networks, we wish to determine the parameters of a network equivalent to the cascade of the two networks. With regard to the power gain, the overall network has a gain $\mathtt{g}$ equal to the product of the gains of the individual networks:

$$\mathtt{g} = \mathtt{g}_1\mathtt{g}_2 \tag{4.67}$$

With regard to the noise characteristics, it is sufficient to determine the noise figure of the cascade of the two networks. For a source at room temperature $\mathtt{T}_0$, from (4.66) for $\mathtt{T}_S = \mathtt{T}_0$, the noise power at the output of the first network is given by

$$\mathtt{p}_{w_{o,1}}(f) = \frac{\mathtt{k}\mathtt{T}_0}{2}\mathtt{F}_1\mathtt{g}_1 \tag{4.68}$$

At the output of the second network we have

$$\mathtt{p}_{w_{o,2}}(f) = \mathtt{p}_{w_{o,1}}(f)\mathtt{g}_2 + \frac{\mathtt{k}}{2}(\mathtt{F}_2 - 1)\mathtt{T}_0\mathtt{g}_2 \tag{4.69}$$

using (4.63) to express the noise power due to the second network only. Then the noise figure of the overall network is given by

$$\mathtt{F} = \frac{\mathtt{p}_{w_{o,2}}(f)}{\mathtt{p}_{w_{o,2(S^0)}}(f)} = \frac{\frac{\mathtt{k}\mathtt{T}_0}{2}\mathtt{g}_1\mathtt{g}_2\mathtt{F}_1 + \frac{\mathtt{k}\mathtt{T}_0}{2}(\mathtt{F}_2 - 1)\mathtt{g}_2}{\frac{\mathtt{k}\mathtt{T}_0}{2}\mathtt{g}_1\mathtt{g}_2} \tag{4.70}$$

Simplifying (4.70) we get

$$\mathtt{F} = \mathtt{F}_1 + \frac{(\mathtt{F}_2 - 1)}{\mathtt{g}_1} \tag{4.71}$$

Extending this result to the cascade of N 2-port networks, A_i, $i = 1, \ldots, N$, characterized by gains $\mathtt{g}_i$ and noise figures $\mathtt{F}_i$, we obtain *Friis formula* of the total noise figure

$$\mathtt{F} = \mathtt{F}_1 + \frac{(\mathtt{F}_2 - 1)}{\mathtt{g}_1} + \frac{(\mathtt{F}_3 - 1)}{\mathtt{g}_1\mathtt{g}_2} + \cdots + \frac{(\mathtt{F}_N - 1)}{\mathtt{g}_1\mathtt{g}_2 \cdots \mathtt{g}_{N-1}} \tag{4.72}$$

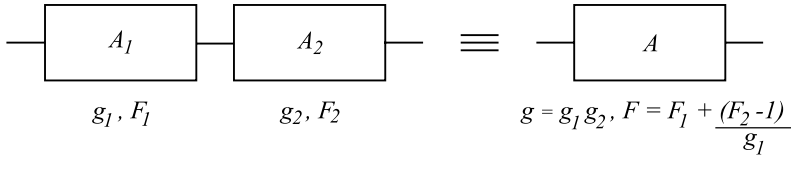


Figure 4.11. Equivalent scheme of a cascade of two 2-port networks.

We observe that $\mathtt{F}$ strongly depends on the gain and noise figure parameters of the first stages; in particular, the smaller $\mathtt{F}_1$ and the larger $\mathtt{g}_1$, the more $\mathtt{F}$ will be reduced. Substituting (4.62), that relates the noise figure to the effective noise temperature, in (4.72), we have that the equivalent noise temperature of the cascade of N 2-port networks, characterized by noise temperatures $\mathtt{T}_{A_i}$, $i = 1, \ldots, N$, is given by

$$\mathtt{T}_A = \mathtt{T}_{A_1} + \frac{\mathtt{T}_{A_2}}{\mathtt{g}_1} + \frac{\mathtt{T}_{A_3}}{\mathtt{g}_1\mathtt{g}_2} + \cdots + \frac{\mathtt{T}_{A_N}}{\mathtt{g}_1\mathtt{g}_2 \cdots \mathtt{g}_{N-1}} \tag{4.73}$$

Obviously the total gain of the cascade is given by

$$\mathtt{g} = \mathtt{g}_1\mathtt{g}_2 \cdots \mathtt{g}_N \tag{4.74}$$

Example 4.2.2
The idealized configuration of a transmission medium consisting of a very long cable where amplifiers are inserted at equally spaced points is illustrated in Figure 4.12. Each section of the cable, with power attenuation $\mathtt{a}_c$ and noise figure $\mathtt{F}_c = \mathtt{a}_c$ (see (4.64)), cascaded with an amplifier, with gain $\mathtt{g}_A$ and noise figure $\mathtt{F}_A$, is called a *repeater section*. To compensate for the attenuation of the cable we choose $\mathtt{g}_A = \mathtt{a}_c$. Then, each *section* has a gain

$$\mathtt{g}_{sr} = \frac{1}{\mathtt{a}_c}\mathtt{g}_A = 1 \tag{4.75}$$

and noise figure

$$\mathtt{F}_{sr} = \mathtt{F}_c + \frac{\mathtt{F}_A - 1}{\mathtt{g}_c} = \mathtt{a}_c + \mathtt{a}_c(\mathtt{F}_A - 1) = \mathtt{g}_A\mathtt{F}_A \tag{4.76}$$

Therefore the N sections have overall unit gain and noise figure

$$\begin{aligned} \mathtt{F} &= \mathtt{F}_{sr} + \frac{(\mathtt{F}_{sr} - 1)}{\mathtt{g}_{sr}} + \cdots + \frac{(\mathtt{F}_{sr} - 1)}{\mathtt{g}_{sr}\mathtt{g}_{sr} \cdots \mathtt{g}_{sr}} \\ &= N(\mathtt{F}_{sr} - 1) + 1 \\ &\simeq N\mathtt{F}_{sr} \end{aligned} \tag{4.77}$$

where $\mathtt{F}_{sr}$ is given by (4.76). We note that the output noise power of N repeater sections is N times the noise power introduced by an individual section.

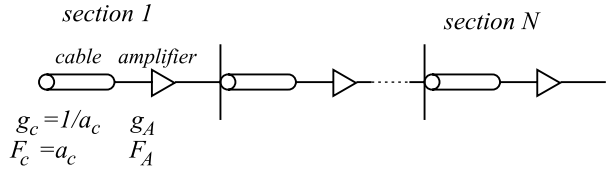


Figure 4.12. Transmission channel composed of N repeater sections.

4.3 Signal-to-noise ratio (SNR)

SNR for a two-terminal device

Let us consider the circuit of Figure 4.3, where the source v_b generates a desired signal s and a noise signal w:

$$v_b(t) = s(t) + w(t) \tag{4.78}$$

To measure the level of the desired signal with respect to the noise, one of the most widely used methods considers the *signal-to-noise ratio* (SNR), defined as the ratio of the statistical powers

$$\Lambda_s = \frac{\mathtt{M}_s}{\mathtt{M}_w} = \frac{E[s^2(t)]}{E[w^2(t)]} = \frac{\displaystyle\int_{-\infty}^{+\infty} \mathcal{P}_s(f)\,df}{\displaystyle\int_{-\infty}^{+\infty} \mathcal{P}_w(f)\,df} \tag{4.79}$$

On the other hand, the effects of the two signals on a certain load Z_c are measured by the average powers. Therefore we also introduce the following signal-to-noise ratio of average powers

$$\Lambda_p = \frac{\mathtt{P}_s}{\mathtt{P}_w} = \frac{\displaystyle\int_{-\infty}^{+\infty} \mathtt{p}_s(f)\,df}{\displaystyle\int_{-\infty}^{+\infty} \mathtt{p}_w(f)\,df} \tag{4.80}$$

where, from (4.9),

$$\mathtt{p}_s(f) = \mathcal{P}_s(f)\frac{R_c}{|Z_b + Z_c|^2} \tag{4.81}$$

$$\mathtt{p}_w(f) = \mathcal{P}_w(f)\frac{R_c}{|Z_b + Z_c|^2} \tag{4.82}$$

Therefore Λ_s and Λ_p are in general different. However, if the term $R_c/|Z_b + Z_c|^2$ is a constant within the passband of s and w, then the two SNRs coincide. Note that if $Z_b = Z_c^*$, that is the condition for maximum transfer of power is satisfied, then $R_c/|Z_b + Z_c|^2 = 1/(4R_b)$. Hence it is sufficient that R_b is constant within the passband of s and w to have

$$\Lambda = \Lambda_s = \Lambda_p \tag{4.83}$$

Moreover, assuming $\mathtt{p}_w$ is constant within the passband of w, with bandwidth B, we have

$$\mathtt{P}_w = \frac{\mathtt{k}}{2}\mathtt{T}_w 2B \tag{4.84}$$

and, from (4.80),

$$\Lambda = \frac{E[s^2(t)]}{E[w^2(t)]} = \frac{\mathtt{P}_s}{\mathtt{k}\mathtt{T}_w B} \tag{4.85}$$

where $\mathtt{P}_s$ is the *available average power* of the desired signal, and $\mathtt{T}_w$ is the *noise temperature* of w. Later we will often use this relation.

SNR for a 2-port network

Let us consider now the connection of a source to the linear 2-port network of Figure 4.2b, where v_i has a desired component s and a noise component w_i (see Figure 4.9b):

$$v_i(t) = s(t) + w_i(t) \tag{4.86}$$

and w_i has an effective noise temperature $\mathtt{T}_{w_i} = \mathtt{T}_S + \mathtt{T}_A$. Therefore $\mathtt{p}_{w_i}(f) = \mathtt{k}\mathtt{T}_{w_i}/2$. The open circuit voltage of the network output is given by

$$v_o(t) = s_o(t) + w_o(t) \tag{4.87}$$

where s_o and w_o depend on s and w_i, respectively. Under matched load conditions (that is $Z_L = Z_2^*$), and assuming (4.83) holds, at the network output we obtain

$$\Lambda_{out} = \frac{E[s_o^2(t)]}{E[w_o^2(t)]} = \frac{\mathtt{P}_{s_o}}{\mathtt{P}_{w_o}} \tag{4.88}$$

We indicate with $\mathcal{B}$ the passband of the network frequency response, usually equal to or including the passband of s, and with B its bandwidth. From the expressions (4.4) and (4.30)

$$\mathtt{P}_{s_o} = \int_{-\infty}^{+\infty} \mathtt{p}_{s_o}(f)\,df = 2\int_{\mathcal{B}} \mathtt{p}_s(f)\mathtt{g}(f)\,df \tag{4.89}$$

and

$$\mathtt{P}_{w_o} = 2\int_{\mathcal{B}} \mathtt{p}_{w_i}(f)\mathtt{g}(f)\,df \tag{4.90}$$

Assuming now that $\mathtt{g}(f)$ is constant within $\mathcal{B}$, we have

$$\mathtt{P}_{s_o} = \mathtt{P}_s\mathtt{g} \quad \text{and} \quad \mathtt{P}_{w_o} = \frac{\mathtt{k}}{2}\mathtt{T}_{w_i}\mathtt{g}2B \tag{4.91}$$

assuming that also the source is matched for maximum transfer of power. Finally we get

$$\Lambda_{out} = \frac{E[s_o^2(t)]}{E[w_o^2(t)]} = \frac{\mathtt{P}_s}{\mathtt{k}\mathtt{T}_{w_i}B} \tag{4.92}$$

where $\mathtt{P}_s$ is the *available power* of the desired signal at the network input, and $\mathtt{T}_{w_i} = \mathtt{T}_S + (\mathtt{F}-1)\mathtt{T}_0$ is the *effective noise temperature* including both the source and the 2-port network. With reference to the above configuration, we observe that the power of w_i could be very high if $\mathtt{T}_{w_i}$ is constant over a wide band, but w_o has much smaller power since its passband coincides with that of the network frequency response. From (4.91) and (4.50), the *effective input noise* due to the connection source-network has an *average power* for $\mathtt{T}_S = \mathtt{T}_0$ ($\mathtt{T}_{w_i} = \mathtt{F}\mathtt{T}_0$) equal to

$$(\mathtt{P}_{w_i})_{dBm} = -114 + 10\log_{10} B\,|_{MHz} + (\mathtt{F})_{dB} \quad (\mathtt{T}_S = \mathtt{T}_0) \tag{4.93}$$

and the average power of the *effective output noise* is given by

$$(\mathtt{P}_{w_o})_{dBm} = (\mathtt{P}_{w_i})_{dBm} + (\mathtt{g})_{dB} \quad (\mathtt{T}_S = \mathtt{T}_0) \tag{4.94}$$

In (4.93), $B|_{MHz}$ denotes the bandwidth in MHz.

Example 4.3.1
A station, receiving signals from a satellite, has an antenna with gain $\mathtt{g}_{ant}$ of 40 dB and a noise temperature $\mathtt{T}_S$ of 60 K (that is the antenna acts as a noisy resistor at a temperature of 60 K). The antenna feeds a preamplifier with a noise temperature $\mathtt{T}_{A_1}$ of 125 K and a gain $\mathtt{g}_1$ of 20 dB. The preamplifier is followed by an amplifier with a noise figure $\mathtt{F}_2$ of 10 dB and a gain $\mathtt{g}_2$ of 80 dB. The transmitted signal bandwidth is 1 MHz. The satellite has an antenna with a power gain of $\mathtt{g}_{sat} = 6$ dB and the total attenuation $\mathtt{a}_\ell$ due to the distance between transmitter and receiver is 190 dB. We want to find:

1. the average power of the thermal noise at the receiver output,
2. the minimum power of the signal transmitted by the satellite to obtain a SNR of 20 dB at the receiver output.

The two receiver amplifiers can be modelled as one amplifier with gain:

$$(\mathtt{g}_A)_{dB} = (\mathtt{g}_1)_{dB} + (\mathtt{g}_2)_{dB} = 20 + 80 = 100\,\text{dB} \tag{4.95}$$

and effective noise temperature:

$$\mathtt{T}_A = \mathtt{T}_{A_1} + \frac{\mathtt{T}_{A_2}}{\mathtt{g}_1} = \mathtt{T}_{A_1} + \frac{(\mathtt{F}_2 - 1)\mathtt{T}_0}{\mathtt{g}_1} = 125 + \frac{(10^{10/10} - 1)290}{10^{20/10}} = 151\,\text{K} \tag{4.96}$$

1. From (4.91) the average power of the output noise is

$$\begin{aligned} \mathtt{P}_{w_o} = \mathtt{k}(\mathtt{T}_S + \mathtt{T}_A)\mathtt{g}_A B &= 1.38 \times 10^{-23}\,(60 + 151)\,10^{100/10}\,10^6 \\ &= 2.91 \times 10^{-5}\,\text{W} = -15.36\,\text{dBm} \end{aligned} \tag{4.97}$$

2. From $\Lambda_{out} = (\mathtt{P}_{s_o}/\mathtt{P}_{w_o}) \geq 20$ dB we get $(\mathtt{P}_{s_o}/\mathtt{P}_{w_o}) \geq 100$. As $\mathtt{P}_{s_o} = \mathtt{P}_s \mathtt{g}_{sat}(1/\mathtt{a}_\ell)\,\mathtt{g}_{ant}\mathtt{g}_A = \mathtt{P}_s 10^{-44/10}$, it follows

$$\mathtt{P}_s \geq 73\,\text{W} \tag{4.98}$$

Relation between noise figure and SNR

For a source at room temperature $\mathtt{T}_S = \mathtt{T}_0$, given that $\mathtt{p}_{w_{i(S^0)}}(f) = \mathtt{k}\mathtt{T}_0/2$, it can be shown that

$$\mathtt{F} = \frac{\mathtt{p}_s(f)/\mathtt{p}_{w_{i(S^0)}}(f)}{\mathtt{p}_{s_o}(f)/\mathtt{p}_{w_o}(f)} \tag{4.99}$$

A more useful relation is obtained assuming that $\mathtt{g}(f)$ is a constant within the passband $\mathcal{B}$ of the network. Given the average power of the noise generated by the source at room temperature

$$\mathtt{P}_{w_{i(S^0)}} = \frac{\mathtt{k}\mathtt{T}_0}{2} 2B \tag{4.100}$$

Table 4.1 Parameters of three devices.

Device	F (dB)	T_A (K)	g (dB)	Frequency
maser	0.16	11	$20 \div 30$	6 GHz
TWT amplifier	2.7	250	$20 \div 30$	3 GHz
IC amplifier	7.0	1163	50	$\leq$70 MHz

and

$$\Lambda_{in} = \frac{P_s}{P_{w_{i(S^0)}}} \tag{4.101}$$

then, from (4.92), we have

$$\Lambda_{out} = \frac{\Lambda_{in}}{F} \ (T_S = T_0) \tag{4.102}$$

In other words, F is a measure of the reduction of the SNR at the output due to the noise introduced by the network. In Table 4.1 the typical values of F, T_A, and gain g are given for three devices. In the last column the frequency range usually considered for the operations of each device is also given.

4.4 Transmission lines

4.4.1 Fundamentals of transmission line theory

In this section, the principles of signal propagation in transmission lines are briefly reviewed. A uniform transmission line consists of a two-conductor cable with a uniform cross-section, that supports the propagation of transverse electromagnetic (TEM) waves [3, 1]. Examples of transmission lines are *twisted-pair* cables and *coaxial* cables. We now develop the basic transmission line theory. With reference to Figure 4.13, which illustrates a uniform line, let

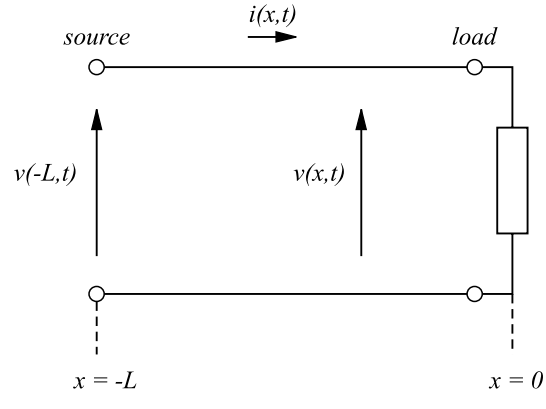


Figure 4.13. Uniform transmission line of length *L*.

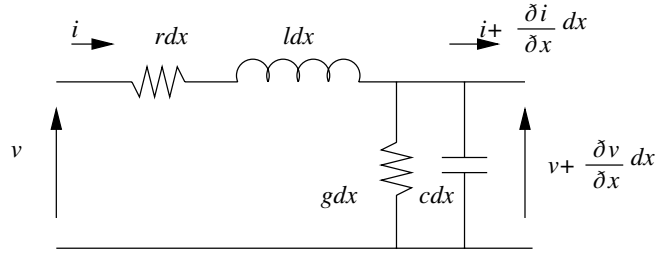


Figure 4.14. Line segment of infinitesimal length *dx*.

x denote the distance from the origin and L be the length of the line. The termination is found at distance $x = 0$ and the signal source at $x = -L$. Let $v = v(x, t)$ and $i = i(x, t)$ be, respectively, the voltage and current at distance x at time t. To determine the law that establishes the voltage and current along the line, let us consider a uniform line segment of infinitesimal length that we assume to be time invariant, depicted in Figure 4.14. The parameters r, ℓ, g, c are known as *primary constants* of the line. They define, respectively, resistance, inductance, conductance and capacitance of the line per unit length. Primary constants are in general slowly time-varying functions of the frequency; however, in this context, they will be considered time invariant. The model of Figure 4.14 is obtained using the first order Taylor series expansion of $v(x, t)$ and $i(x, t)$ as a function of distance x.

Ideal transmission line

We initially assume an ideal lossless transmission line characterized by $r = g = 0$. Voltage and current variations in the segment dx are given by

$$\begin{cases} \dfrac{\partial v}{\partial x} dx = -(\ell\, dx) \dfrac{\partial i}{\partial t} \\ \dfrac{\partial i}{\partial x} dx = -(c dx) \dfrac{\partial v}{\partial t} \end{cases} \tag{4.103}$$

Differentiating the first equation with respect to distance and the second with respect to time, we obtain

$$\begin{cases} \dfrac{\partial^2 v}{\partial x^2} = -\ell \dfrac{\partial^2 i}{\partial x \partial t} \\ \dfrac{\partial^2 i}{\partial t \partial x} = -c \dfrac{\partial^2 v}{\partial t^2} \end{cases} \tag{4.104}$$

Substituting $\partial^2 i / \partial t \partial x$ in the first equation with the expression obtained from the second, we get the wave equation

$$\frac{\partial^2 v}{\partial x^2} = \ell c \frac{\partial^2 v}{\partial t^2} = \frac{1}{v^2} \frac{\partial^2 v}{\partial t^2} \tag{4.105}$$

where $v = 1/\sqrt{\ell c}$ represents the *velocity of propagation* of the signal on a lossless transmission line. The general solution to the wave equation for a lossless transmission line is given by

$$v(x,t) = \varphi_1\left(t - \frac{x}{v}\right) + \varphi_2\left(t + \frac{x}{v}\right) \tag{4.106}$$

where φ_1 and φ_2 are arbitrary functions. Noting that from (4.103) $\partial i/\partial t = -(1/\ell)\partial v/\partial x$, (4.106) yields

$$-\ell\frac{\partial i}{\partial t} = -\frac{1}{v}\varphi_1'\left(t - \frac{x}{v}\right) + \frac{1}{v}\varphi_2'\left(t + \frac{x}{v}\right) \tag{4.107}$$

where φ_1' and φ_2' are the derivatives of φ_1 and φ_2, respectively.

Integrating by parts (4.107) we get

$$i(x,t) = \frac{1}{\ell v}\left[\varphi_1\left(t - \frac{x}{v}\right) - \varphi_2\left(t + \frac{x}{v}\right)\right] + \varphi(x) \tag{4.108}$$

where $\varphi(x)$ is time independent and can therefore be ignored in the study of propagation. Defining the *characteristic impedance of a lossless transmission line* as

$$Z_0 = \ell v = \sqrt{\frac{\ell}{c}} \tag{4.109}$$

the expression for the current is given by

$$i(x,t) = \frac{1}{Z_0}\left[\varphi_1\left(t - \frac{x}{v}\right) - \varphi_2\left(t + \frac{x}{v}\right)\right] \tag{4.110}$$

From the general solution to the wave equation we find that the voltage (or the current), considered as a function of distance along the line, consists of two waves that propagate in opposite directions: the wave that propagates from the source to the line termination is called the *source or incident wave*, that which propagates in the opposite direction is called *reflected wave*. We consider now the propagation of a sinusoidal wave with frequency $f = \omega/2\pi$ in an ideal transmission line. The voltage at distance $x = 0$ is given by

$$v(0,t) = V_0\cos(\omega t) \tag{4.111}$$

The wave propagating in the positive direction of x is given by $v_+(x,t) = |V_+|\cos[\omega(t - x/v)]$, that propagating in the negative direction is given by $v_-(x,t) = |V_-|\cos[\omega(t + x/v) + \theta_p]$. The transmission line voltage is obtained as the sum of the two components and is given by

$$v(x,t) = |V_+|\cos\left[\omega\left(t - \frac{x}{v}\right)\right] + |V_-|\cos\left[\omega\left(t + \frac{x}{v}\right) + \theta_p\right] \tag{4.112}$$

The current has the expression

$$i(x,t) = \frac{|V_+|}{Z_0}\cos\left[\omega\left(t - \frac{x}{v}\right)\right] - \frac{|V_-|}{Z_0}\cos\left[\omega\left(t + \frac{x}{v}\right) + \theta_p\right] \tag{4.113}$$

Let us consider a point on the x-axis individuated at each time instant t by the condition that the argument of the function $F(t-x/v)$ is a constant. This point is seen by an observer as moving at velocity v in the positive direction of the x-axis. For sinuosoidal waves the velocity for which the phase is a constant is called *phase velocity* v. It is useful to write (4.112) and (4.113) in complex notation, where the phasors V and I represent amplitude and phase at distance x of the sinusoidal signals (4.112) and (4.113), respectively,

$$V = V_+ e^{-j\beta x} + V_- e^{j\beta x} \tag{4.114}$$

$$I = \frac{1}{Z_0}(V_+ e^{-j\beta x} - V_- e^{j\beta x}) \tag{4.115}$$

where $\beta = \omega/v$ denotes the *phase constant*. We define the *wavelength* as $\lambda = 2\pi/\beta$. We note that frequency f and wavelength λ are related by

$$\lambda = \frac{v}{f} \tag{4.116}$$

In particular, the propagation in free space is characterized by $v = c = 3 \cdot 10^8$ m/s. If V_+ is taken as the reference phasor with phase equal to zero, then $V_- = |V_-|e^{j\theta_p}$, where θ_p is the phase rotation between the incident and the reflected waves at $x = 0$. Let us consider a transmission line having as termination an impedance Z_L. By Kirchhoff laws, the voltage and current at the termination are given by

$$\begin{cases} V_L = V_+ + V_- \\ I_L = \dfrac{V_L}{Z_L} = \dfrac{V_+}{Z_0} - \dfrac{V_-}{Z_0} \end{cases} \tag{4.117}$$

The *reflection coefficient* is defined as the ratio between the phasors representing, respectively, the reflected and incident waves, $\varrho = V_-/V_+$. The *transmission coefficient* is defined as the ratio between the phasors representing, respectively, the termination voltage and the incident wave $\tau = V_L/V_+$. From (4.117), it turns out

$$\varrho = \frac{Z_L - Z_0}{Z_L + Z_0} = \left|\frac{V_-}{V_+}\right| e^{j\theta_p} \tag{4.118}$$

and

$$\tau = \frac{2Z_L}{Z_L + Z_0} \tag{4.119}$$

At the termination, defining the incident power as $\mathtt{P}_+ = |V_+|^2/Z_0$ and the reflected power as $\mathtt{P}_- = |V_-|^2/Z_0$, we obtain $\mathtt{P}_-/\mathtt{P}_+ = |\varrho|^2$; the ratio between the power delivered to the load and the incident power is hence given by $1-|\varrho|^2$. Let us consider some specific cases:

- if $Z_L = Z_0$, $\varrho = 0$ and there is no reflection;
- if $Z_L = \infty$, the line is open-circuited, $\varrho = 1$ and $V_- = V_+$;
- if $Z_L = 0$, the line is short-circuited, $\varrho = -1$ and $V_- = -V_+$.

Non-ideal transmission line

Typically, in a transmission line the primary constants r and g are different from zero. For sinusoidal waves in steady state, the changes in voltage and current in a line segment of infinitesimal length characterized by an impedance Z and an admittance Y per unit length can be expressed using complex notation as

$$\begin{cases} \dfrac{dV}{dx} = -ZI \\ \dfrac{dI}{dx} = -YV \end{cases} \tag{4.120}$$

Differentiating and substituting in the first equation the expression of dI/dx obtained from the second, we get

$$\frac{d^2V}{dx^2} = \gamma^2 V \tag{4.121}$$

where

$$\gamma = \sqrt{ZY} \tag{4.122}$$

is a characteristic constant of the transmission line called *propagation constant*. Let α and β be, respectively, the real and imaginary parts of γ: α is the *attenuation constant* measured in *neper per unit of length*, and β is the *phase constant* measured in *radians per unit of length*. The solution of the differential equation for the voltage can be expressed in terms of exponential functions as

$$V = V_+ e^{-\gamma x} + V_- e^{\gamma x} = V_+ e^{-\alpha x} e^{-j\beta x} + V_- e^{\alpha x} e^{j\beta x} \tag{4.123}$$

The expression of the current is given by

$$I = \frac{1}{Z_0}\left(V_+ e^{-\gamma x} - V_- e^{\gamma x}\right) \tag{4.124}$$

where

$$Z_0 = \sqrt{\frac{Z}{Y}} \tag{4.125}$$

is the *characteristic impedance* of the transmission line. The propagation constant and the characteristic impedance are also known as *secondary constants* of the transmission line.

Frequency response

Let us consider the transmission line of Figure 4.15, with a sinusoidal voltage source v_i and a load Z_L. From (4.123) the voltage at the load can be expressed as $V_L = V_+(1+\varrho)$. Recalling that $V_-/V_+ = \varrho$, we define the voltage $V_o = V_L \mid_{Z_L=\infty} = V_+(1+\varrho) \mid_{\varrho=1} = 2V_+$.

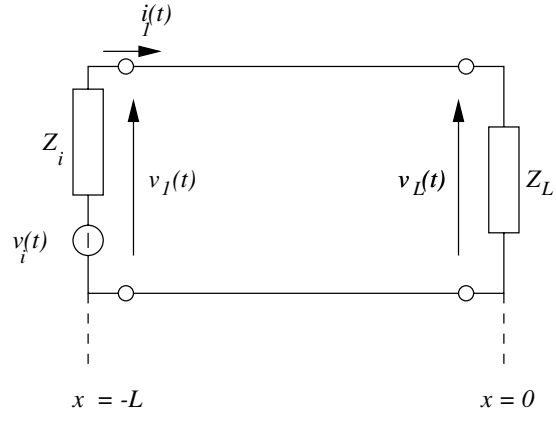


Figure 4.15. Transmission line with sinusoidal voltage generator v_i and load Z_L.

For the voltage V_1 and current I_1 we find

$$\begin{cases} V_1 = V_i - Z_i I_1 = V_+(e^{\gamma L} + \varrho e^{-\gamma L}) \\ I_1 = \dfrac{V_+}{Z_0}(e^{\gamma L} - \varrho e^{-\gamma L}) \end{cases} \tag{4.126}$$

where Z_i denotes the generator impedance.

The input and output impedances of the 2-port network are, respectively, given by:

$$Z_1 = \frac{V_1}{I_1} = Z_0 \frac{1 + \varrho e^{-2\gamma L}}{1 - \varrho e^{-2\gamma L}} \tag{4.127}$$

$$Z_2 = \frac{V_{o_{oc}}}{I_{L_{sc}}} = \frac{V_+(1+\varrho)\mid_{\varrho=1}}{\frac{V_+}{Z_0}(1-\varrho)\mid_{\varrho=-1}} = Z_0 \tag{4.128}$$

where $I_{L_{sc}} = I_L \mid_{Z_L=0}$ and $V_{o_{oc}} = V_L \mid_{Z_L=\infty}$.

We now want to determine the ratio between the voltage V_L and the voltage V_1, defined as $\mathcal{G}_{Ch} = V_L/V_1$. Observing the above relations we find the following frequency responses:

$$\mathcal{G}_L = \frac{V_L}{V_o} = V_+(1+\varrho)\frac{1}{2V_+} = \frac{1+\varrho}{2} = \frac{Z_L}{Z_L + Z_0} \tag{4.129}$$

$$\mathcal{G}_{1-o} = \frac{V_o}{V_1} = \frac{2e^{-\gamma L}}{1 + \varrho e^{-2\gamma L}} \tag{4.130}$$

$$\mathcal{G}_i = \frac{V_1}{V_i} = \frac{Z_1}{Z_i + Z_1} = \frac{Z_0(1 + \varrho e^{-2\gamma L})}{Z_0(1 + \varrho e^{-2\gamma L}) + Z_i(1 - \varrho e^{-2\gamma L})} \tag{4.131}$$

Then, from (4.1), the channel frequency response is given by:

$$\mathcal{G}_{Ch} = \mathcal{G}_{1-o}\mathcal{G}_L = \frac{(1+\varrho)e^{-\gamma L}}{1+\varrho e^{-2\gamma L}} \tag{4.132}$$

Let us consider some specific cases:

- *Matched transmission line*: $\varrho = 0$ for $Z_i = Z_L = Z_0$ ($\mathcal{G}_i = 1/2$)

$$\mathcal{G}_{Ch} = e^{-\gamma L} \tag{4.133}$$

- *Short-circuited transmission line*: $\varrho = -1$

$$\mathcal{G}_{Ch} = 0 \tag{4.134}$$

- *Open-circuited transmission line*: $\varrho = 1$

$$\mathcal{G}_{Ch} = \frac{2e^{-\gamma L}}{1+e^{-2\gamma L}} = \frac{1}{\cosh(\gamma L)} \tag{4.135}$$

To determine the power gain of the network, we can use the general equation (4.25), or observe (4.23); in any case, we obtain

$$\mathtt{g}(f) = \frac{1-|\varrho|^2}{1-|\varrho|^2 e^{-4\alpha L}} e^{-2\alpha L} \tag{4.136}$$

where $\alpha = Re[\gamma]$.

We note that, for a matched transmission line, the *available attenuation* is given by

$$\mathtt{a}_d(f) = \frac{1}{|e^{-\gamma L}|^2} = e^{2\alpha L} \tag{4.137}$$

In (4.137), α expresses the attenuation in *neper per unit of length*. Alternatively, one can introduce an attenuation in *dB per unit of length*, $(\tilde{\mathtt{a}}_d(f))_{dB}$, as

$$\mathtt{a}_d(f) = 10^{\frac{1}{10}(\tilde{\mathtt{a}}_d(f))_{dB} L} \tag{4.138}$$

The relation between α and $(\tilde{\mathtt{a}}_d(f))_{dB}$ is given by

$$(\tilde{\mathtt{a}}_d(f))_{dB} = 8.68\alpha \tag{4.139}$$

From (4.139), the attentuation in dB introduced by the transmission line is equal to

$$(\mathtt{a}_d(f))_{dB} = (\tilde{\mathtt{a}}_d(f))_{dB} L \tag{4.140}$$

In a transmission line with a non-matched resistive load that satisfies the condition $Z_L \ll Z_0$, from (4.118) we get $1+\varrho \simeq 2Z_L/Z_0$, $\varrho^2 e^{-4\alpha L} \simeq 0$, and $\varrho^2 \simeq 1 - 4Z_L/Z_0$. Therefore (4.136) yields

$$(\mathtt{a}_d(f))_{dB} = (\tilde{\mathtt{a}}_d(f))_{dB} L - 10\log_{10} 4 \left|\frac{Z_L}{Z_0}\right| \tag{4.141}$$

Conditions for the absence of signal distortion

We recall that Heaviside conditions for the absence of signal distortion are satisfied if $\mathcal{G}_{Ch}(f)$ has a constant amplitude and a linear phase, at least within the passband of the source. For a matched transmission line, these conditions are satisfied if α is a constant and β is a linear function of the frequency. The secondary parameters of the transmission line can be expressed as $\gamma = \alpha + j\beta = \sqrt{(r + j\omega\ell)(g + j\omega c)}$, and $Z_0 = \sqrt{(r + j\omega\ell)/(g + j\omega c)}$. For a matched transmission line, it can be shown that Heaviside conditions are equivalent to the condition

$$rc = g\ell \tag{4.142}$$

In the special case $g = 0$, we obtain

$$\alpha = \omega\sqrt{\frac{\ell c}{2}}\left\{\left(1 + \frac{r^2}{\omega^2\ell^2}\right)^{1/2} - 1\right\}^{1/2} \tag{4.143}$$

and

$$\beta = \omega\sqrt{\frac{\ell c}{2}}\left\{\left(1 + \frac{r^2}{\omega^2\ell^2}\right)^{1/2} + 1\right\}^{1/2} \tag{4.144}$$

For frequencies at which $r \ll \omega\ell$, using the approximation

$$\left(1 + \frac{r^2}{\omega^2\ell^2}\right)^{1/2} \simeq 1 + \frac{1}{2}\frac{r^2}{\omega^2\ell^2} \tag{4.145}$$

we find

$$\alpha \simeq \frac{r}{2}\sqrt{\frac{c}{\ell}} \quad \text{and} \quad \beta \simeq \omega\sqrt{\ell c} \tag{4.146}$$

Impulse response of a non-ideal transmission line

For commonly used transmission lines, a more accurate model of the propagation constants, that takes into account the variation of r with the frequency due to the skin effect, shows that both the attenuation constant and the phase constant must include a term proportional to the square root of frequency. An expression of the propagation constant generally used to characterize the propagation of TEM waves over a metallic transmission line [1] is

$$\gamma(f) = K\sqrt{\frac{\omega}{2}} + jK\sqrt{\frac{\omega}{2}} + j\omega\sqrt{\ell c} \tag{4.147}$$

where K is a constant that depends on the transmission line. The expression (4.147) is valid for both coaxial and twisted-pair cables insulated with plastic material. The attenuation constant of the transmission line is therefore given by

$$\alpha(f) = K\sqrt{\pi f} \text{ (neper/m)} \tag{4.148}$$

and the attenuation introduced by the transmission line can be expressed as

$$(\tilde{a}_d(f))_{dB} = 8.68K\sqrt{\pi f} \text{ (dB/m)} \tag{4.149}$$

We note that, given the value of $\alpha(f)$ at a certain frequency $f = f_0$, we can obtain the value of K. Therefore it is possible to determine the attenuation constant at every other frequency. From the expression (4.133) of the frequency response of a matched transmission line, with γ given by (4.147), without considering the delay $\sqrt{\ell c}$ introduced by the term $j\omega\sqrt{\ell c}$, the impulse response has the following expression

$$g_{Ch}(t) = \frac{KL}{2\sqrt{\pi t^3}} e^{-\frac{(KL)^2}{4t}} 1(t) \tag{4.150}$$

The pulse signal g_{Ch} is shown in Figure 4.16 for various values of the product KL. We note a larger dispersion of g_{Ch} for increasing values of KL.

Secondary constants of some transmission lines

In Table 4.2 we give the values of Z_0 and $\gamma = \alpha + j\beta$ experimentally measured for some telephone transmission lines characterized by a certain diameter, which is usually indicated by a parameter called *gauge*.

The behavior of α as a function of frequency is given in Figure 4.17 for four telephone lines [2]; we may note that it follows the $\sqrt{f}$ law in the range of frequencies $f < 10\,\text{kHz}$. For some transmission lines this law is followed also for $f > 100\,\text{kHz}$,

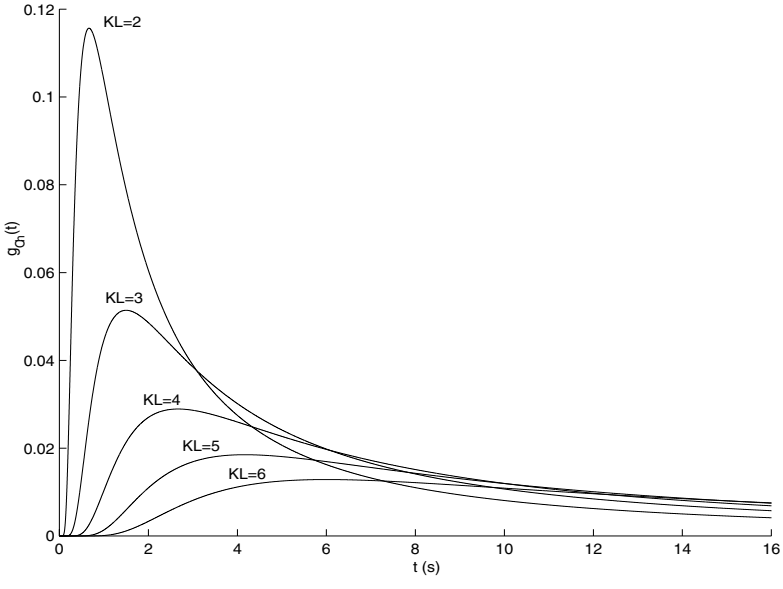


Figure 4.16. Impulse response of a matched transmission line for various values of *KL*.

Table 4.2 Secondary constants of some telephone lines.

Gauge diameter (mm)	Frequency (Hz)	Characteristic impedance Z_0 (Ω)	Propagation constant $\alpha + j\beta$ (neper/km) (rad/km)	Attenuation $\tilde{a}_d = 8.68\alpha$ (dB/km)
19 (0.9119)	1000	$297 - j278$	$0.09 + j0.09$	0.78
	2000	$217 - j190$	$0.12 + j0.14$	1.07
	3000	$183 - j150$	$0.15 + j0.18$	1.27
22 (0.6426)	1000	$414 - j401$	$0.13 + j0.14$	1.13
	2000	$297 - j279$	$0.18 + j0.19$	1.57
	3000	$247 - j224$	$0.22 + j0.24$	1.90
24 (0.5105)	1000	$518 - j507$	$0.16 + j0.17$	1.43
	2000	$370 - j355$	$0.23 + j0.24$	2.00
	3000	$306 - j286$	$0.28 + j0.30$	2.42
26 (0.4039)	1000	$654 - j645$	$0.21 + j0.21$	1.81
	2000	$466 - j453$	$0.29 + j0.30$	2.55
	3000	$383 - j367$	$0.35 + j0.37$	3.10

© 1982 Telephone Laboratories. Reproduced with permission of Lucent Technologies, Inc./Bell Labs.

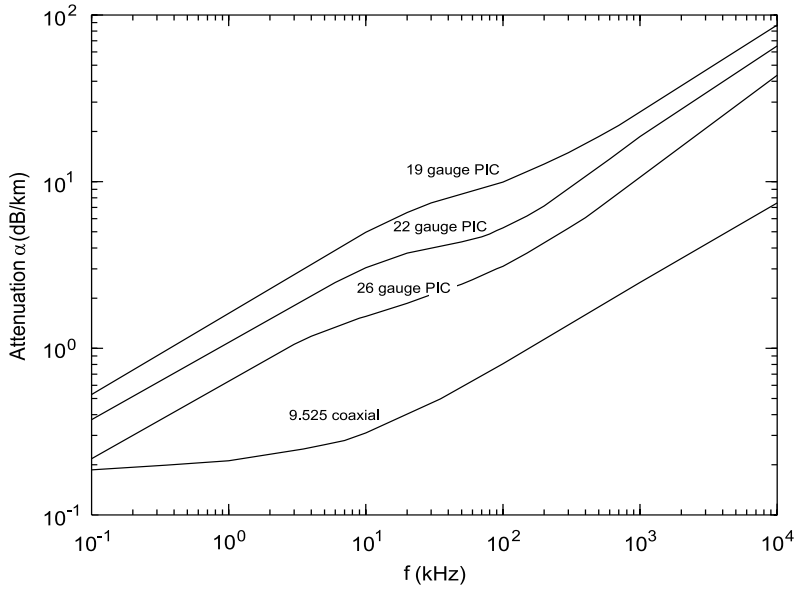


Figure 4.17. Attenuation as a function of frequency for some telephone transmission lines: three are *polyethylene-insulated cables* (PIC) and one is a coaxial cable with a diameter of 9.525 mm. [*© 1982 Bell Telephone Laboratories. Reproduced with permission of Lucent Technologies, Inc./Bell Labs.*]

albeit with a different constant of proportionality. In any case in the *local-loop*,[5] to force the primary constants to satisfy Heaviside conditions in the voice band, which goes from 300 to 3400 Hz, formerly some lump inductors were placed at equidistant points along the transmission line. This procedure, called *inductive loading*, causes $\alpha(f)$ to be flat in the voice band, but considerably increases the attenuation outside of the voice band. Moreover, the phase $\beta(f)$ may result very distorted in the passband. Typical behavior of α and β in the frequency band $0 \div 4000$ Hz, with and without *loading*, are given in Figure 4.18 for a transmission line with gauge 22 [2]. The *digital subscriber line* (DSL) technologies, introduced for data transmission in the local loop, require a bandwidth much greater than 4 kHz, up to about 20 MHz for the VDSL technology (see Chapter 17). For DSL applications it is therefore necessary to remove possible *loading* coils that are present in the *local loops*. The frequency response of a DSL transmission line can also be modified by the presence of one or more *bridged-taps*. A *bridged-tap* consists of a twisted pair cable of a certain length L_{BT}, terminated by an open circuit and connected in parallel to a *local loop*. At the connection point, the incident signal separates into two components. The component propagating along the bridged-tap is reflected at the point of the open circuit: the component propagating on the transmission line must therefore be calculated taking also into consideration this reflected component. At the frequencies $f_{BT} = v/\lambda_{BT}$, where λ_{BT} satisfies the condition $(2n+1)\lambda_{BT}/4 = L_{BT}$, $n = 0, 1, \ldots$, at the connection point we get destructive interference between the reflected and incident component: this interference reveals itself as a *notch* in the frequency response of the transmission line. Given

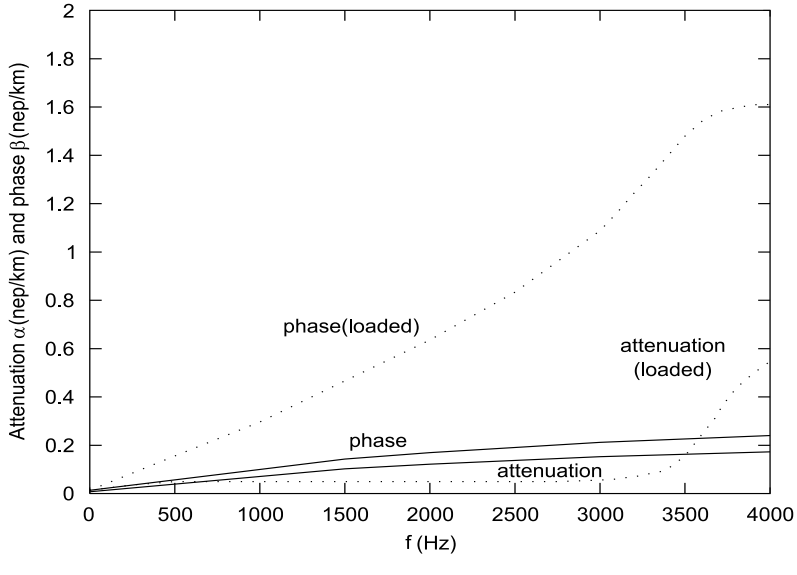


Figure 4.18. Attenuation constant α and phase constant β for a telephone transmission line with and without *loading*. [*© 1982 Bell Telephone Laboratories. Reproduced with permission of Lucent Technologies, Inc./Bell Labs.*]

[5] By *local-loop* we intend the transmission line that goes from the user telephone set to the central office.

Table 4.3 Transmission characteristics defined by the EIA/TIA for unshielded twisted pair (UTP) cables.

	Signal attenuation at 16 MHz	NEXT attenuation at 16 MHz	Characteristic impedance
UTP-3	13.15 dB/100 m	$\geq$23 dB	100 $\Omega \pm 15\%$
UTP-4	8.85 dB/100 m	$\geq$38 dB	100 $\Omega \pm 15\%$
UTP-5	8.20 dB/100 m	$\geq$44 dB	100 $\Omega \pm 15\%$

the large number of transmission lines actually in use, to evaluate the performance of DSL systems we usually refer to a limited number of *loop characteristics*, which can be viewed as samples taken from the ensemble of frequency responses. On the other hand, the transmission characteristics of *unshielded twisted-pair* (UTP) cables commonly used for data transmission over local area networks are defined by the EIA/TIA and ISO/IEC standards. As illustrated in Table 4.3, the cables are divided into different categories according to the values of 1) the signal attenuation per unit of length, 2) the attenuation of the near-end cross-talk signal, or NEXT, that will be defined in the next section, and 3) the characteristic impedance. Cables of category three (UTP-3) are commonly called *voice-grade*, those of categories four and five (UTP-4 and UTP-5) are *data-grade*. We note that the signal attenuation and the intensity of NEXT are substantially larger for UTP-3 cables than for UTP-4 and UTP-5 cables.

4.4.2 Cross-talk

The interference signal that is commonly referred to as cross-talk is determined by magnetic coupling and unbalanced capacitance between two adjacent transmission lines. Let us consider the two transmission lines of Figure 4.19, where the terminals $(1, 1')$ belong to the disturbing transmission line and the terminals $(2, 2')$ belong to the disturbed transmission line. In the study of the interference signal produced by magnetic coupling, we consider

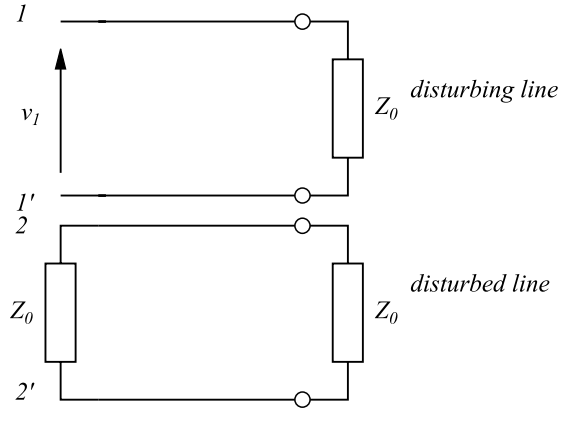


Figure 4.19. Transmission lines configuration for the study of cross-talk.

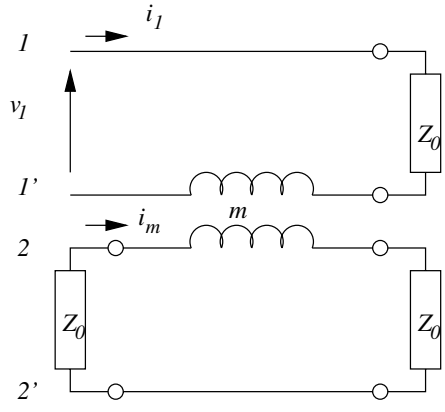


Figure 4.20. Interference signal produced by magnetic coupling.

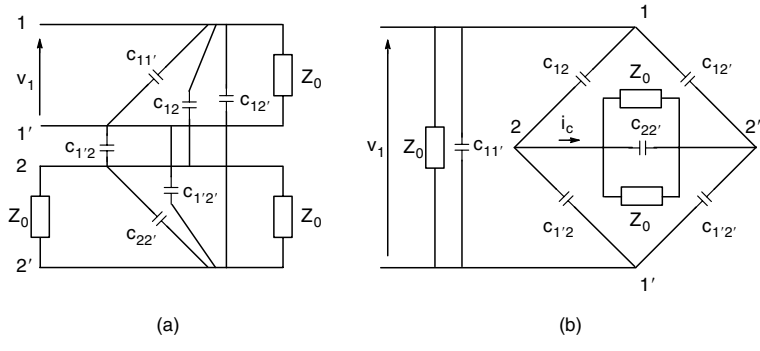


Figure 4.21. Interference signal produced by unbalanced capacitance.

the circuit of Figure 4.20. We will assume that the length of the transmission line is much longer than the wavelength corresponding to the maximum transmitted frequency and that the impedance Z_0 is much higher than the inductor reactance. The induced *electromagnetic force* (EMF) is given by $E = j2\pi f m I_1$, where $I_1 \simeq V_1/Z_0$. The EMF produces a current $I_m = \frac{E}{(1/(2Z_0))} = \frac{j2\pi f m}{(1/(2Z_0))} I_1$, that can be expressed as $I_m = \frac{j2\pi f m}{(1/(2Z_0^2))} V_1$.

To study the interference signal due to unbalanced capacitance, we consider the circuit of Figure 4.21a, that can be redrawn in an equivalent way as illustrated in Figure 4.21b. We assume that the impedance Z_0 is much smaller than the reactance of the capacitors that can be found on the bridge. Applying the principle of the equivalent generator we find

$$I_c = \frac{V_{22'}|_{I_c=0}}{Z_{22'}} = -\left(\frac{\frac{1}{c_{1'2'}}}{\frac{1}{c_{1'2'}} + \frac{1}{c_{12'}}} - \frac{\frac{1}{c_{1'2}}}{\frac{1}{c_{1'2}} + \frac{1}{c_{12}}} \right) \frac{1}{\frac{1}{c_{12} + c_{1'2}} + \frac{1}{c_{12'} + c_{1'2'}}} j2\pi f V_1 \tag{4.151}$$

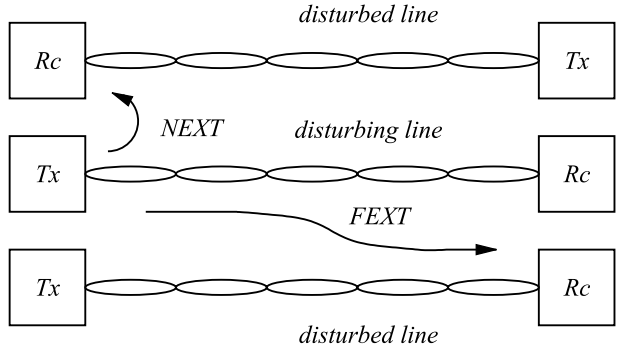


Figure 4.22. Illustration of near-end cross-talk (NEXT) and far-end cross-talk (FEXT) signals.

from which we obtain

$$I_c = \frac{c_{12}c_{1'2'} - c_{12'}c_{1'2}}{c_{12} + c_{1'2} + c_{12'} + c_{1'2'}} j2\pi f V_1 = j2\pi \Delta c V_1 \tag{4.152}$$

Recalling that the current I_c is equally divided between the impedances Z_0 on which the transmission line terminates, we find that the cross-talk current produced at the transmitter side termination is $I_p = -I_m + I_c/2$, and the cross-talk current produced at the receiver side termination is $I_t = I_m + I_c/2$. As illustrated in Figure 4.22, the interference signals are called *near-end cross-talk* or NEXT, or *far-end cross-talk* or FEXT, depending on whether the receiver side of the disturbed line is the same as the transmitter side of the disturbing line, or the opposite side, respectively. We now evaluate the total contribution of the near and far-end cross-talk signals for lines with distributed impedances.

Near-end cross-talk

Let

$$a_p(x) = -\frac{m(x)}{2Z_0} + \frac{\Delta c(x)}{2} Z_0 \tag{4.153}$$

be the near-end cross-talk coupling function at distance x from the origin. In complex notation, the NEXT signal is expressed as

$$V_p = Z_0 I_p = \int_0^L V_1 e^{-2\gamma x} j2\pi f a_p(x)\, dx \tag{4.154}$$

To calculate the power spectral density of NEXT we need to know the autocorrelation function of the random process $a_p(x)$. A model commonly used in practice assumes that $a_p(x)$ is a white stationary random process, with autocorrelation

$$\mathrm{r}_{a_p}(z) = E[a_p(x+z)a_p^*(x)] = \mathrm{r}_p(0)\delta(z) \tag{4.155}$$

For NEXT the following relation holds

$$E[|V_p(f)|^2] = E[|V_1(f)|^2]\frac{\pi^{3/2}}{K}\mathtt{r}_p(0)f^{3/2}(1 - e^{4K\sqrt{\pi f}L}) \simeq E[|V_1(f)|^2]k_p\, f^{3/2} \tag{4.156}$$

where K is defined by (4.148), and

$$k_p = \frac{\pi^{3/2}\mathtt{r}_p(0)}{K} \tag{4.157}$$

Using (1.449), the level of NEXT coupling is given by[6]

$$|\mathcal{G}_p(f)|^2 = \frac{E[|V_p(f)|^2]}{E[|V_1(f)|^2]} \simeq k_p\, f^{3/2} \tag{4.158}$$

To perform computer simulations of data transmission systems over metallic lines in the presence of NEXT, it is required to characterize not only the amplitude, but also the phase of NEXT coupling. In addition to experimental models obtained through laboratory measurements, the following stochastic model is used:

$$a_p(x) = \sum_{i=0}^{\frac{L}{\Delta x}-1} a_i \mathtt{w}_{\Delta x}(x - i\Delta x) \tag{4.159}$$

with

$$\mathtt{w}_{\Delta x}(x) = \begin{cases} 1 & \text{if } x \in [0, \Delta x) \\ 0 & \text{otherwise} \end{cases} \tag{4.160}$$

where a_i, $i = 0, \ldots, L/\Delta x - 1$, denote statistically independent Gaussian random variables with zero mean and variance

$$E[a_i^2] = \frac{\mathtt{r}_p(0)}{\Delta x} \tag{4.161}$$

A NEXT coupling function is thus given by

$$\mathcal{G}_{NEXT}(f) = \sum_{i=0}^{\frac{L}{\Delta x}-1} j2\pi f a_i \mathtt{w}_{\Delta x}(x - i\Delta x) e^{-2(K\sqrt{\pi f} + jK\sqrt{\pi f} + j2\pi f\sqrt{\ell c})(i+\frac{1}{2})\Delta x} \tag{4.162}$$

If we know the parameters of the transmission line K and k_p, then from (4.157) and (4.161) the variance of a_i to be used in the simulations is given by

$$E[a_i^2] = \frac{Kk_p}{\pi^{3/2}\Delta x} \tag{4.163}$$

[6] Observing (1.449), $|\mathcal{G}_p(f)|^2$ is also equal to the ratio between the PSDs of v_p and v_1.

Far-end cross-talk

Let

$$a_t(x) = \frac{m(x)}{2Z_0} + \frac{\Delta c(x)}{2} Z_0 \tag{4.164}$$

be the far-end cross-talk coupling function at distance x from the origin. In complex notation, the FEXT signal is given by

$$V_t = Z_0 I_t = \int_0^L V_1 \, e^{-\gamma L} \, j2\pi f \, a_t(x) \, dx \tag{4.165}$$

Analogously to the case of NEXT, we assume that a_t is a white stationary random process, with autocorrelation

$$\mathtt{r}_{a_t}(z) = E[a_t(x+z)a_t^*(x)] = \mathtt{r}_t(0)\delta(z) \tag{4.166}$$

For the FEXT signal the following relation holds

$$E[|V_t(f)|^2] = E[|V_1(f)|^2] e^{-2K\sqrt{\pi f}L} (2\pi f)^2 \mathtt{r}_t(0) L \tag{4.167}$$

where L is the length of the transmission line. The level of FEXT coupling is given by

$$|\mathcal{G}_t(f)|^2 = \frac{E[|V_t(f)|^2]}{E[|V_1(f)|^2]} = k_t f^2 L e^{-2K\sqrt{\pi f}L} \tag{4.168}$$

where $k_t = (2\pi)^2 \mathtt{r}_t(0)$.

We note that for high-speed data transmission systems over unshielded twisted-pair cables, NEXT usually represents the dominant source of interference.

Example 4.4.1

For local-area network (LAN) applications, the maximum length of cables connecting stations is typically limited to 100 m. Deviations from the characteristic expressed by (4.147) may be caused by losses in the dielectric material of the cable, the presence of connectors, non-homogeneity of the transmission line, etc.

For the IEEE Standard 100BASE-T2, which defines the physical layer for data transmission at 100 Mb/s over UTP-3 cables in Ethernet LANs (see Chapter 17), the following worst-case frequency response is considered:

$$\mathcal{G}_{Ch}(f) = 10^{-\frac{1.2}{20}} e^{-(0.00385\sqrt{jf} + 0.00028 f)L} \tag{4.169}$$

where f is expressed in MHz and L in meters. In (4.169), the term $e^{-j2\pi f\sqrt{\ell c}L}$ is ignored, as it indicates a constant propagation delay. A frequency independent attenuation of 1.2 dB has been included to take into account the attenuation caused by the possible presence of connectors. The amplitude of the frequency response obtained for a cable length $L = 100$ m is shown in Figure 4.23 [4]. We note that the signal attenuation at the frequency of 16 MHz is equal to 14.6 dB, a higher value than that indicated in Table 4.3 for UTP-3 cables.

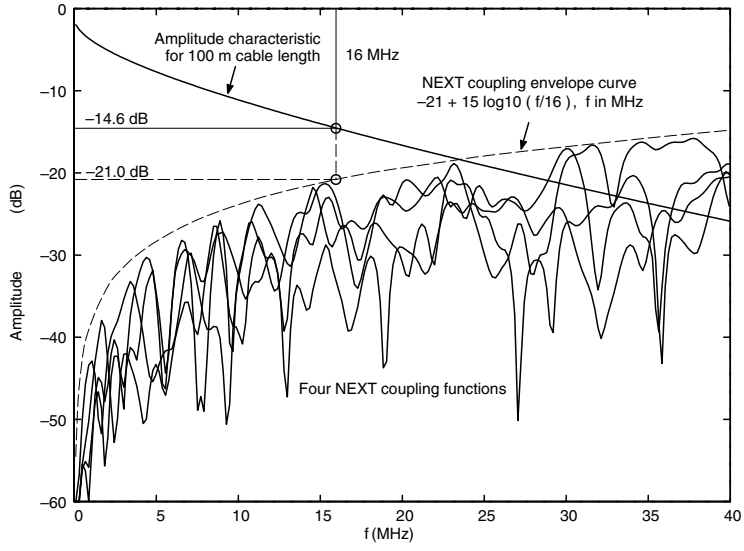


Figure 4.23. Amplitude of the frequency response for a voice-grade twisted-pair cable with length equal to 100 m, and four realizations of NEXT coupling function. [© *1997 IEEE*.]

The level of NEXT coupling (4.158) is illustrated in Figure 4.23 as a dotted line; we note the increase as a function of frequency of 15 dB/decade, due to the factor $f^{3/2}$. The level of NEXT coupling equal to 21 dB at the frequency of 16 MHz is larger than that given in Table 4.3 for UTP-3 cables. The amplitude characteristics of four realizations of the NEXT coupling function (4.162) are also shown in Figure 4.23.

4.5 Optical fibers

Transmission systems using light pulses that propagate over thin glass fibers were introduced in the 1970s and have since then undergone continuous development and experienced an increasing penetration, to the point that they now constitute a fundamental element of modern information highways. For in-depth study of optical fiber properties and of optical component characteristics we refer the reader to the vast literature existing on the subject [5, 6, 7]; in this section we limit ourselves to introducing some fundamental concepts.

The term "optical communications" is used to indicate the transmission of information by the propagation of electromagnetic fields at frequencies typically of the order of $10^{14} \div 10^{15}$ Hz, that are found in the optical band and are much higher than the frequency of radio waves or microwaves; to identify a transmission band, the wavelength rather than the frequency is normally used. We recall that for electromagnetic wave propagation in free space, the relation (4.116) holds: a frequency of $3 \cdot 10^{14}$ Hz corresponds therefore to a wavelength of 1 μm for transmission over optical fibers. The signal attenuation as a function of the wavelength exhibits the behavior shown in Figure 4.24 [8, 9]; we note that the useful interval for transmission is in the range from 800 to 1600 nm, that corresponds

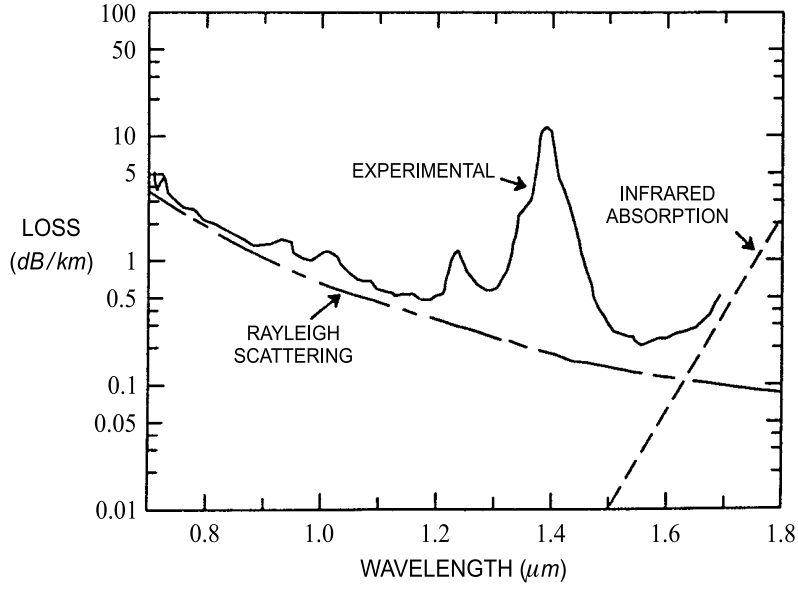


Figure 4.24. Attenuation curve as a function of wavelength for an optical fiber. [*From Li (1980), see also Miya et al. (1979), © 1980 IEEE.*]

to a bandwidth of $2 \cdot 10^{14}$ Hz. Three regions are typically used for transmission: the first window goes from 800 to 900 nm, the second from 1250 to 1350 nm, and the third from 1500 to 1600 nm.

We immediately realize the enormous capacity of fiber transmission systems: for example, a system that uses only 1% of the $2 \cdot 10^{14}$ Hz bandwidth mentioned above, has an available bandwidth of $2 \cdot 10^{12}$ Hz, equivalent to that needed for the transmission of ~300.000 television signals, each with a bandwidth of 6 MHz. To efficiently use the band in the optical spectrum, *multiplexing* techniques using optical devices have been developed, such as *wavelength-division multiplexing* (WDM) and *optical frequency-division multiplexing* (O-FDM). Moreover, we note that, although the propagation of electromagnetic fields in the atmosphere at these frequencies is also considered for transmission (see Section 17.2.1), the majority of optical communication systems employ as transmission medium an optical fiber, which acts as a waveguide. A fundamental device in optical communications is represented by the laser, which, beginning in the 1970s, made coherent light sources available for the transmission of signals.

Description of a fiber-optic transmission system

The main components of a fiber-optic transmission system are illustrated in Figure 4.25 [10]. Optical transmission lines with lengths of over a few hundred meters use fiber glass, because they present less attenuation with respect to fibers using plastic material. Dispersion in the transmission medium causes "spreading" of the transmitted pulses; this phenomenon in turn causes intersymbol interference and limits the available bandwidth of the transmission

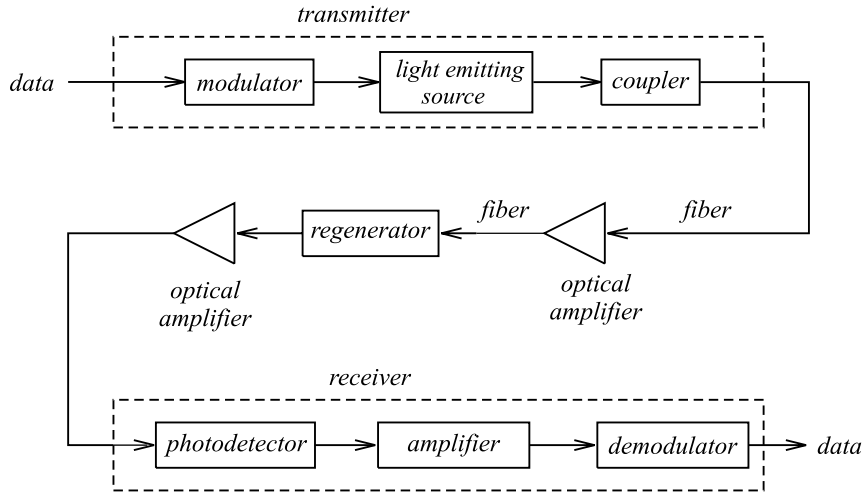


Figure 4.25. Elements of a typical fiber-optic transmission system.

medium. A measure of the pulse dispersion is given by

$$\Delta\tau = (M + M_g)\, L\, \Delta\lambda \tag{4.170}$$

where M is the dispersion coefficient of the material, M_g is the dispersion coefficient related to the geometry of the waveguide, L denotes the length of the fiber and $\Delta\lambda$ denotes the spectral width of the light source. The total dispersion $(M + M_g)$ has values near 120, 0, and 15 ps/(nm×km) at wavelengths of 850, 1300, and 1550 nm, respectively.

The bandwidth of the transmission medium is inversely proportional to the dispersion; we note that the dispersion is minimum in the second window, with values near zero around the wavelength of 1300 nm for conventional fibers. Special fibers are designed to compensate for the dispersion introduced by the material; because of the low attenuation and dispersion, these fibers are normally used in very long distance connections.

Multimode fibers allow the propagation of more than one mode of the electromagnetic field. In this case the medium introduces signal distortion caused by the fact that propagation of energy for different modes has different speeds: for this reason multimodal fibers are used in applications where the transmission bandwidth and the length of the transmission line are not large.

Monomode fibers limit the propagation to a single mode, thus eliminating the dispersion caused by multimode propagation. Because in this case the dispersion is due only to the material and the geometry of the waveguide, monomodal fibers are preferred for applications that require wide transmission bandwidth and very long transmission lines.

In Table 4.4 typical values of the transmission bandwidth, normalized by the length of the optical fiber, are given for different types of fibers. The *step-index* (SI) fiber is characterized by a constant value of the refraction index, whereas the *graded-index* (GRIN) fiber has a refraction index decreasing with the distance from the fiber axis. As noticed previously, the monomodal fibers are characterized by larger bandwidths; to limit the number of modes

Table 4.4 Characteristic parameters of various types of optical fibers.

Fiber	Wavelength (nm)	Source	Bandwidth (MHz·km)
multimode SI	850	LED	30
multimode GRIN	850	LD	500
multimode GRIN	1300	LD o LED	1000
monomode	1300	LD	>10000
monomode	1550	LD	>10000

to one, the diameter of the monomodal fiber is related to the wavelength and is normally about one order of magnitude smaller than that of multimodal fibers.

Semiconductor *laser diodes* (LD) or *light-emitting diodes* (LED) are used as signal light sources in most applications; these sources are usually modulated by electronic devices. The conversion from a current signal to an electromagnetic field that propagates along the fiber can be described in terms of light signal power by the relation

$$\mathrm{P}_{Tx} = k_0 + k_1 i \tag{4.171}$$

where k_0 and k_1 are constants. The transmitted waveform can therefore be seen as a replica of the modulation signal, in this case the current signal. Laser diodes are characterized by a smaller spectral width $\Delta\lambda$ as compared to that of LEDs, and therefore lead to a lower dispersion (see (4.170)).

The more widely used photodetector devices are semiconductor photodiodes, which convert the optical signal into a current signal according to the relation

$$i = \rho\, \mathrm{P}_{Rc} \tag{4.172}$$

where i is the device output current, P_{Rc} is the power of the incident optical signal and ρ is the photodetector response. Typical values of ρ are of the order of 0.5 mA/mW.

Signal quality is measured by the signal-to-noise ratio expressed as

$$\Lambda = \frac{(g_i\, \rho\, \mathrm{P}_{Rc})^2\, R_L}{g_i^n\, 2e\, R_L B(I_D + \rho\, \mathrm{P}_{Rc}) + 4\mathrm{k}\mathrm{T}_w B} \tag{4.173}$$

where g_i is the photodetector current gain, n is a parameter that indicates the photodetector excess noise, B is the receiver bandwidth, k is Boltzmann constant, e is the charge of the electron, T_w is the effective noise temperature in Kelvin, I_D is the photodetector *dark current*, and R_L is the resistance of the load that follows the photodetector. We note that in the denominator of (4.173) the first term is due to shot noise and the second term to thermal noise.

4.6 Radio links

The term *radio* is used to indicate the transmission of an electromagnetic field that propagates in free space. Some examples of radio transmission systems are:

- point-to-point terrestrial links [11];
- mobile terrestrial communication systems [12, 13, 14, 15, 16];

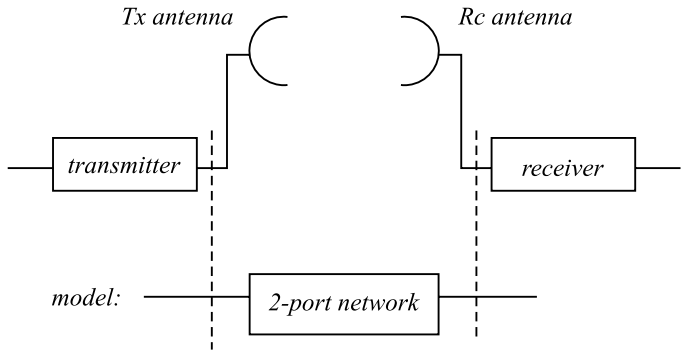


Figure 4.26. Radio link model.

- earth-satellite links (with satellites employed as signal repeaters) [17];
- deep-space communication systems (with space probes at a large distance from earth).

A radio link model is illustrated in Figure 4.26, where we assume that the transmit antenna input impedance and the receive antenna output impedance are matched for maximum transfer of power.

4.6.1 Frequency ranges for radio transmission

Frequencies used for radio transmission are in the range from about 100 kHz to some tens of GHz. The choice of the carrier frequency depends on various factors, among which the dimensions of the transmit antenna play an important role. In fact, to achieve an efficient radiation of electromagnetic energy, one of the dimensions of the antenna must be at least equal to $1/10$ of the carrier wavelength. This means that an AM radio station, with carrier frequency $f_0 = 1\,\text{MHz}$ and wavelength $\lambda = c/f_0 = 300\,\text{m}$, where c is the speed of light in free space, requires an antenna of at least 30 m.

A radio wave usually propagates as a ground wave (or surface wave), via *reflection* and *scattering* in the atmosphere (or via tropospheric scattering), or as a direct wave. Recall that, if the atmosphere is non-homogeneous (in terms of temperature, pressure, humidity, ...), the electromagnetic propagation depends on the changes of the refraction index of the medium. In particular, this gives origin to the reflection of electromagnetic waves. We speak of diffusion or scattering phenomena if molecules that are present in the atmosphere absorb part of the power of the incident wave and then re-emit it in all directions. Obstacles such as mountains, buildings, etc., give also origin to signal reflection and/or diffusion. In any case, these are phenomena that permit transmission between two points that are not in *line-of-sight* (LOS).

We will now consider the types of propagation associated with frequency bands.

Very low frequency (VLF) for $f_0 < 0.3\,\text{MHz}$. The earth and the ionosphere form a waveguide for the electromagnetic waves. At these frequencies the signals propagate around the earth.

Medium frequency (MF) for $0.3 < f_0 < 3\,\text{MHz}$. The waves propagate as ground waves up to a distance of 160 km.

High frequency (HF) for $3 < f_0 < 30\,\text{MHz}$. The waves are reflected by the ionosphere at an altitude that may vary between 50 and 400 km.

Very high frequency (VHF) for $30 < f_0 < 300\,\text{MHz}$. For $f_0 > 30\,\text{MHz}$, the signal propagates through the ionosphere with small attenuation. Therefore these frequencies are adopted for satellite communications. They are also employed for line-of-sight transmissions, using high towers where the antennas are positioned to *cover* a wide area. The limit to the coverage is set by the earth curvature. If h is the height of the tower in meters, the range covered expressed in km is $r = 1.3\sqrt{h}$: for example, if $h = 100\,\text{m}$, coverage is up to about $r = 13\,\text{km}$. However, ionospheric and tropospheric scattering (at an altitude of 16 km or less) are present at frequencies in the range 30–60 MHz and 40–300 MHz, respectively, which cause the signal to propagate over long distances with large attenuations.

Ultra high frequency (UHF) for $300\,\text{MHz} < f_0 < 3\,\text{GHz}$.

Super high frequency (SHF) for $3 < f_0 < 30\,\text{GHz}$. At frequencies of about 10 GHz, atmospheric conditions play an important role in signal propagation. We note the following *absorption phenomena*, which cause additional signal attenuation:

1. due to oxygen: for $f_0 > 30\,\text{GHz}$, with peak attenuation at 60 GHz;
2. due to water vapor: for $f_0 > 20\,\text{GHz}$, with peak attenuation at around 20 GHz;
3. due to rain: for $f_0 > 10\,\text{GHz}$, assuming the diameter of the rain drops is of the order of the signal wavelength.

We note that, if the antennas are not positioned high enough above the ground, the electromagnetic field propagates not only into the free space but also through ground waves.

Extremely high frequency (EHF) for $f_0 > 30\,\text{GHz}$.

Radiation masks

A radio channel by itself does not set constraints on the frequency band that can be used for transmission. In any case, to prevent interference among radio transmissions, regulatory bodies specify power radiation masks: a typical example is given in Figure 4.27, where the plot represents the limit on the power spectrum of the transmitted signal with reference to the power of a non-modulated carrier. To comply with these limits, a filter is usually employed at the transmitter front-end.

4.6.2 Narrowband radio channel model

The propagation of electromagnetic waves should be studied using Maxwell equations with appropriate boundary conditions. Nevertheless, for our purposes a very simple model, which

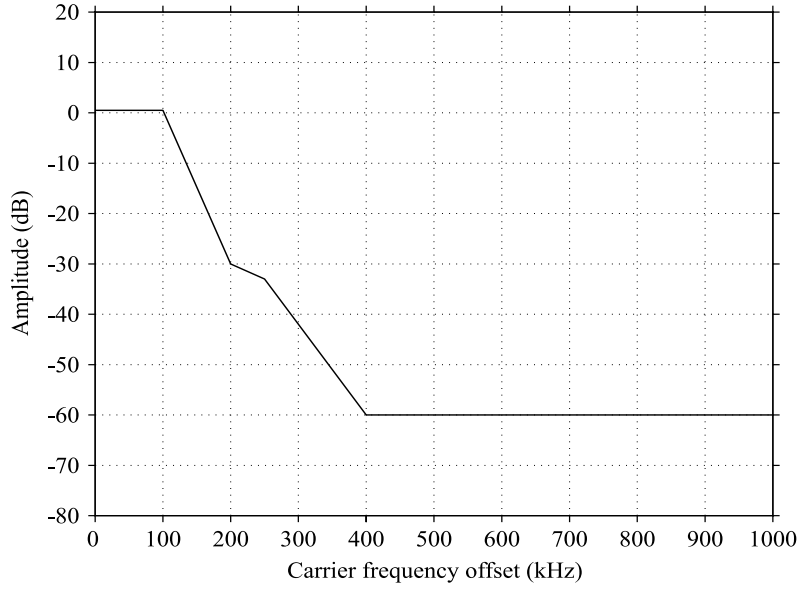


Figure 4.27. Radiation mask of the GSM system with a bandwidth of 200 kHz around the carrier.

consists in approximating an electromagnetic wave as a ray (in the optical sense), is often adequate.

The deterministic model is used to evaluate the power of the received signal when there are no obstacles between the transmitter and receiver, that is in the presence of line of sight: in this case we can think of only one wave that propagates from the transmitter to the receiver. This situation is typical of transmissions between satellites and terrestrial radio stations in the microwave frequency range ($3 < f_0 < 70$ GHz).

Let $\mathtt{P}_{Tx}$ be the power of the signal transmitted by an ideal *isotropic* antenna, which uniformly radiates in all directions in the free space. At a distance d from the antenna, the power density is

$$\Phi_0 = \frac{\mathtt{P}_{Tx}}{4\pi d^2} \,(\mathrm{W/m^2}) \tag{4.174}$$

where $4\pi d^2$ is the surface of a sphere of radius d that is uniformly illuminated by the antenna. We observe that the power density decreases with the square of the distance. On a logarithmic scale (dB) this is equivalent to a decrease of 20 dB-per-decade with the distance.

In the case of a *directional* antenna, the power density is concentrated within a cone and is given by

$$\Phi = \mathtt{G}_{Tx}\Phi_0 = \frac{\mathtt{G}_{Tx}\mathtt{P}_{Tx}}{4\pi d^2} \tag{4.175}$$

where $\mathtt{G}_{Tx}$ is the transmit antenna *gain*. Obviously, $\mathtt{G}_{Tx} = 1$ for an isotropic antenna; usually, $\mathtt{G}_{Tx} \gg 1$ for a directional antenna.

At the receive antenna, the *available power* in conditions of matched impedance is given by

$$\mathtt{P}_{Rc} = \Phi \mathtt{A}_{Rc} \eta_{Rc} \tag{4.176}$$

where $\mathtt{P}_{Rc}$ is the received power, $\mathtt{A}_{Rc}$ is the *effective area* of the receive antenna and η_{Rc} is the *efficiency* of the receive antenna. The factor $\eta_{Rc} < 1$ takes into account the fact that the antenna does not capture all the incident radiation, because a part is reflected or lost. To conclude, the power of the received signal is given by

$$\mathtt{P}_{Rc} = \mathtt{P}_{Tx} \frac{\mathtt{A}_{Rc}}{4\pi d^2} \mathtt{G}_{Tx} \eta_{Rc} \tag{4.177}$$

The antenna gain can be expressed as [1]

$$\mathtt{G} = \frac{4\pi \mathtt{A}}{\lambda^2} \eta \tag{4.178}$$

where $\mathtt{A}$ is the effective area of the antenna, $\lambda = c/f_0$ is the wavelength of the transmitted signal, f_0 is the carrier frequency and η is the efficiency factor. The (4.178) holds for the transmit as well as for the receive antenna.

We note that, because of the factor $\mathtt{A}/\lambda^2$, working at higher frequencies presents the advantage of being able to use smaller antennas, for a given $\mathtt{G}$. Usually $\eta \in [0.5, 0.6]$ for *parabolic* antennas, while $\eta \simeq 0.8$ for *horn* antennas.

Observing (4.178), we get

$$\mathtt{P}_{Rc} = \mathtt{P}_{Tx} \mathtt{G}_{Tx} \mathtt{G}_{Rc} \left(\frac{\lambda}{4\pi d} \right)^2 \tag{4.179}$$

The (4.179) is known as the *Friis transmission equation* and is valid in conditions of maximum transfer of power. The term $(\lambda/4\pi d)^2$ is called *free space path loss*. Later, we will use the following definition:

$$\mathtt{P}_0 = \mathtt{P}_{Tx} \mathtt{G}_{Tx} \mathtt{G}_{Rc} \left(\frac{\lambda}{4\pi} \right)^2 \tag{4.180}$$

which represents the power of a signal received at the distance of 1 meter from the transmitter.

In any case, (4.179) does not take into account attenuation due to rain or other environmental factors, nor the possibility that the antennas may not be correctly positioned.

The *available attenuation* of the medium, expressed in dB, is

$$(\mathtt{a}_d)_{dB} = 10 \log_{10} \frac{\mathtt{P}_{Tx}}{\mathtt{P}_{Rc}} = 32.4 + 20 \log_{10} d|_{km} + 20 \log_{10} f_0|_{MHz} - (\mathtt{G}_{Tx})_{dB} - (\mathtt{G}_{Rc})_{dB} \tag{4.181}$$

where $32.4 = 10 \log_{10}(4\pi/c)^2$, d is expressed in km, f_0 in MHz, and $(\mathtt{G}_{Tx})_{dB}$ and $(\mathtt{G}_{Rc})_{dB}$ in dB.

It is worthwhile making the following observations on the attenuation $\mathtt{a}_d$ expressed by (4.181): a) it increases with distance as $\log_{10} d$, whereas for metallic transmission lines the dependency is linear (see (4.140)); b) it increases with frequency as $\log_{10} f_0$.

For $\mathtt{G}_{Tx} = \mathtt{G}_{Rc} = 1$, $(\mathtt{a}_d)_{dB}$ coincides with the free space path loss.

Equivalent circuit at the receiver

We redraw in Figure 4.28 the electrical equivalent circuit at the receiver, using a slightly different notation from that of Figure 4.10. The antenna produces the desired signal s, and w represents the total noise due to the antenna and the amplifier. The amplifier has a bandwidth B around the carrier frequency f_0. The spectral density of the open circuit noise voltage is $\mathcal{P}_w(f) = 2\mathtt{k}\mathtt{T}_w R_i$, and the available noise power per unit of frequency is $\mathtt{p}_w(f) = (\mathtt{k}/2)\mathtt{T}_w$. The effective noise temperature at the input is $\mathtt{T}_w = \mathtt{T}_S + (\mathtt{F} - 1)\mathtt{T}_0$, where $\mathtt{T}_S$ is the effective noise temperature of the antenna, and $\mathtt{T}_A = (\mathtt{F} - 1)\mathtt{T}_0$ is the noise temperature of the amplifier; $\mathtt{T}_0$ is the room temperature and $\mathtt{F}$ is the noise figure of the amplifier.

From (4.92), for matched input and output circuits, the signal-to-noise ratio at the amplifier output is equal to

$$\Lambda = \frac{\mathtt{P}_{Rc}}{\mathtt{k}\mathtt{T}_w B} = \frac{\text{available power of received desired signal}}{\text{available power of effective input noise}} \tag{4.182}$$

We note that there are two noise sources, introduced by the antenna (w_S) and by the receiver (w_A). The noise temperature of the antenna depends on the direction in which the antenna is pointed: for example

$$\mathtt{T}_{S,\text{Sun}} > \mathtt{T}_{S,\text{atmosphere}} \tag{4.183}$$

Multipath

It is useful to study the propagation of a sinusoidal signal hypothesizing that the one-ray model is adequate, which implies using a directional antenna. Let s_{Tx} be a narrowband

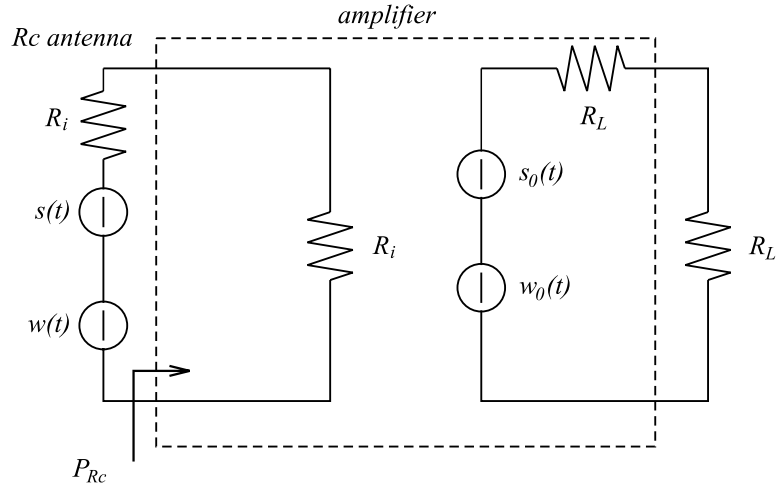


Figure 4.28. Electrical equivalent circuit at the receiver.

transmitted signal, that is

$$s_{Tx}(t) = Re[A_{Tx} e^{j2\pi f_0 t}] \tag{4.184}$$

The received signal at a distance d from the transmitter is given by

$$s_{Rc}(t) = Re[A_{Rc} e^{j2\pi f_0 (t-\tau_1)}] = Re[A_{Rc} e^{j\varphi_{Rc}} e^{j2\pi f_0 t}] \tag{4.185}$$

where $\tau_1 = d/c$ denotes the propagation delay, A_{Rc} is the amplitude of the received signal, and $\varphi_{Rc} = -2\pi f_0 \tau_1 = -2\pi f_0 d/c$ is the phase of the received signal.

Using the definition (1.150) of $h^{(a)}(t)$, the radio channel associated with (4.185) has impulse response

$$g_{Ch}(\tau) = Re\left[\frac{A_{Rc}}{A_{Tx}} h^{(a)}(\tau - \tau_1)\right] \tag{4.186}$$

that is the channel attenuates the signal and introduces a delay equal to τ_1. Choosing f_0 as the carrier frequency, the baseband equivalent of g_{Ch} is given by[7]

$$g_{Ch}^{(bb)}(\tau) = \frac{2A_{Rc}}{A_{Tx}} e^{-j2\pi f_0 \tau_1} \delta(\tau - \tau_1) \tag{4.187}$$

Limited to *signals s_{Tx} of the type* (4.184), (4.186) can be rewritten as

$$g_{Ch}(\tau) = Re\left[\frac{A_{Rc}}{A_{Tx}} e^{j\varphi_{Rc}} h^{(a)}(\tau)\right] \tag{4.188}$$

Thus, (4.188) indicates that the received signal exhibits a phase shift of $\varphi_{Rc} = -2\pi f_0 \tau_1$ with respect to the transmitted signal, because of the propagation delay. As the propagation delay is given by $\tau = d/c$, the delay per unit of distance is equal to 3.3 ns/m.

As the power decreases with the square of the distance between transmitter and receiver, the amplitude of the received signal decreases linearly with the distance, hence $A_{Rc} \propto A_{Tx}/d$; in particular, if A_0 is the amplitude of the received signal at the distance of 1 meter from the transmitter, then $A_{Rc} = A_0/d$, and the power of the received signal is given by $\mathrm{P}_{Rc} = A_{Rc}^2/2$.

Reflection and scattering phenomena imply that the one-ray model is applicable only to propagation in free space, and is not adequate to characterize radio channels, such as for example the channel between a fixed radio station and a mobile receiver.

We will now consider the propagation of a narrowband signal in the presence of reflections. If a ray undergoes a reflection caused by a surface, a part of its power is absorbed by the surface while the rest is re-transmitted in another direction. If the i-th ray has undergone K_i reflections before arriving at the receiver and if a_{ij} is a complex number denoting the reflection coefficient of the j-th reflection of the i-th ray, the total reflection factor is

$$a_i = \prod_{j=1}^{K_i} a_{ij} \tag{4.189}$$

[7] The constraint that $\mathcal{G}_{Ch}^{(bb)}(f) = 0$ for $f < -f_0$ was removed because the input already satisfies the condition $\mathcal{S}_{Tx}^{(bb)}(f) = 0$ for $f < -f_0$.

Therefore signal amplitudes, corresponding to rays that are not the direct or line of sight ray, undergo an attenuation due to reflections that is added to the attenuation due to distance. The total phase shift asociated with each ray is obtained by summing the phase shifts introduced by the various reflections and the phase shift due to the distance traveled. If N_c is the number of paths and d_i is the distance traveled by the i-th ray, extending the channel model (4.186) we get

$$g_{Ch}(\tau) = Re\left[\frac{A_0}{A_{Tx}} \sum_{i=1}^{N_c} \frac{a_i}{d_i} h^{(a)}(\tau - \tau_i)\right] \tag{4.190}$$

where $\tau_i = d_i/c$ is the delay of the i-th ray. The complex envelope of the channel impulse response (4.190) around f_0 is equal to

$$g_{Ch}^{(bb)}(\tau) = \frac{2A_0}{A_{Tx}} \sum_{i=1}^{N_c} \frac{a_i}{d_i} e^{-j2\pi f_0 \tau_i} \delta(\tau - \tau_i) \tag{4.191}$$

We note that the only difference between the passband model and its baseband equivalent is constituted by the additional phase term $e^{-j2\pi f_0 \tau_i}$ for the i-th ray.

Limited to *narrowband signals*, extending the channel model (4.188) to the case of many reflections, the received signal can still be written as

$$s_{Rc}(t) = Re[A_{Rc}\, e^{j\varphi_{Rc}}\, e^{j2\pi f_0 t}] \tag{4.192}$$

where now amplitude and phase are given by

$$A_{Rc}\, e^{j\varphi_{Rc}} = A_0 \sum_{i=1}^{N_c} \frac{a_i}{d_i} e^{j\varphi_i} \tag{4.193}$$

with $\varphi_i = -2\pi f_0 \tau_i$. Let A_i and ψ_i be amplitude and phase, respectively, of the term $A_0(a_i/d_i)e^{j\varphi_i}$; from (4.193) the resulting signal is given by the sum of $A_i e^{j\psi_i}$, $i = 1, \ldots, N_c$, as represented in Figure 4.29. As $\mathtt{P}_0 = A_0^2/2$, the received power is

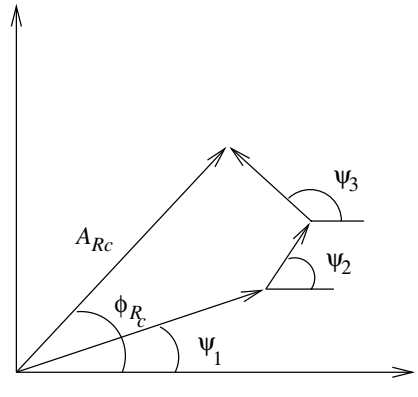


Figure 4.29. Representation of (4.193) in the complex plane.

$$\mathrm{P}_{Rc} = \mathrm{P}_0 \left| \sum_{i=1}^{N_c} \frac{a_i}{d_i} e^{j\varphi_i} \right|^2 \tag{4.194}$$

and is independent of the total phase of the first ray. We will now give two examples of application of the previous results.

Example 4.6.1 (Power attenuation as a function of distance in mobile radio channels)
We consider two antennas, one transmitting and the other receiving, with height respectively h_1 and h_2, that are placed at a distance d. Moreover, it is assumed that $d \gg h_1$ and $d \gg h_2$ (see Figure 4.30).

We consider the case of two paths: one is the straight path (LOS), and the other is reflected by the earth surface with reflection coefficient $a_1 = -1$, i.e. the earth acts as an ideal reflecting surface and does not absorb power. Observing (4.194), and considering that for the above assumptions the lengths of the two paths are both approximately equal to d, the received power is given by

$$\mathrm{P}_{Rc} \simeq \frac{\mathrm{P}_0}{d^2} |1 - e^{j\Delta\varphi}|^2 \tag{4.195}$$

where $\Delta\varphi = 2\pi f_0 \Delta d/c = 2\pi \Delta d/\lambda$ is the phase shift between the two paths, and $\Delta d = 2h_1h_2/d$ is the difference between the lengths of the two paths. For small values of $\Delta\varphi$ we obtain:

$$|1 - e^{j\Delta\varphi}|^2 \simeq |\Delta\varphi|^2 = 16\pi^2 \frac{h_1^2 h_2^2}{\lambda^2 d^2} \tag{4.196}$$

from which, by substituting (4.180) in (4.195), we get

$$\mathrm{P}_{Rc} = \frac{\mathrm{P}_0}{d^2} |\Delta\varphi|^2 = \mathrm{P}_{Tx} \mathrm{G}_{Tx} \mathrm{G}_{Rc} \frac{h_1^2 h_2^2}{d^4} \tag{4.197}$$

We note that the received power decreases as the fourth-power of the distance d, that is 40 dB/decade instead of 20 dB/decade as in the case of free space. Therefore the law of power attenuation as a function of distance changes in the presence of multipath with respect to the case of propagation in free space.

Example 4.6.2 (Fading caused by multipath)
Consider again the previous example, but assume that transmitter and receiver are positioned in a room, so that the inequalities between the antenna heights and the distance d are no

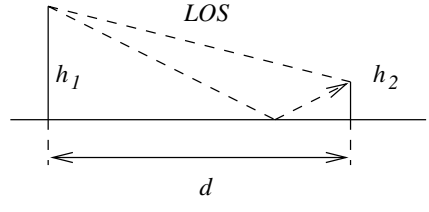


Figure 4.30. Two-ray propagation model.

longer valid. It is assumed, moreover, that the rays that reach the receive antenna are due, respectively, to LOS, reflection from the floor, and reflection from the ceiling. As a result the received power is given by

$$\mathtt{P}_{Rc} = \mathtt{P}_0 \left| \sum_{i=1}^{3} \frac{a_i e^{j\varphi_i}}{d_i} \right|^2 \tag{4.198}$$

where the reflection coefficients are $a_1 = 1$ for the LOS path, and $a_2 = a_3 = -0.7$. With these assumptions, one finds that the power decreases with the distance in an erratic way, in the sense that by varying the position of the antennas the received power presents fluctuations of about $20 \div 30$ dB. In fact, depending on the position, the phases of the various rays change and the sum in (4.193) also varies: in some positions all rays are aligned in phase and the received power is high, whereas in others the rays cancel each other and the received power is low. In the previous example this phenomenon is not observed because the distance d is much larger than the antenna heights, and the phase difference between the two rays remains always small.

4.6.3 Doppler shift

In the presence of relative motion between transmitter and receiver, the frequency of the received signal undergoes a shift with respect to the frequency of the transmitted signal, known as a *Doppler shift*.

We now analyze in detail the Doppler shift. With reference to Figure 4.31, we consider a transmitter radio T_x and a receiver radio that moves with speed v_p from a point P to a point Q. The variation in distance between the transmitter and the receiver is $\Delta\ell = v_p \Delta t \cos\theta$, where v_p is the speed of the receiver relative to the transmitter, Δt is the time required for the receiver to go from P to Q, and θ is the angle of incidence of the signal with respect to the direction of motion (θ is assumed to be the same in P and in Q). The phase variation of the received signal because of the different path length in P and Q is

$$\Delta\varphi = \frac{2\pi\,\Delta\ell}{\lambda} = \frac{2\pi\, v_p \Delta t}{\lambda} \cos\theta \tag{4.199}$$

and hence the apparent change in frequency or Doppler shift is

$$f_s = \frac{1}{2\pi} \frac{\Delta\varphi}{\Delta t} = \frac{v_p}{\lambda} \cos\theta \tag{4.200}$$

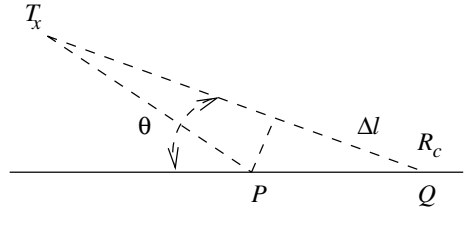


Figure 4.31. Illustration of the Doppler shift.

This implies that if a narrowband signal given by (4.184) is transmitted, the received signal is

$$s_{Rc}(t) = Re[A_{Rc}e^{j2\pi(f_0 - f_s)t}] \tag{4.201}$$

The (4.200) relates the Doppler shift to the speed of the receiver and the angle θ; in particular, for $\theta = 0$ we get

$$f_s = 9.259\ 10^{-4}\ v_p|_{km/h}\ f_0|_{MHz}\ (\text{Hz}) \tag{4.202}$$

where $v_p\big|_{km/h}$ is the speed of the mobile in km/h, and $f_0|_{MHz}$ is the carrier frequency in MHz. For example, if $v_p = 100$ km/h and $f_0 = 900$ MHz we have $f_s = 83$ Hz. We note that if the receiver moves towards the transmitter the Doppler shift is positive, if it moves away from the transmitter the Doppler shift is negative.

We now consider a narrowband signal transmitted in an *indoor* environment[8] where the signal received by the antenna is given by the contribution of many rays, each with a different length. If the signal propagation were taking place through only one ray, the received signal would undergo only one Doppler shift. But according to (4.200) the frequency shift f_s depends on the angle θ. Therefore, because of the different paths, the received signal is no longer monochromatic, and we speak of a *Doppler spectrum* to indicate the spectrum of the received signal around f_0. This phenomenon manifests itself also if both the transmitter and the receiver are static, but a person or an object moves modifying the signal propagation.

The Doppler spectrum is characterized by the *Doppler spread*, which measures the dispersion in the frequency domain that is experienced by a transmitted sinusoidal signal. It is intuitive that the more the characteristics of the radio channel vary with time, the larger the Doppler spread will be. An important consequence of this observation is that the convergence time of algorithms used in receivers, e.g., to perform adaptive equalization, must be much smaller than the inverse of the Doppler spread of the channel, thus enabling the adaptive algorithms to follow the channel variations.

Example 4.6.3 (Doppler shift)
Consider a transmitter that radiates a sinusoidal carrier at the frequency of $f_0 = 1850$ MHz. For a vehicle traveling at 96.55 km/h (26.82 m/s), we want to evaluate the frequency of the received carrier if the vehicle is moving: a) approaching the transmitter, b) going away from the transmitter, c) perpendicular to the direction of arrival of the transmitted signal.

The wavelength is

$$\lambda = \frac{c}{f_0} = \frac{3 \times 10^8}{1850 \times 10^6} = 0.162\,\text{m} \tag{4.203}$$

a) The Doppler shift is positive; the received frequency is

$$f_{Rc} = f_0 + f_s = 1850 \times 10^6 + \frac{26.82}{0.162} = 1850.000166\,\text{MHz} \tag{4.204}$$

[8] The term *indoor* is usually referred to areas inside buildings, possibly separated by walls of various thickness, material, and height. The term *outdoor*, instead, is usually referred to areas outside of buildings: these environments can be of various types, for example, urban, suburban, rural, etc.

b) The Doppler shift is negative; the received frequency is

$$f_{Rc} = f_0 - f_s = 1850 \times 10^6 - \frac{26.82}{0.162} = 1849.999834\,\text{MHz} \tag{4.205}$$

c) In this case $\cos(\theta) = 0$; therefore there is no Doppler shift.

4.6.4 Propagation of wideband signals

For a wideband signal with spectrum centered around the carrier frequency f_0, the channel model (4.191) is still valid; we rewrite the channel impulse response as a function of both the time variable t and the delay τ for a given t:

$$g_{Ch}^{(bb)}(t, \tau) = \sum_{i=1}^{N_c} g_i(t)\delta(\tau - \tau_i(t)) \tag{4.206}$$

where g_i represents the complex-valued gain of the i-th ray that arrives with delay τ_i. For a given receiver location, (4.206) models the channel as a linear filter having time-varying impulse response, where the channel variability is due to the motion of transmitter and/or receiver, or to changes in the surrounding environment, or to both factors.

If the channel is time-invariant, or at least it is time-invariant within a short time interval, in this time interval the impulse response is only a function of τ.

The transmitted signal undergoes three phenomena: a) *fading* of some gains g_i due to multipath, which implies rapid changes of the received signal power over short distances (of the order of the carrier wavelength) and brief time intervals, b) *time dispersion* of the impulse response caused by diverse propagation delays of multipath rays, c) *Doppler shift*, which introduces a random frequency modulation that is in general different for different rays.

In a digital transmission system the effect of multipath depends on the relative duration of the symbol period and the channel impulse response. If the duration of the channel impulse response is very small with respect to the duration of the symbol period, i.e. the transmitted signal is narrowband with respect to the channel, then the one-ray model is a suitable channel model; if the gain of the single ray varies in time we speak of a *flat fading* channel. Otherwise, an adequate model must include several rays: in this case if the gains vary in time we speak of a *frequency selective fading* channel.

Neglecting the absolute delay $\tau_1(t)$, letting $\tilde{\tau}_2 = \tau_2 - \tau_1$ a simple two-ray radio channel model has impulse response

$$g_{Ch}^{(bb)}(t, \tau) = g_1(t)\delta(\tau) + g_2(t)\delta(\tau - \tilde{\tau}_2(t)) \tag{4.207}$$

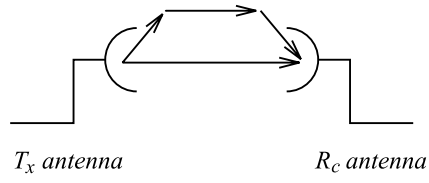
At a given instant t, the channel is equivalent to a filter with impulse response illustrated in Figure 4.32 and frequency response given by:[9]

$$\mathcal{G}_{Ch}^{(bb)}(t, f) = g_1(t) + g_2(t)e^{-j2\pi f \tilde{\tau}_2(t)} \tag{4.209}$$

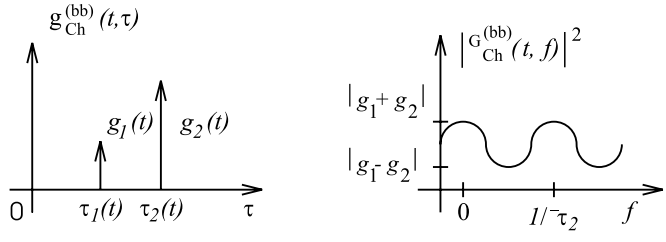
[9] If we normalize the coefficients with respect to g_1, (4.209) becomes

$$\mathcal{G}_{Ch}^{(bb)}(f) = 1 + b\, e^{-j2\pi f \tau} \tag{4.208}$$

where b is a complex number. In the literature (4.208) is called *Rummler model* of the radio channel.



(a)



(b)

Figure 4.32. Physical representation and model of a two-ray radio channel, where g_1 and g_2 are assumed to be positive.

It is evident that the channel has a selective frequency behavior, as the attenuation depends on frequency. For g_1 and g_2 real-valued, from (4.209) the following frequency response is obtained

$$\left|\mathcal{G}_{Ch}^{(bb)}(t,\,f)\right|^2 = g_1^2(t) + g_2^2(t) + 2g_1(t)g_2(t)\cos(2\pi f\tilde{\tau}_2(t)) \tag{4.210}$$

shown in Figure 4.32. In any case, the signal distortion depends on the signal bandwidth in comparison to $1/\tilde{\tau}_2$.

Going back to the general case, for wideband communications, rays with different delays are assumed to be independent, that is they do not interact with each other. In this case from (4.206) the received power is

$$\mathrm{P}_{Rc} = \mathrm{P}_{Tx}\sum_{i=1}^{N_c}|g_i|^2 \tag{4.211}$$

From (4.211) we note that the received power is given by the sum of the squared amplitude of all the rays. Conversely, in the transmission of narrowband signals the received power is the square of the vector amplitude resulting from the vector sum of all the received rays. Therefore, for a given transmitted power, the received power will be lower for a narrowband signal as compared to a wideband signal.

Channel parameters in the presence of multipath

To study the performance of mobile radio systems it is convenient to introduce a measure of the channel dispersion in the time domain known as *multipath delay spread* (MDS). The MDS is the measure of the time interval that elapses between the arrival of the first and the last ray; the simplest measure is the delay time that it takes for the amplitude of the ray to decrease by x dB below the maximum value; this time is also called *excess delay spread* (EDS). However, the EDS is not a very meaningful parameter, because channels that exhibit considerably different distributions of the gains g_i may have the same value of EDS. A parameter that is normally used to define conveniently the MDS of the channel is the *root-mean square* (rms) *delay spread*, τ_{rms}, which corresponds to the second-order central moment of the channel impulse response, that is

$$\tau_{rms} = \sqrt{\overline{\tau^2} - \overline{\tau}^2} \tag{4.212}$$

where

$$\overline{\tau^n} = \frac{\sum_{i=1}^{N_c} |g_i|^2 \tau_i^n}{\sum_{i=1}^{N_c} |g_i|^2} \qquad n = 1, 2 \tag{4.213}$$

The above formulae give the rms delay spread for an instantaneous channel impulse response. With reference to the time-varying characteristics of the channels, we use the (*average*) rms delay spread $\overline{\tau}_{rms}$ obtained by substituting in (4.213) in place of $|g_i|^2$ its expectation. In this case $\overline{\tau}_{rms}$ measures the *mean* time dispersion that a signal undergoes because of multipath.

Typical values of (average) rms delay spread are of the order of μs in *outdoor* mobile radio channels, and of the order of some tenths of *ns* in *indoor* channels.

We define as *power delay profile*, also called *delay power spectrum* or *multipath intensity profile*, the expectation of the squared amplitude of the channel impulse response, $E[|g_i|^2]$, as a function of delay τ_i. In Table 4.5 power delay profiles are given for some typical channels.

Statistical description of fading channels

The most widely used statistical description of the gains $\{g_i\}$ is given by

$$\begin{aligned} g_1 &= C + \tilde{g}_1 \qquad i = 1 \\ g_i &= \tilde{g}_i \qquad\quad i = 2, \ldots, N_c \end{aligned} \tag{4.214}$$

where C is a real-valued constant and $\tilde{g}_i$ is a complex-valued random variable with zero mean and Gaussian distribution (see Example 1.9.3 on page 67). In other words, whereas the first ray contains a direct (deterministic) component in addition to a random component, all the other rays are assumed to have only a random component: therefore the distribution

Table 4.5 Values of $E[|g_i|^2]$ (in dB) and τ_i (in ns) for three typical channels.

Standard GSM		Indoor offices		Indoor business	
τ_i	$E[\|g_i\|^2]$	τ_i	$E[\|g_i\|^2]$	τ_i	$E[\|g_i\|^2]$
0	−3.0	0	0.0	0	−4.6
200	0	50	−1.6	50	0
500	−2.0	150	−4.7	150	−4.3
1600	−6.0	325	−10.1	225	−6.5
2300	−8.0	550	−17.1	400	−3.0
5000	−10.0	700	−21.7	525	−15.2
				750	−21.7

of $|g_i|$ will be a Rice distribution for $|g_1|$ and a Rayleigh distribution for $|g_i|$, $i \neq 1$. In particular, letting $\bar{g}_i = g_i/\sqrt{E[|g_i|^2]}$, we have

$$\begin{aligned} p_{|\bar{g}_1|}(a) &= 2(1+K)\,a \exp[-K-(1+K)a^2] I_0[2a\sqrt{K(1+K)}]1(a) \\ p_{|\bar{g}_i|}(a) &= 2a\,e^{-a^2}1(a) \end{aligned} \tag{4.215}$$

where I_0 is the *modified Bessel function of the first type and order zero*,

$$I_0(x) = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{x\cos\alpha}\,d\alpha \tag{4.216}$$

The probability density (4.215) is given in Figure 4.33 for various values of K.

In (4.214) the phase of $\tilde{g}_i$ is uniformly distributed in $[0, 2\pi)$. For a one-ray channel model, the parameter $K = C^2/E[|\tilde{g}_1|^2]$, known as the *Rice factor*, is equal to the ratio between the power of the direct component and the power of the reflected and/or scattered component. In general for a model with more rays we take $K = C^2/\mathtt{M}_d$, where $\mathtt{M}_d$ is the statistical power of all reflected and/or scattered components, that is $\mathtt{M}_d = \sum_{i=1}^{N_c} E[|\tilde{g}_i|^2]$. Assuming that the power delay profile is normalized such that

$$\sum_{i=1}^{N_c} E[|g_i|^2] = 1 \tag{4.217}$$

we obtain $C = \sqrt{K/(K+1)}$. Typical reference values for K are 3 and 10 dB. If $C = 0$, i.e. no direct component exists, it is $K = 0$, and the Rayleigh distribution is obtained for all the gains $\{g_i\}$. For $K \to \infty$, i.e. with no reflected and/or scattered components and, hence, $C = 1$, we find the model having only the deterministic component.

To justify the Rice model for $|g_1|$ we consider the transmission of a sinusoidal signal (4.184). In this case the expression of the received signal is given by (4.192), which we rewrite as follows:

$$s_{Rc}(t) = [\tilde{g}_{1,I}(t) + C]\cos 2\pi f_0 t - \tilde{g}_{1,Q}(t)\sin 2\pi f_0 t \tag{4.218}$$

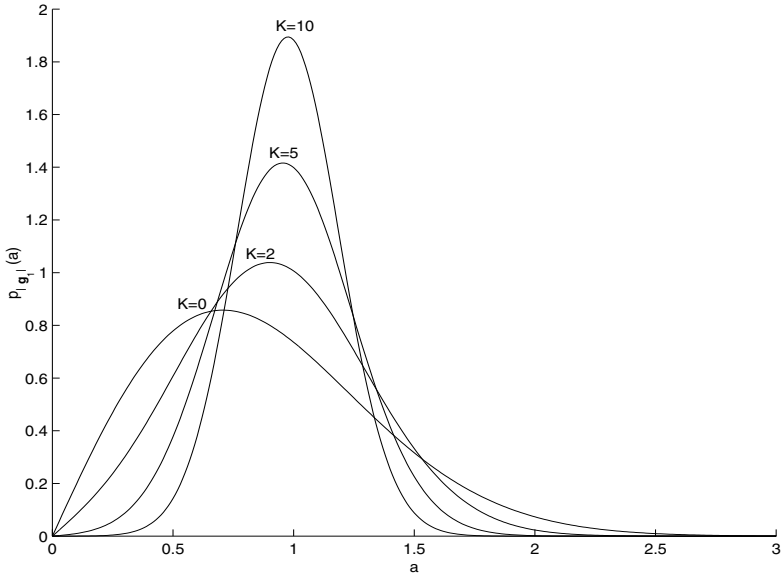


Figure 4.33. The Rice probability density function for various values of K. The Rayleigh density function is obtained for $K = 0$.

where C represents the contribution of the possible direct component of the propagation signal, and $\tilde{g}_{1,I}$ and $\tilde{g}_{1,Q}$ are due to the scattered component. As the gains $\tilde{g}_{1,I}$ and $\tilde{g}_{1,Q}$ are given by the sum of a large number of random components, they can be approximated by independent Gaussian random processes with zero mean. The instantaneous envelope of the received signal is then given by

$$\sqrt{[\tilde{g}_{1,I}(t) + C]^2 + \tilde{g}_{1,Q}^2(t)} \tag{4.219}$$

which, in the assumption just formulated, is a Rice random variable for each instant t.

4.6.5 Continuous-time channel model

The channel model previously studied is especially useful for system simulations, as will be discussed later. A general continuous-time model is now presented.

Assuming that the signal propagation occurs through a large number of paths, which in turn are subject to a very large number of random phenomena, the (baseband equivalent) channel impulse response can be represented with good approximation as a time-varying complex-valued Gaussian random process $g(t, \tau)$. In particular $g(t, \tau)$ represents the channel output at the instant t in response to an impulse applied at the instant $(t - \tau)$. We now evaluate the autocorrelation function of the impulse response evaluated at two different instants and two different delays,

$$\varkappa_g(t, t - \Delta t; \tau, \tau - \Delta\tau) = E[g(t, \tau)g^*(t - \Delta t, \tau - \Delta\tau)] \tag{4.220}$$

According to the model known as the *wide-sense stationary uncorrelated scattering* (WSSUS), the values of g for rays that arrive with different delays are uncorrelated, and g is stationary in t. Therefore we have

$$\mathtt{r}_g(t, t-\Delta t;\tau, \tau-\Delta\tau) = \mathtt{r}_g(\Delta t;\tau)\delta(\Delta\tau) \tag{4.221}$$

In other words, the autocorrelation is non-zero only for impulse responses that are considered for the same delay time. Moreover, as g is stationary in t, if the delay time is the same, the autocorrelation only depends on the difference of the times at which the two impulse responses are evaluated.

Power delay profile

For $\Delta t = 0$ we define the function $\mathtt{M}(\tau) = E[|g(t,\tau)|^2]$, that is called channel *power delay profile* and represents the statistical power of the gain $g(t,\tau)$ for a given delay τ. For a Rayleigh channel model, three typical curves are now given for $\mathtt{M}(\tau)$, where $\overline{\tau}_{rms}$ is the parameter defined in (4.225):

1. *Two rays, with equal power*

$$\mathtt{M}(\tau) = \tfrac{1}{2}\delta(\tau) + \tfrac{1}{2}\delta(\tau - 2\overline{\tau}_{rms}) \tag{4.222}$$

2. *Gaussian, unilateral*

$$\mathtt{M}(\tau) = \sqrt{\frac{2}{\pi}}\frac{1}{\overline{\tau}_{rms}} e^{-\tau^2/(2\overline{\tau}^2_{rms})} \qquad \tau \geq 0 \tag{4.223}$$

3. *Exponential, unilateral*

$$\mathtt{M}(\tau) = \frac{1}{\overline{\tau}_{rms}} e^{-\tau/(\overline{\tau}_{rms})} \qquad \tau \geq 0 \tag{4.224}$$

The measure of the set of values τ for which $\mathtt{M}(\tau)$ is above a certain threshold is called (*average*) *excess delay spread* of the channel. As in the case of the discrete channel model previously studied, we define the (*average*) *rms delay spread* as

$$\overline{\tau}^2_{rms} = \frac{\displaystyle\int_{-\infty}^{\infty} (\tau - \overline{\tau})^2 \mathtt{M}(\tau)\, d\tau}{\displaystyle\int_{-\infty}^{\infty} \mathtt{M}(\tau)\, d\tau} \tag{4.225}$$

where

$$\overline{\tau} = \int_{-\infty}^{\infty} \tau \mathtt{M}(\tau)\, d\tau \tag{4.226}$$

The inverse of the (average) rms delay spread is called the *coherence bandwidth* of the channel.

For digital transmission over such channels, we observe that if $\overline{\tau}_{rms}$ is of the order of 20% of the symbol period, or larger then signal distortion is non-negligible. Equivalently, if the *coherence bandwidth* of the channel is lower than 5 times the modulation rate of the transmission system, then we speak of a *frequency selective fading* channel, otherwise the channel is *flat fading*. However, in the presence of *flat fading* the received signal may vanish completely, whereas *frequency selective fading* produces several replicas of the transmitted signal at the receiver, so that a suitably designed receiver can recover the transmitted information.

Example 4.6.4 (Power delay profile)
We compute the average rms delay spread for the multipath delay profile of Figure 4.34, and determine the *coherence bandwidth*, defined as $B_c = \frac{1}{5\bar{\tau}_{rms}}$.

From (4.213) we have

$$\bar{\tau} = \frac{(1)(5) + (0.1)(1) + (0.1)(2) + (0.01)(0)}{0.01 + 0.1 + 0.1 + 1} = 4.38 \ \mu \text{ s} \tag{4.227}$$

and

$$\bar{\tau^2} = \frac{(1)(5)^2 + (0.1)(1)^2 + (0.1)(2)^2 + (0.01)(0)}{0.01 + 0.1 + 0.1 + 1} = 21.07 \, (\mu \text{ s})^2 \tag{4.228}$$

Therefore we get

$$\bar{\tau}_{rms} = \sqrt{21.07 - (4.38)^2} = 1.37 \ \mu \text{ s} \tag{4.229}$$

Consequently the coherence bandwidth of the channel is equal to $B_c = 146\,\text{kHz}$.

Doppler spectrum

We now analyze the WSSUS channel model with reference to time variations. First we introduce the correlation function of the channel frequency response taken at instants t and $t - \Delta t$, and, respectively, at frequencies f and $f - \Delta f$,

$$\mathrm{r}_G(t, t - \Delta t; f, f - \Delta f) = E[G(t, f)G^*(t - \Delta t, f - \Delta f)] \tag{4.230}$$

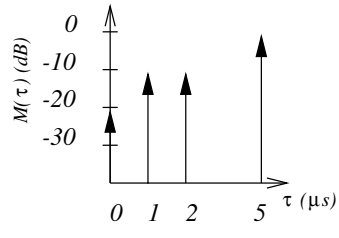


Figure 4.34. Multipath delay profile.

Substituting in (4.230) the relation

$$G(t,\, f) = \int_{-\infty}^{+\infty} g(t,\, \tau)\, e^{-j2\pi f \tau}\, d\tau \tag{4.231}$$

we find that $\mathtt{r}_G$ depends only on Δt and Δf; moreover it holds

$$\mathtt{r}_G(\Delta t;\Delta f) = \int_{-\infty}^{+\infty} \mathtt{r}_g(\Delta t;\tau)\, e^{-j2\pi(\Delta f)\tau}\, d\tau \tag{4.232}$$

that is $\mathtt{r}_G(\Delta t;\Delta f)$ is the Fourier transform of $\mathtt{r}_g(\Delta t;\tau)$. The Fourier transform of $\mathtt{r}_G$ is given by

$$\mathcal{P}_G(\lambda,\, \Delta f) = \int_{-\infty}^{\infty} \mathtt{r}_G(\Delta t;\Delta f) e^{-j2\pi\lambda(\Delta t)} d(\Delta t) \tag{4.233}$$

The time variation of the frequency response is measured by $\mathcal{P}_G(\lambda, 0)$.

Now we introduce the *Doppler spectrum* $\mathcal{D}(\lambda)$, which represents the power of the Doppler shift for different values of the frequency λ. We recall that the Doppler shift is caused by the motion of terminals or surrounding objects. We define $\mathcal{D}(\lambda)$ as the Fourier transform of the autocorrelation function of the impulse response, in correspondence of the same delay τ, evaluated at two different instants, that is:[10]

$$\mathcal{D}(\lambda) = \int_{-\infty}^{+\infty} \frac{\mathtt{r}_g(\Delta t;\tau)}{\mathtt{r}_g(0;\tau)}\, e^{-j2\pi\lambda\Delta t}\, d(\Delta t) \tag{4.234}$$

The term $\mathtt{r}_g(0;\tau)$ in (4.234) represents a normalization factor such that

$$\int_{-\infty}^{+\infty} \mathcal{D}(\lambda)\, d\lambda = 1 \tag{4.235}$$

We note that (4.234) implies that $\mathtt{r}_g(\Delta t;\tau)$ is a separable function,

$$\mathtt{r}_g(\Delta t;\tau) = \mathrm{d}(\Delta t) \cdot \mathtt{r}_g(0;\tau) = \mathrm{d}(\Delta t)\, \mathtt{M}(\tau) \tag{4.236}$$

where $\mathrm{d}(\Delta t) = \mathcal{F}^{-1}\,[\mathcal{D}(\lambda)]$, with

$$\mathrm{d}(0) = 1 \tag{4.237}$$

and $\mathtt{M}(\tau)$ is the power delay profile, so that

$$\int_{-\infty}^{+\infty} \mathtt{M}(\tau)\, d\tau = 1 \tag{4.238}$$

With the above assumptions the following equality holds:

$$\mathcal{D}(\lambda) = \mathcal{P}_G(\lambda;0) \tag{4.239}$$

[10] In very general terms, we could have a different Doppler spectrum for each path, or gain $g(t,\, \tau)$, of the channel.

Therefore, $\mathcal{D}(\lambda)$ can also be obtained as the Fourier transform of $\mathtt{r}_G(\Delta t;0)$, which in turn can be determined by transmitting a sinusoidal signal (hence $\Delta f = 0$) and estimating the autocorrelation function of the amplitude of the received signal.

The maximum frequency f_d of the Doppler spectrum support is called the *Doppler spread* of the channel and gives a measure of the *fading rate* of the channel. Another measure of the support of $\mathcal{D}(\lambda)$ can be obtained through the *rms Doppler spread* or second order central moment of the Doppler spectrum. The inverse of the Doppler spread is called *coherence time*: it gives a measure of the time interval within which a channel can be assumed to be time invariant or static. Let T be the symbol period in a digital transmission system; we usually say that the channel is *fast fading* if $f_d T > 10^{-2}$, and *slow fading* if $f_d T < 10^{-3}$.

Doppler spectrum models

A widely used model, known as the *Jakes model* or *classical Doppler spectrum*, to represent the Doppler spectrum is due to Clarke. If f_d denotes the Doppler spread, then

$$\mathcal{D}(f) = \begin{cases} \dfrac{1}{\pi f_d} \dfrac{1}{\sqrt{1-(f/f_d)^2}} & |f| \le f_d \\ 0 & \text{otherwise} \end{cases} \tag{4.240}$$

For the channel model (4.206), the corresponding autocorrelation function of the channel impulse response is given by

$$\mathtt{r}_g(\Delta t;\tau) = \sum_{i=1}^{N_c} J_0(2\pi f_d \Delta t)\, \mathtt{M}(\tau_i)\, \delta(\tau - \tau_i) \tag{4.241}$$

where J_0 is the Bessel function of the first type and order zero. The model of the Doppler spectrum described above agrees with the experimental results obtained for mobile radio channels.

For *indoor* radio channels, thanks to the study conducted by a special commission (JTC), it was demonstrated that the Doppler spectrum can be modelled as

$$\mathcal{D}(f) = \begin{cases} \dfrac{1}{2f_d} & |f| \le f_d \\ 0 & \text{elsewhere} \end{cases} \tag{4.242}$$

with a corresponding autocorrelation function given by

$$\mathtt{r}_g(\Delta t;\tau) = \sum_{i=1}^{N_c} \operatorname{sinc}(2 f_d \Delta t)\, \mathtt{M}(\tau_i)\, \delta(\tau - \tau_i) \tag{4.243}$$

A further model assumes that the Doppler spectrum is described by a second or third-order Butterworth filter with the 3 dB cutoff frequency equal to f_d.

Shadowing

The simplest relation between average transmitted power and average received power is

$$\mathtt{P}_{Rc} = \frac{\mathtt{P}_0}{d^{\alpha}} \tag{4.244}$$

where α is equal to 2 for propagation in free space and to 4 for the simple 2-ray model described before. For indoor and urban outdoor radio channels the relation depends on the environment, according to the number of buildings, their dimensions, and also the material used for their construction; in general, however, variations of the average received power are lower in *outdoor* environments than in *indoor* environments.

Shadowing takes into account the fact that the average received power may present fluctuations around the value obtained by deterministic models. These fluctuations are modelled as a log-normal random variable, that is e^{ξ}, where ξ is a Gaussian random variable with zero mean and variance σ_{ξ}^2. If P_{Rc} is the average received power obtained by deterministic rules, in the presence of *shadowing* it becomes $e^{\xi}\,\mathrm{P}_{Rc}$; in practice *shadowing* provides a measure of the adequacy of the adopted deterministic model.

A propagation model that completely ignores any information on land configuration, and therefore is based only on the distance between transmitter and receiver, has a *shadowing* with $\sigma_{(\xi)_{dB}} = 12\,\mathrm{dB}$. The relation between σ_{ξ} and $\sigma_{(\xi)_{dB}}$ is $\sigma_{\xi} = 0.23\sigma_{(\xi)_{dB}}$. Improving the accuracy of the propagation model, for example, by using more details regarding the environmental configuration, the *shadowing* can be reduced; in case we had an enormous amount of topographic data and the means to elaborate them, we would have a model with $\sigma_{\xi} = 0$. Hence, *shadowing* should be considered in the performance evaluation of mobile radio systems, whereas for the correct design of a network it is good practice to make use of the largest possible quantity of topographic data.

Final remarks

A signal that propagates in a radio channel for mobile communications undergoes a type of fading that depends on the signal as well as on the channel characteristics. In particular, whereas the delay spread due to multipath leads to dispersion in the time domain and therefore frequency selective fading, the Doppler spread causes dispersion in the domain of the variable λ and therefore time selective fading.

The first type of fading can be divided into flat fading and frequency selective fading. In the first case the channel has a constant gain; in other words, the inverse of the transmitted signal bandwidth is much larger than the delay spread of the channel and $g(t, \tau)$ can be approximated by a delta function, with random amplitude and phase, centered at $\tau = 0$. In the second case instead the channel has a time-varying frequency response within the passband of the transmitted signal and consequently the signal undergoes frequency selective fading; these conditions occur when the inverse of the transmitted signal bandwidth is of the same order or smaller than the delay spread of the channel. The received signal consists of several attenuated and delayed versions of the transmitted signal.

A channel can be fast fading or slow fading. In a fast fading channel, the impulse response of the channel changes within a symbol period, that is the coherence time of the channel is smaller than the symbol period; this condition leads to signal distortion, which increases with increasing Doppler spread. Usually there are no remedies to compensate for such distortion unless the symbol period is decreased; on the other hand, this choice leads to larger intersymbol interference. In a slow fading channel, the impulse response changes much more slowly with respect to the symbol period. In general, the channel can be assumed as time invariant for a time interval that is proportional to the inverse of the Doppler spread.

4.6.6 Discrete-time model for fading channels

Our aim is to approximate a transmission channel defined in the continuous-time domain by a channel in the discrete-time domain characterized by sampling period T_Q. We immediately notice that the various delays in (4.206) must be multiples of T_Q and consequently we need to approximate the delays of the power delay profile (see, e.g., Table 4.5). Starting from a continuous-time model of the power delay profile (see, e.g., (4.224)), we need to obtain a sampled version of $\mathtt{M}(\tau)$. The discrete-time model of the radio channel is represented, as illustrated in Figure 4.35, by a time-varying linear filter where the coefficient g_i corresponds to the complex gain of the ray with delay iT_Q, $i = 0, 1, \ldots, N_c - 1$; in the case of flat fading we choose $N_c = 1$.

If the channel is time invariant ($f_d = 0$), all coefficients $\{g_i\}$, $i = 0, \ldots, N_c - 1$, are constant, and are obtained as realizations of N_c random variables. In general, however, $\{g_i\}$ are random processes. To generate each process $g_i(kT_Q)$, the scheme of Figure 4.36 is used, where $\bar{w}_i(\ell T_P)$ is complex-valued Gaussian white noise with zero mean and unit variance, h_{ds} is a narrrowband filter that produces a signal g_i' with the desired Doppler spectrum, and h_{int} is an interpolator filter (see Section 1.A.7). Usually we choose $f_d T_Q \ll 1$, and $1/10 \leq f_d T_P \leq 1/5$. The interpolator output signal is then multiplied by a constant $\sigma_i = \sqrt{\mathtt{M}(iT_Q)}$, which imposes the desired power delay profile.

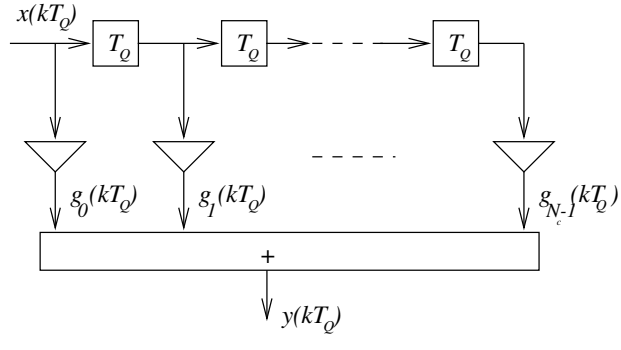


Figure 4.35. Discrete time model of a radio channel.

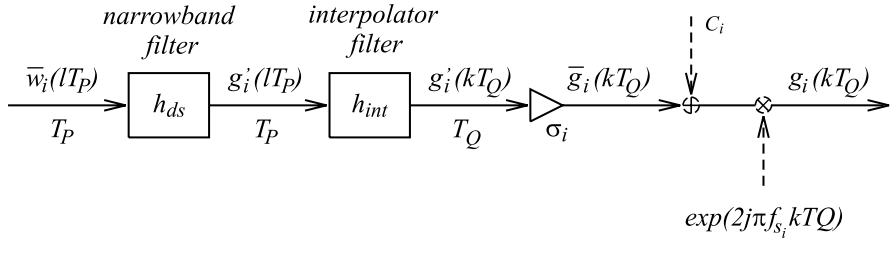


Figure 4.36. Model to generate the *i*-th coefficient of a time-varying channel.

If the channel model includes a deterministic component for the ray with delay $\tau_i = iT_Q$, a constant C_i must be added to the random component $\bar{g}_i$. Furthermore, if the channel model includes a Doppler shift f_{s_i} for the i-th branch, then we need to multiply the term $C_i + \bar{g}_i$ by the exponential function $\exp(j2\pi f_{s_i} kT_Q)$. Observing (4.211), to avoid modifying the average transmitted power, the coefficients $\{g_i\}$, $i = 0, 1, \ldots, N_c - 1$, are scaled, so that

$$\sum_{i=0}^{N_c-1} E[|g_i(kT_Q)|^2] = 1 \tag{4.245}$$

For example, the above condition is satisfied if each signal g_i' has unit statistical power[11] and $\{\sigma_i\}$ satisfy the condition

$$\sum_{i=0}^{N_c-1} (\sigma_i^2 + C_i^2) = 1 \tag{4.246}$$

Generation of a process with a pre-assigned spectrum

The procedure can be generalized for a signal g_i' with a generic Doppler spectrum of the type (4.240) or (4.242) in two ways: 1) implement a filter h_{ds} such that $|\mathcal{H}_{ds}(f)|^2 = \mathcal{D}(f)$, 2) generate a set of N_f (at least 10) complex sinusoids with frequencies $\{\pm f_m\}$, $m = 1, \ldots, N_f$, in the range from $-f_d$ to f_d.

We analyze the two methods.

1) *Using a filter.* We give the description of h_{ds} for two cases

1.a) *Second-order Butterworth filter*. Given $\omega_d = 2\pi f_d$, where f_d is the Doppler spread, the transfer function of the discrete-time filter is

$$H_{ds}(z) = \frac{c_0(1 + z^{-1})^2}{\left(1 + \sum_{n=1}^{2} \mathtt{a}_n z^{-n}\right)} \tag{4.247}$$

where, defining

$$\omega_0 = \tan(\omega_d T_P/2)$$

where T_P is the sampling period, we have [18]

$$\mathtt{a}_1 = -\frac{2(1 - \omega_0^2)}{1 + \omega_0^2 + \sqrt{2}\,\omega_0} \tag{4.248}$$

$$\mathtt{a}_2 = \frac{1 + \omega_0^4}{(1 + \omega_0^2 + \sqrt{2}\,\omega_0)^2} \tag{4.249}$$

$$c_0 = \tfrac{1}{4}\,(1 + \mathtt{a}_1 + \mathtt{a}_2) \tag{4.250}$$

[11] Based on the Example 1.9.10 on page 72, it is $\mathtt{M}_{g_i'} = 1$ if $\mathtt{M}_{\bar{w}_i} = 1$; the equivalent interpolator filter, given by the cascade of h_{ds} and h_{int}, has energy equal to the interpolation factor T_P/T_Q.

The filter output gives

$$g_i'(\ell T_P) = -\mathtt{a}_1\, g_i'((\ell-1)T_P) - \mathtt{a}_2\, g_i'((\ell-2)T_P) + c_0(\tilde{w}_i(\ell T_P) + 2\tilde{w}_i((\ell-1)T_P) + \tilde{w}_i((\ell-2)T_P)) \tag{4.251}$$

1.b) *IIR filter with classical Doppler spectrum.* Now h_{ds} is implemented as the cascade of two filters. The first, $H_{ds_1}(z)$, is an FIR shaping filter with amplitude characteristic of the frequency response given by the square root of the function in (4.240). The second, $H_{ds_2}(z)$, is a Chebychev lowpass filter, with cutoff frequency f_d. Table 4.6 reports values of the overall filter parameters for $f_d T_P = 0.1$ [19].

2) *Using sinusoidal signals.* Let

$$g_i'(\ell T_P) = \sum_{m=1}^{N_f} A_{i,m}[e^{j(2\pi f_m \ell T_P + \varphi_{i,m})} e^{j\Phi_{i,I}} + e^{-j(2\pi f_m \ell T_P + \varphi_{i,m})} e^{-j\Phi_{i,Q}}] \tag{4.252}$$

The spacing between the different frequencies is Δf_m; letting $f_1 = \Delta f_1/2$, for $m > 1$ we have $f_m = f_{m-1} + \Delta f_m$. Each Δf_m can be chosen as a constant,

$$\Delta f_m = \frac{f_d}{N_f} \tag{4.253}$$

or, defining $K_d = \int_0^{f_d} \mathcal{D}^{1/3}(f)df$, as

$$\Delta f_m = \frac{K_d}{N_f \mathcal{D}^{1/3}(f_m)} \qquad m = 1, \ldots, N_f \tag{4.254}$$

Table 4.6 Parameters of an IIR filter which implements a classical Doppler spectrum. [*From Anastasopoulos and Chugg (1997). © 1997 IEEE.*]

$H_{ds}(z) = B(z)/A(z)$ $\quad f_d T_P = 0.1$			
$\{\mathtt{a}_n\},\ n = 0, \ldots, 11$:			
1.0000 e + 0	−4.4153 e + 0	8.6283 e + 0	−9.4592 e + 0
6.1051 e + 0	−1.3542 e + 0	−3.3622 e + 0	7.2390 e + 0
−7.9361 e + 0	5.1221 e + 0	−1.8401 e + 0	2.8706 e − 1
$\{\mathtt{b}_n\},\ n = 0, \ldots, 21$:			
1.3651 e − 4	8.1905 e − 4	2.0476 e − 3	2.7302 e − 3
2.0476 e − 3	9.0939 e − 4	6.7852 e − 4	1.3550 e − 3
1.8067 e − 3	1.3550 e − 3	5.3726 e − 4	6.1818 e − 5
−7.1294 e − 5	−9.5058 e − 5	−7.1294 e − 5	−2.5505 e − 5
1.3321 e − 5	4.5186 e − 5	6.0248 e − 5	4.5186 e − 5
1.8074 e − 5	3.0124 e − 6		

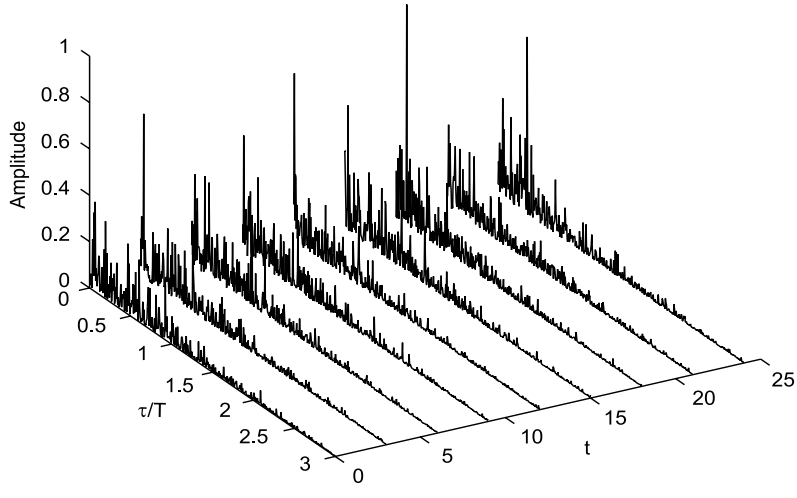


Figure 4.37. Nine realizations of $|g_{Ch}^{(bb)}(t,\tau)|$ for a Rayleigh channel with an exponential power delay profile having $\overline{\tau}_{rms} = 0.5\,T$.

Suppose $\overline{f}_0 = 0$ and $\overline{f}_m = \overline{f}_{m-1} + \Delta f_m$, $m = 1, \ldots, N_f$, the choice (4.254) corresponds to minimizing the error

$$\sum_{m=1}^{N_f} \int_{\overline{f}_{m-1}}^{\overline{f}_m} (f_m - f)^2 \mathcal{D}(f)\, df \tag{4.255}$$

The phases $\varphi_{i,m}$, $\Phi_{i,I}$ and $\Phi_{i,Q}$ are uniformly distributed in $[0, 2\pi)$ and statistically independent. This choice for $\Phi_{i,I}$ and $\Phi_{i,Q}$ ensures that the real and imaginary parts of g_i' are *statistically independent*.

The amplitude is given by $A_{i,m} = \sqrt{\mathcal{D}(f_m)\Delta f_m}$. If $\mathcal{D}(f)$ is flat, by the central limit theorem we can claim that g_i' is a Gaussian process; if instead $\mathcal{D}(f)$ presents some frequencies with large amplitude, $A_{i,m}$ must be generated as a Gaussian random variable with zero mean and variance $\mathcal{D}(f_m)\Delta f_m$.

In Figure 4.37 are represented nine realizations of the amplitude of the impulse response of a Rayleigh channel obtained by the simulation model of Figure 4.35, for an exponential power delay profile with $\overline{\tau}_{rms} = 0.5\,T$. The Doppler frequency f_d was assumed to be zero. We point out that the parameter $\overline{\tau}_{rms}$ provides scarce information on the actual behavior of $g_{Ch}^{(bb)}$, which can scatter for a duration equal to 4-5 times $\overline{\tau}_{rms}$.

4.7 Telephone channel

4.7.1 Characteristics

Telephone channels, originally conceived for the transmission of voice, today are extensively used also for the transmission of data. Transmission of a signal over a telephone

channel is achieved by utilizing several transmission media, such as symmetrical transmission lines, coaxial cables, optical fibers, radio, and satellite links. Therefore channel characteristics depend on the particular connection established. As a statistical analysis made in 1983 indicated [2], a telephone channel is characterized by the following disturbances and distortions.

Linear distortion

The frequency response $\mathcal{G}_{Ch}(f)$ of a telephone channel can be approximated by a passband filter with band in the range of frequencies from 300 to 3400 Hz.

The plots of the attenuation

$$\mathtt{a}(f) = -20 \log_{10} |\mathcal{G}_{Ch}(f)| \tag{4.256}$$

and of the group delay or envelope delay (see (1.149))

$$\tau(f) = -\frac{1}{2\pi} \frac{d}{df} \arg \mathcal{G}_{Ch}(f) \tag{4.257}$$

are illustrated in Figure 4.38 for two typical channels. The attenuation and envelope delay distortion are normalized by the values obtained for $f = 1004$ Hz and $f = 1704$ Hz, respectively.

Noise sources

Impulse noise. It is caused by electromechanical switching devices and is measured by the number of times the noise level exceeds a certain threshold per unit of time.

Quantization noise. It is introduced by the digital representation of voice signals and is the dominant noise in telephone channels (see Chapter 5). For a single quantizer, the signal-to-quantization noise ratio Λ_q has the behavior illustrated in Figure 4.39.

Thermal noise. It is described in Section 4.2 and is present at a level of $20 \div 30$ dB below the desired signal.

Non-linear distortion

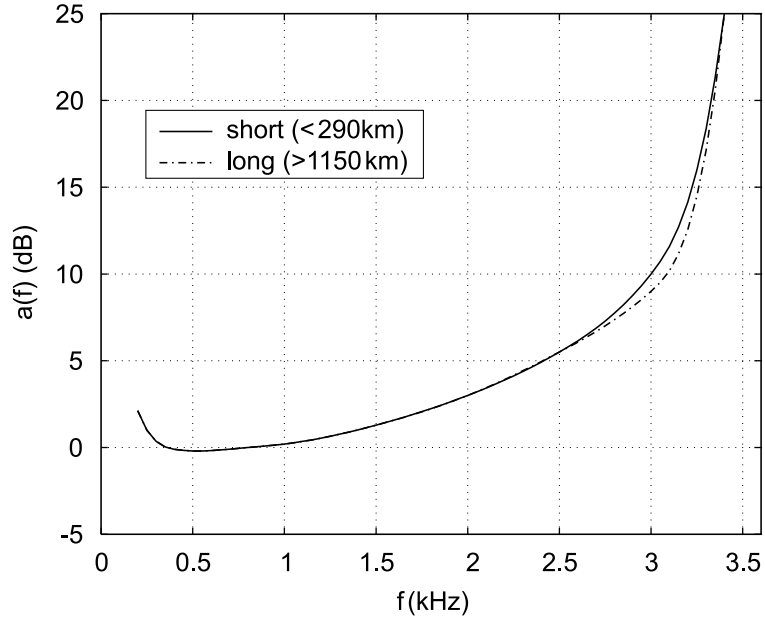
It is caused by amplifiers and by non-linear A-law and μ-law converters (see Chapter 5).

Frequency offset

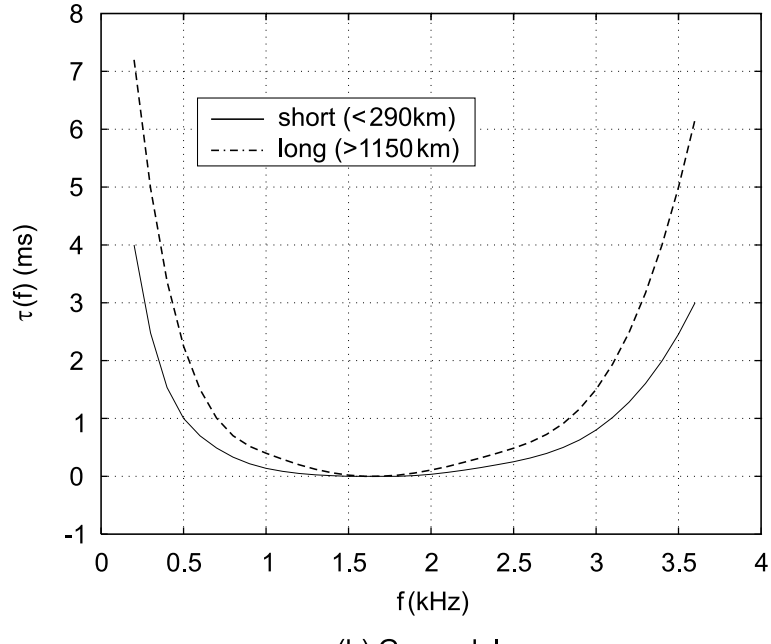
It is caused by the use of carriers for frequency up and downconversion. The relation between the channel input $x(t)$ and output $y(t)$ is given by

$$Y(f) = \begin{cases} X(f - f_{off}) & f > 0 \\ X(f + f_{off}) & f < 0 \end{cases} \tag{4.258}$$

Usually $f_{off} \leq 5$ Hz.



(a) Attenuation



(b) Group delay

Figure 4.38. Attenuation and envelope delay distortion for two typical telephone channels.

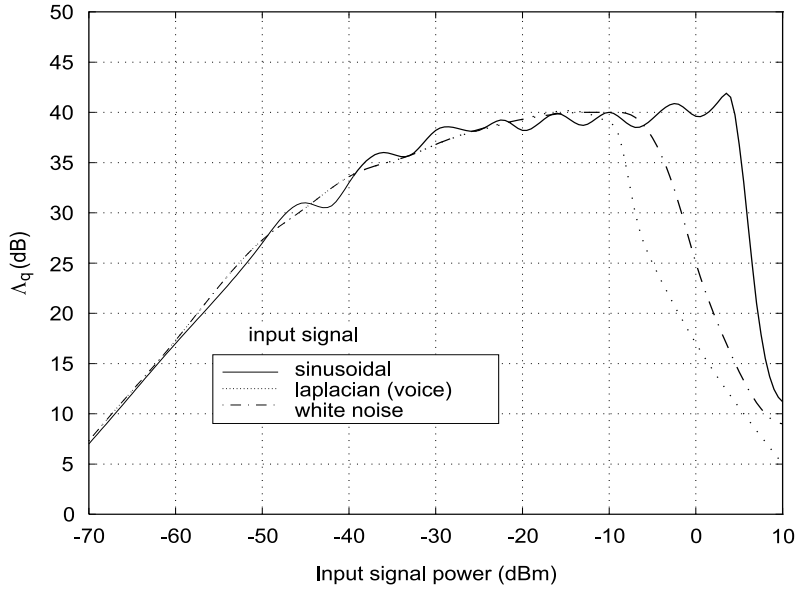


Figure 4.39. Signal to quantization noise ratio as a function of the input signal power for three different inputs.

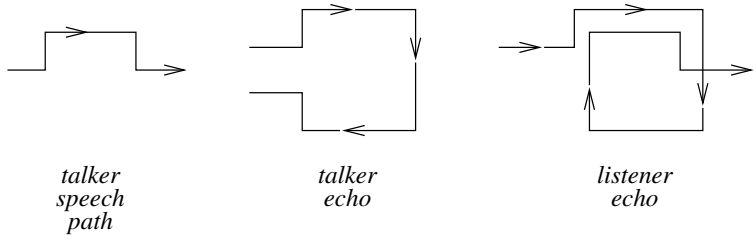


Figure 4.40. Three of the many signal paths in a simplified telephone channel with a single two-to-four wire conversion at each end.

Phase jitter

It is a generalization of the frequency *offset* (see (4.270)).

Echo

As discussed in Section 3.6.5, it is caused by the mismatched impedances of the *hybrid*. As illustrated in Figure 4.40, there are two types of echoes:

1. *Talker echo*: part of the signal is reflected and input to the receiver at the transmit side. If the echo is not very delayed, then it is practically indistinguishable from the original voice signal;

2. *Listener echo*: if the echo is reflected a second time, it returns to the listener and disturbs the original signal.

On terrestrial channels the round-trip delay of echoes is of the order of $10 \div 60$ ms, whereas on satellite links it may be as large as 600 ms. We note that the effect of echo is similar to multipath fading in radio systems. To mitigate the effect of echo there are two strategies:

- use echo suppressors that attenuate the unused connection of a four-wire transmission line;
- use echo cancellers that cancel the echo at the source, as illustrated in the scheme of Figure 3.36.

4.8 Transmission channel: general model

In this section we will describe a transmission channel model that takes into account the non-linear effects due to the transmitter and the disturbance introduced by the receiver and by the channel. We will now analyze the various blocks of the baseband equivalent model illustrated in Figure 4.41.

Power amplifier (HPA)

The final transmitter stage in a communication system usually consists of a *high power amplifier* (HPA). The HPA is a non-linear device with saturation in the sense that, in addition to not amplifying the input signal above a certain value, it introduces non-linear distortion of the signal itself. The non-linearity of a HPA can be described by a *memoryless envelope model*. Let $s(t)$ be the input signal of the HPA, expressed as

$$s(t) = A(t)\cos[2\pi f_0 t + \varphi(t)] \tag{4.259}$$

where $A(t) \geq 0$ is the signal envelope, and $\varphi(t)$ is the instantaneous phase deviation.

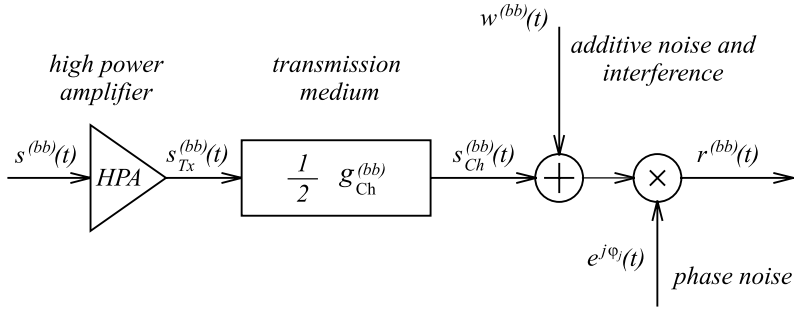


Figure 4.41. Baseband equivalent model of a transmission channel including a non-linear device.

The envelope and the phase of the output signal, $s_{Tx}(t)$, depend on the instantaneous, i.e. without memory, transformations of the input:

$$s_{Tx}(t) = G[A(t)]\cos(2\pi f_0 t + \varphi(t) + \Phi[A(t)]) \tag{4.260}$$

It is usually more convenient to refer to baseband equivalent signals:

$$s^{(bb)}(t) = A(t)e^{j\varphi(t)} \tag{4.261}$$

and

$$s_{Tx}^{(bb)}(t) = G[A(t)]e^{j(\varphi(t)+\Phi[A(t)])} \tag{4.262}$$

The functions $G[A]$ and $\Phi[A]$, called *envelope transfer functions*, represent respectively the amplitude/amplitude (AM/AM) conversion and the amplitude/phase (AM/PM) conversion of the amplifier.

In practice, the HPA are of two types. For each type we give the AM/AM and AM/PM functions commonly adopted for the analysis. First, however, we need to introduce some normalizations. As a rule, the point at which the amplifier operates is identified by the *back-off*. We adopt here the following definitions for the *input back-off* (IBO) and the *output back-off* (OBO):

$$\text{IBO} = 20\log_{10}\left(\frac{S}{\sqrt{\mathtt{M}_s}}\right) \quad \text{(dB)} \tag{4.263}$$

$$\text{OBO} = 20\log_{10}\left(\frac{S_{Tx}}{\sqrt{\mathtt{M}_{s_{Tx}}}}\right) \quad \text{(dB)} \tag{4.264}$$

where $\mathtt{M}_s$ is the statistical power of the input signal s, $\mathtt{M}_{s_{Tx}}$ is the statistical power of the output signal s_{Tx}, and S and S_{Tx} are the amplitudes of the input and output signals, respectively, that lead to saturation of the amplifier. Here we assume $S = 1$ and $S_{Tx} = G[1]$ for all the amplifiers considered.

TWT. The *travelling wave tube* (TWT) is a device characterized by a strong AM/PM conversion. The conversion functions are

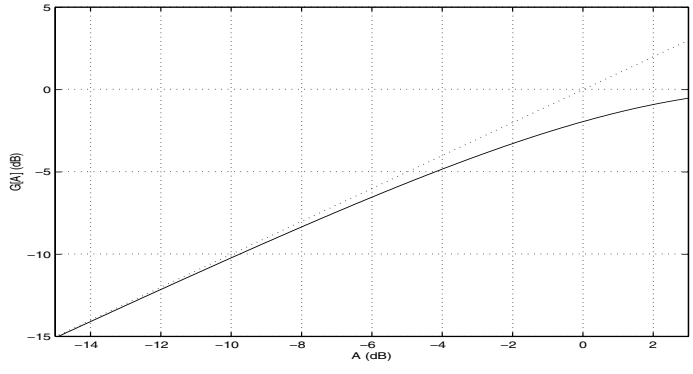
$$G[A] = \frac{\alpha_A A}{1 + \beta_A A^2} \tag{4.265}$$

$$\Phi[A] = \frac{\alpha_\Phi A^2}{1 + \beta_\Phi A^2} \tag{4.266}$$

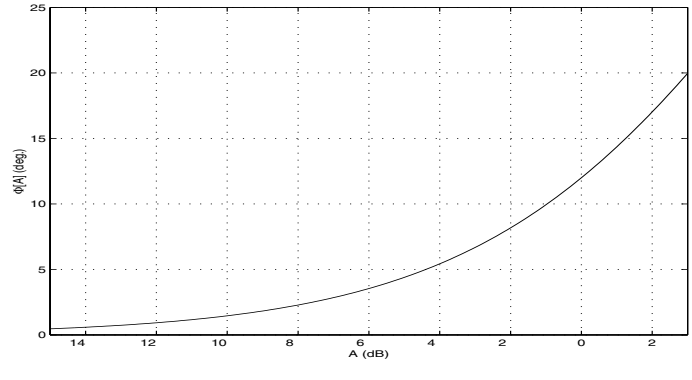
where α_A, β_A, α_Φ and β_Φ suitable parameters.

The (4.265) and (4.266) are illustrated in Figure 4.42 for $\alpha_A = 1$, $\beta_A = 0.25$, $\alpha_\Phi = 0.26$ and $\beta_\Phi = 0.25$.

SSPA. The *solid state power amplifier* (SSPA) has a more linear behavior in the region of small signals as compared to the TWT. The AM/PM conversion is usually negligible.



(a)



(b)

Figure 4.42. AM/AM and AM/PM characteristics of a TWT for $\alpha_A = 1$, $\beta_A = 0.25$, $\alpha_\Phi = 0.26$ and $\beta_\Phi = 0.25$.

Therefore the conversion functions are

$$G[A] = \frac{A}{(1 + A^{2p})^{1/2p}} \tag{4.267}$$

$$\Phi[A] = 0 \tag{4.268}$$

where p is a suitable parameter.

In Figure 4.43 the function $G[A]$ is plotted for three values of p; the superimposed dashed line is an ideal curve given by

$$G[A] = \begin{cases} A & 0 < A < 1 \\ 1 & A \geq 1 \end{cases} \tag{4.269}$$

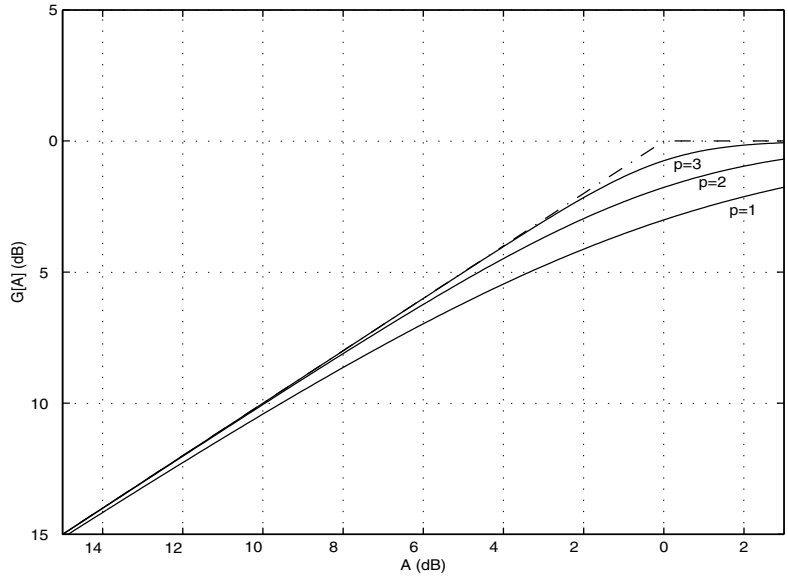


Figure 4.43. AM/AM characteristic of a SSPA.

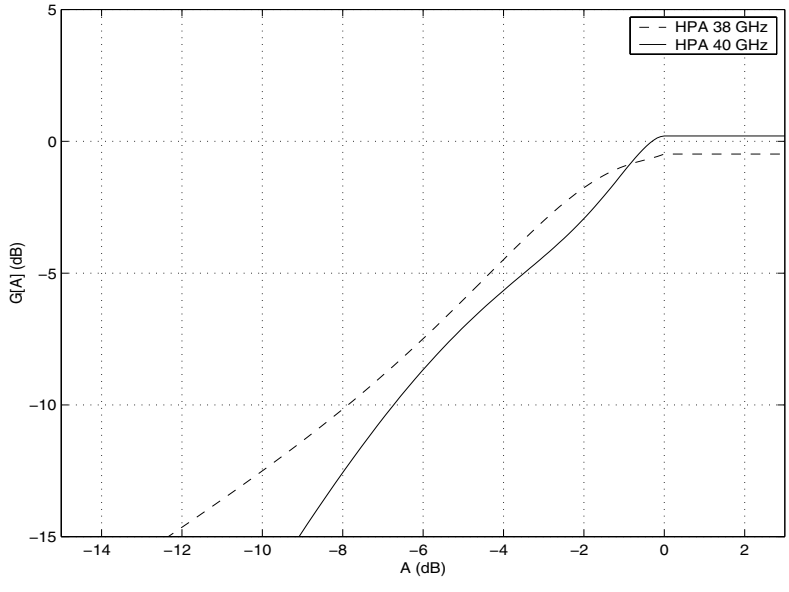


Figure 4.44. AM/AM experimental characteristic of two amplifiers operating at 38 GHz and 40 GHz.

It is interesting to compare the above analytical models with the behavior of a practical HPA. Figure 4.44 illustrates the AM/AM characteristics of two waveguide HPA operating at frequencies of 38 GHz and 40 GHz.

Transmission medium

The transmission medium is typically modelled as a filter. For transmission lines and radio links the models are given respectively in Sections 4.4 and 4.6.

Additive noise

Several noise sources that cause a degradation of the received signal may be present in a transmission system. Consider, for example, the noise introduced by a receive antenna or the thermal noise and shot noise generated by the pre-amplifier stage of a receiver. At the receiver input, all these noise signals are modelled as an effective additive white Gaussian noise (AWGN) signal, statistically independent of the desired signal. The power spectral density of the AWGN noise can be obtained by the analysis of the system devices, or by experimental measurements.

Phase noise

The demodulators used at the receivers are classified as "coherent" or "non-coherent", depending on whether they use or not a carrier signal, which ideally should have the same phase and frequency as the carrier at the transmitter, to demodulate the received signal. Typically both phase and frequency are recovered from the received signal by a *phase locked loop* (PLL) system, which employs a local oscillator. The recovered carrier may differ from the transmitted carrier because of the *phase noise*,[12] due to short-term stability, i.e. *frequency drift*, of the oscillator, and because of the dynamics and transient behavior of the PLL.

The recovered carrier is expressed as

$$v(t) = V_o[1 + a(t)]\cos\left(\omega_0 t + \varphi_j(t) + \frac{dt^2}{2}\right) \tag{4.270}$$

where d (long-term drift) represents the effect due to ageing of the oscillator, $a(t)$ is the amplitude noise, and $\varphi_j(t)$ denotes the phase noise. Often the amplitude noise $a(t)$, as well as the effect of ageing, can be neglected. The phase noise is usually represented in a transmission system model as in Figure 4.41.

The *phase noise* $\varphi_j(t)$ consist of deterministic components and random noise. For example, temperature change, supply voltage, and the output impedance of the oscillator are included among deterministic components.

[12] Sometimes also called *phase jitter*.

Ignoring the deterministic effects, with the exception of the *frequency drift*, a PSD model of the of $\varphi_j(t)$ comprises five terms:

$$\mathcal{P}_{\varphi_j}(f) = \underbrace{k_{-4}\frac{f_0^2}{f^4}}_{\substack{\text{random}\\ \text{frequency}\\ \text{walk}}} + \underbrace{k_{-3}\frac{f_0^2}{f^3}}_{\substack{\text{flicker}\\ \text{frequency}\\ \text{noise}}} + \underbrace{k_{-2}\frac{f_0^2}{f^2}}_{\substack{\text{random phase}\\ \text{walk or white}\\ \text{frequency noise}}} + \underbrace{k_{-1}\frac{f_0^2}{f}}_{\substack{\text{flicker}\\ \text{phase noise}}} + \underbrace{k_0 f_0^2}_{\substack{\text{white}\\ \text{phase noise}}} \tag{4.271}$$

for $f_\ell \le f \le f_h$.

A simplified model, often used, is given by

$$\mathcal{P}_{\varphi_j}(f) = c + \begin{cases} a & |f| \le f_1 \\ b\dfrac{1}{f^2} & f_1 \le |f| < f_2 \end{cases} \tag{4.272}$$

where the parameters a and c are typically of the order of -65 dBc/Hz and -125 dBc/Hz, respectively, and b is a scaling factor that depends on f_1 and f_2 and assures continuity of the PSD. dBc means dB *carrier*, that is it represents the statistical power of the *phase noise*, expressed in dB, with respect to the statistical power of the desired signal received in the passband.

Depending on the values of a, b, c, f_1 and f_2, typical values of the statistical power of $\varphi_j(t)$ are in the range from 10^{-2} to 10^{-4}. The plot of (4.272) is shown in Figure 4.45 for $f_1 = 0.1$ MHz, $f_2 = 2$ MHz, $a = -65$ dBc/Hz, and $c = -125$ dBc/Hz.

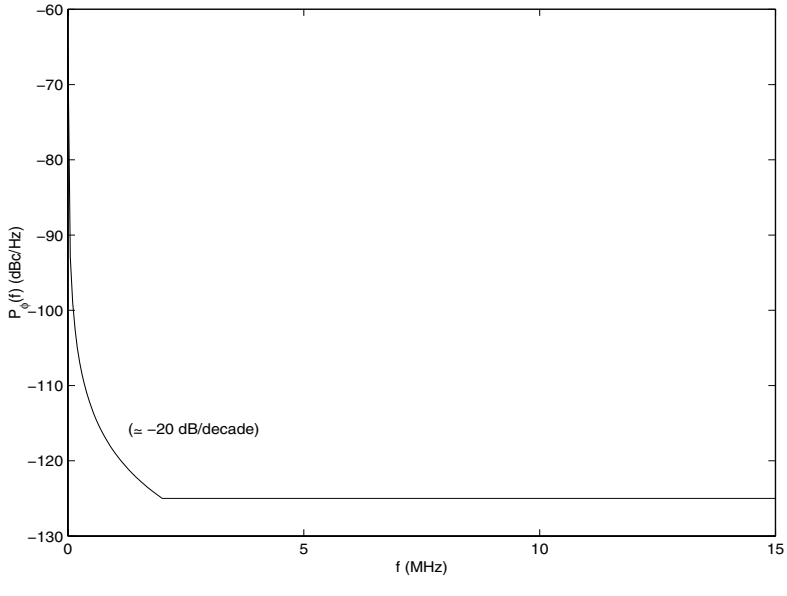


Figure 4.45. Simplified model of the phase-noise power spectral density.

Bibliography

[1] C. G. Someda, *Electromagnetic waves*. London: Chapman & Hall, 1998.

[2] M. of the Technical Staff, *Transmission systems for communications*. Winston, NC: Bell Telephone Laboratories, 5th ed., 1982.

[3] S. Ramo, J. R. Whinnery, and T. Van Duzer, *Fields and Waves in Communication Electronics*. New York: John Wiley & Sons, 1965.

[4] G. Cherubini, S. Ölçer, G. Ungerboeck, J. Creigh, and S. K. Rao, "100BASE-T2: a new standard for 100 Mb/s ethernet transmission over voice-grade cables", *IEEE Communications Magazine*, vol. 35, pp. 115–122, Nov. 1997.

[5] R. J. Hoss, *Fiber optic communications*. Englewood Cliffs, NJ: Prentice-Hall, 1990.

[6] L. B. Jeunhomme, *Single-mode fiber optics*. New York: Marcel Dekker, 2nd ed., 1990.

[7] J. C. Palais, *Fiber optic communications*. Englewood Cliffs, NJ: Prentice-Hall, 3rd ed., 1992.

[8] T. Li, "Structures, parameters, and transmission properties of optical fibers", *Proc. IEEE*, p. 1175, Oct. 1980.

[9] T. Miya, Y. Terunuma, T. Hosaka, and T. Miyashita, "Ultimate low-loss single-mode fibre at 1.55 µm", *IEE Electronics Letters*, vol. 15, pp. 106–108, Feb. 1979.

[10] J. C. Palais, "Fiber optic communications systems", in *The Communications Handbook* (J. D. Gibson, ed.), ch. 54, pp. 731–739, Boca Raton: CRC Press, 1997.

[11] D. G. Messerschmitt and E. A. Lee, *Digital communication*. Boston, MA: Kluwer Academic Publishers, 2nd ed., 1994.

[12] K. Feher, *Wireless digital communications*. Upper Saddle River, NJ: Prentice-Hall, 1995.

[13] T. S. Rappaport, *Wireless communications: principles and practice*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[14] K. Pahlavan and A. H. Levesque, *Wireless information networks*. New York: John Wiley & Sons, 1995.

[15] W. C. Jakes, *Microwave mobile communications*. New York: IEEE Press, 1993.

[16] G. L. Stuber, *Principles of mobile communication*. Norwell, MA: Kluwer Academic Publishers, 1996.

[17] J. J. Spilker, *Digital communications by satellite*. Englewood Cliffs, NJ: Prentice-Hall, 1977.

[18] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[19] A. Anastasopoulos and K. Chugg, "An efficient method for simulation of frequency selective isotropic Rayleigh fading", in *Proc. 1997 IEEE Vehicular Technology Conference*, pp. 2084–2088, May 1997.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 5

Digital representation of waveforms

Figure 5.1a illustrates the conventional transmission of an analog signal, for example, speech or video, over an analog channel; in this scheme the transmitter usually consists of an amplifier and possibly a modulator, the analog transmission channel is of the type discussed in Chapter 4, and the receiver consists of an amplifier and possibly a demodulator. Alternatively, the transmission may take place by first encoding[1] the information contained in the analog signal into a sequence of bits using for example an *analog-to-digital converter* (ADC), as illustrated in Figure 5.1b.

If T_b is the time interval between two consecutive bits of the sequence, the *bit rate* of the ADC is $R_b = 1/T_b$ (bit/s). The binary message is converted by a digital modulator into a waveform that is suitable for transmission over an analog channel. At the receiver, the reverse process occurs: in this case a digital demodulator restores the message, whereas the conversion of the sequence of bits to an analog signal is performed by a *digital-to-analog converter* (DAC). The system that has as an input the sequence of bits produced by the ADC, and as an output the sequence of bits produced by the digital demodulator is called a *binary channel* (see Chapter 7).

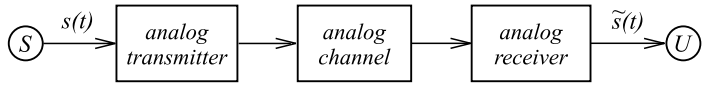
In this chapter the principles and methods for the conversion of analog signals into binary messages and viceversa will be discussed; as a practical example we will use speech, but the principles may be extended to any analog signal. To compute system performance, a fundamental parameter is the signal-to-noise ratio. Let $s(t)$ be the original signal, $\tilde{s}(t)$ the reconstructed signal, and $e_q(t) = \tilde{s}(t) - s(t)$; then the signal-to-noise ratio is defined as

$$\Lambda_q = \frac{E[s^2(t)]}{E[e_q^2(t)]} \tag{5.1}$$

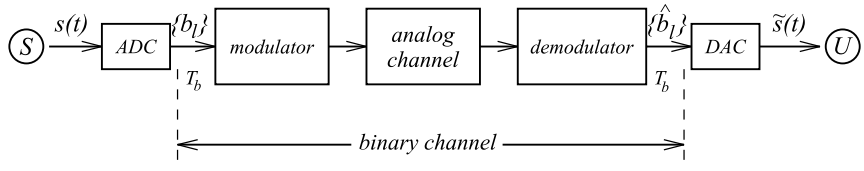
5.1 Analog and digital access

Analog access over a telephone channel in the *public switched telephone network* (PSTN) is illustrated in Figure 5.2. With reference to the figure, the word "modem" is the contraction of *mod*ulator-*dem*odulator. Its function is to convert a binary message, or data signal, into

[1] We bring to the attention of the reader that the terms "encoder" and "decoder" are commonly used to indicate various devices in a communication system. In this chapter we will deal with encoders and decoders for the digital representation of analog waveforms.



(a) Conventional model



(b) Digital channel model

Figure 5.1. Analog vs. digital transmission.

an analog passband signal that can be transmitted over the telephone channel. In Figure 5.2, the source generates a speech signal or a data file; in the latter case, a modem is required to transmit the signal. The analog signal $s(t)$ that has a band of approximately 300–3400 Hz is sent over a local loop to the central office (see Chapter 4): here it is usually converted into a binary digital message via PCM at 64 kbit/s; in turn this message is modulated before being transmitted over an analog channel. After having crossed several central offices where switching (routing) of the signal takes place, the PCM encoded message arrives at the destination central office: here it is converted into an analog signal and sent over a local loop to the end user. It is here that the signal must be identified as a speech signal or a digitally modulated signal; in the latter case a modem will demodulate it to reproduce the data message.

Figure 5.3 illustrates the concept of direct digital access at the user's premises. An analog signal is converted into a digital message via an ADC. The user digital message is then sent over the analog channel by a modulator. At the receiver the inverse process is established, where the digital message obtained at the output of the demodulator may be used to restore an analog signal via a DAC.

In comparing the two systems, we note the waste of capacity of the system in Figure 5.2. For example, for a 9600 bit/s modem, the modulated PCM encoded signal requires a standard capacity of $R_b = 64$ kbit/s. By directly accessing the PCM link at the user's home, we could transmit $64000/9600 \simeq 6$ data signals at 9600 bit/s.

5.1.1 Digital representation of speech

Some waveforms

Some examples of speech waveforms for an interval of 0.25 s are given in Figure 5.4. From these plots, we can obtain a speech model as a succession of *voiced speech* spurts (see Figure 5.5a), or *unvoiced speech* spurts (see Figure 5.5b). In the first case, the signal

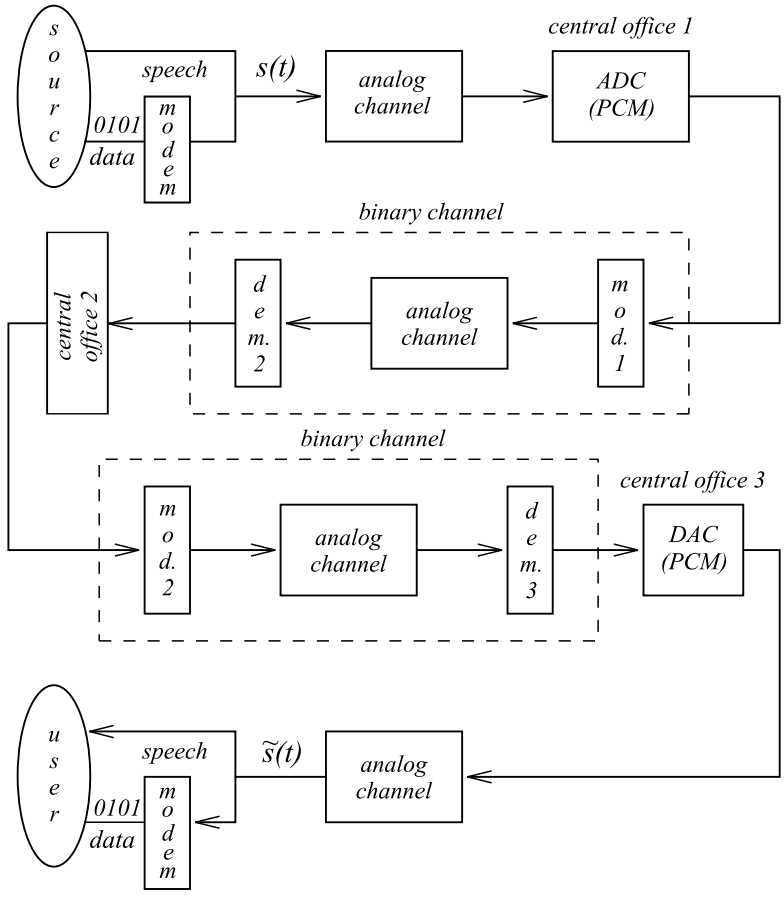


Figure 5.2. User line with analog access.

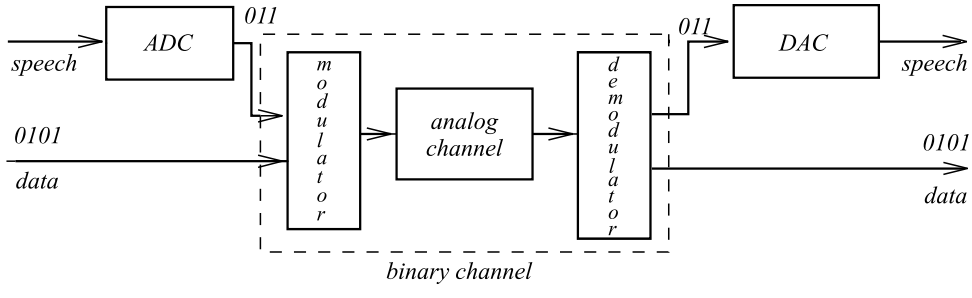


Figure 5.3. User line with digital access.

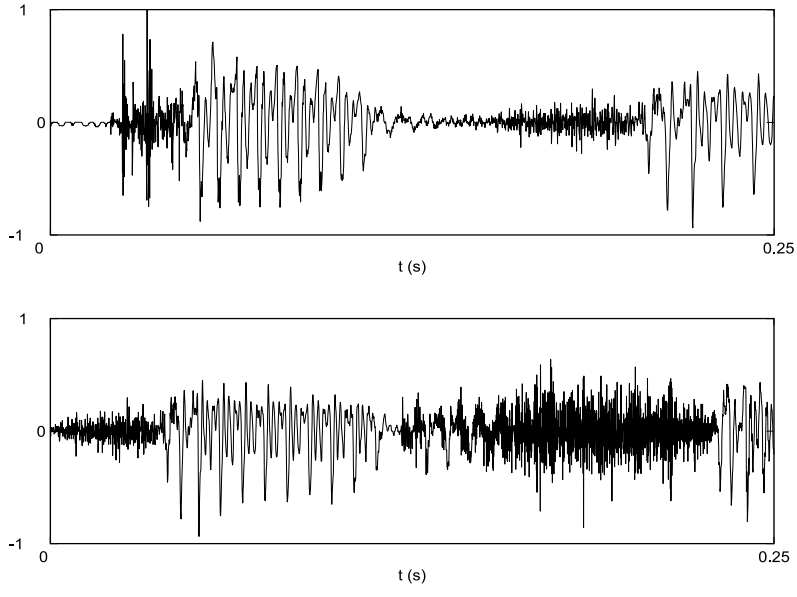


Figure 5.4. Speech waveforms.

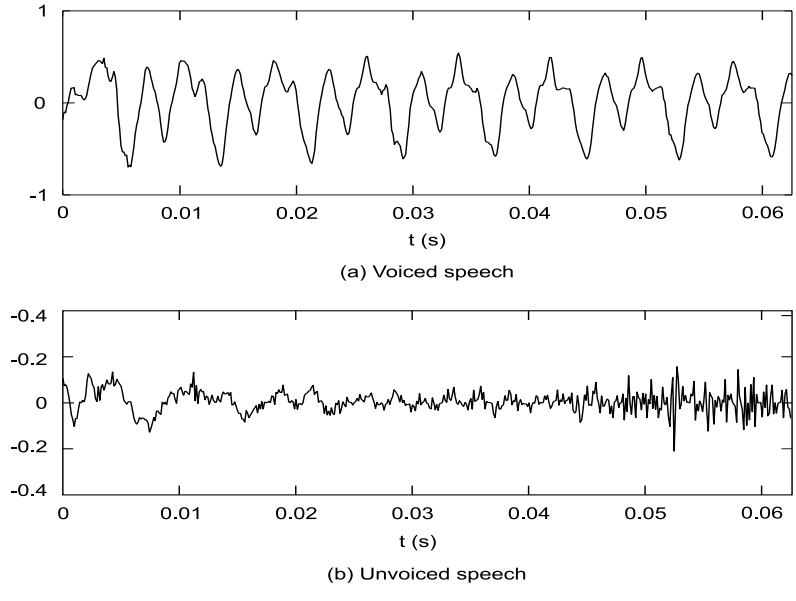


Figure 5.5. Voiced and unvoiced speech spurts.

is strongly correlated and almost periodic, with a period that is called *pitch*, and exhibits large amplitudes; conversely in an *unvoiced speech* spurt the signal is weakly correlated and has small amplitudes. We note moreover that the average level of speech changes in time: indeed speech is a non-stationary signal. In Figure 5.6 it is interesting to observe the instantaneous spectrum of some voiced and unvoiced sounds; we also note that the latter may have a bandwidth larger than 10 kHz.

Concerning the *amplitude distribution* of speech signals, we observe that over short time intervals, of the order of a few tenths of milliseconds (or of a few hundreds of samples at a sampling frequency of 8 kHz), the amplitude statistic is Gaussian with good approximation; over long time intervals, because of the numerous pauses in speech, it tends to exhibit a gamma or Laplacian distribution. We give here the probability density functions of the amplitude that are usually adopted. Let σ_s be the standard deviation of the signal $s(t)$; then we have

$$\begin{aligned} &\text{gamma:} && p_s(a) = \left(\frac{\sqrt{3}}{8\pi\sigma_s|a|}\right)^{\frac{1}{2}} e^{-\frac{\sqrt{3}|a|}{2\sigma_s}} \\ &\text{Laplacian:} && p_s(a) = \frac{1}{\sqrt{2}\sigma_s} e^{-\frac{\sqrt{2}|a|}{\sigma_s}} \\ &\text{Gaussian:} && p_s(a) = \frac{1}{\sqrt{2\pi}\sigma_s} e^{-\frac{1}{2}\left(\frac{a}{\sigma_s}\right)^2} \end{aligned} \tag{5.2}$$

As mentioned above, analog modulated signals generated by modems, often called *voice-band data* signals, are also transmitted over telephone channels. Figure 5.7 illustrates a

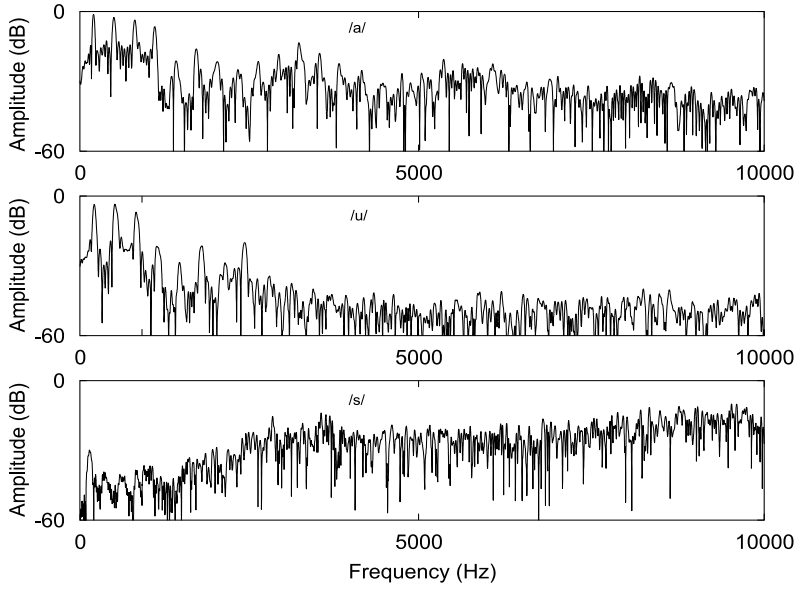


Figure 5.6. Spectrum of voiced and unvoiced sounds for a sampling frequency of 20 kHz.

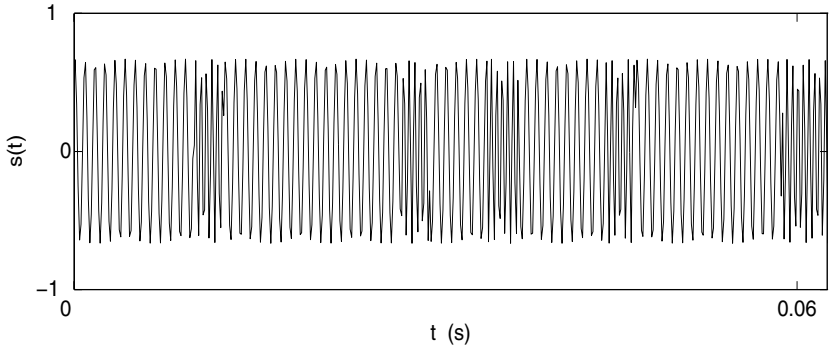


Figure 5.7. Signal generated by a modem employing FSK modulation for the transmission of 1200 bit/s.

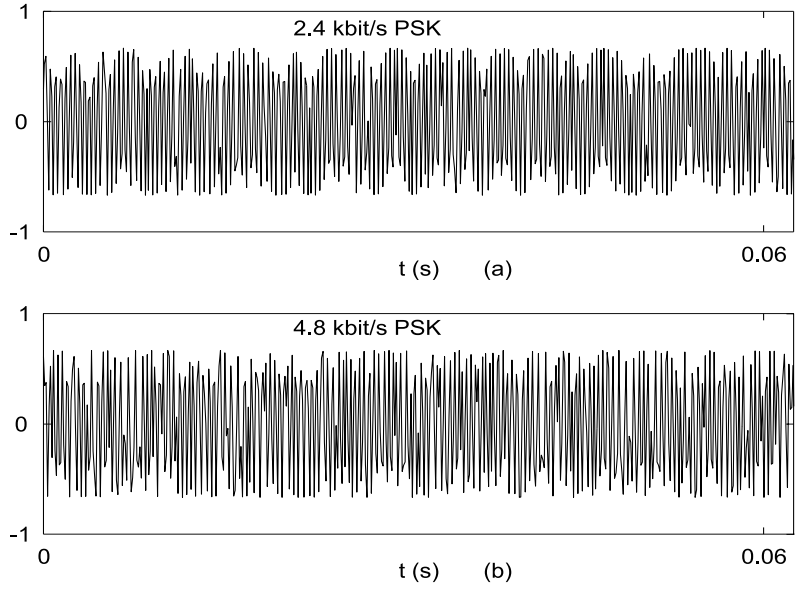


Figure 5.8. Signal generated by modems employing PSK modulation for the transmission of: (a) 2400 bit/s; (b) 4800 bit/s.

signal produced by the 202S modem, which employs FSK modulation for the transmission of 1200 bit/s, whereas Figure 5.8a and Figure 5.8b illustrate signals generated by the 201C and 208B modems, which employ PSK modulation for the transmission of 2400 and 4800 bit/s, respectively. For the definition of FSK and PSK modulation we refer the reader to Chapter 6.

In general, we note that the average level of signals generated by modems is stationary; moreover, if the bit rate is low, signals are strongly correlated.

Speech coding

Speech coding addresses person-to-person communications and is strictly related to the transmission, for example, over the public network, and storage of speech signals. The aim is to represent, using an encoder, speech as a digital signal that requires the lowest possible bit rate to recreate, by an appropriate decoder, the speech signal at the receiver [1].

Depicted in Figure 5.9 is a basic scheme, denoted as ADC, that provides the analog-to-digital conversion (encoding) of the signal, consisting of:

1. an anti-aliasing filter followed by a sampler at sampling frequency $1/T_c$;
2. a quantizer;
3. an *inverse bit mapper* (IBMAP) followed by a parallel-to-serial (P/S) converter.

As indicated by the sampling theorem, the choice of the sampling frequency $F_c = 1/T_c$ is related to the bandwidth of the signal $s(t)$ (see (1.142)). In practice, there is a trade-off between the complexity of the anti-aliasing filter and the choice of the sampling frequency, which must be greater than twice the signal bandwidth. For audio signals, F_c depends on the signal quality that we wish to maintain and therefore it depends on the application, see Table 5.1 [2].

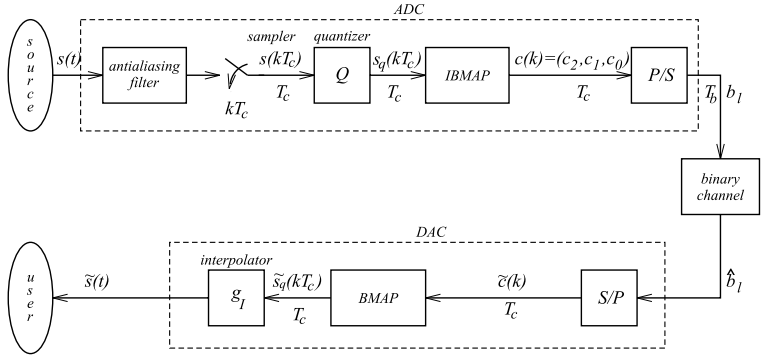


Figure 5.9. Basic scheme for the digital transmission of an analog signal.

Table 5.1 Sampling frequency of the audio signal in three applications.

Application	Passband (Hz)	F_c (Hz)
telephone	$300 - 3400$ (narrow band speech)	8000
broadcasting	$50 - 7000$ (wide band speech)	16000
audio, compact disc	$10 \div 20000$	44100
digital audio tape	$10 \div 20000$	48000

The choice of the quantizer parameters is somehow more complicated and will be dealt with in detail in the following sections. We will consider now the quantizer as an instantaneous non-linear transformation that maps the real values of s in a finite number of values of s_q. To illustrate the principle of an ADC, let us assume that s_q assumes values that are taken from a set of 8 elements:[2]

$$\mathcal{Q}[s(kT_c)] = s_q(kT_c) \in \{\mathcal{Q}_{-4}, \mathcal{Q}_{-3}, \mathcal{Q}_{-2}, \mathcal{Q}_{-1}, \mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3, \mathcal{Q}_4\} \tag{5.3}$$

Therefore $s_q(kT_c)$ may assume only a finite number of values, which can be represented as binary values, for example, using the inverse bit mapper of Table 5.2.

It is convenient to consider the sequence of bits that gives the binary representation of $\{s_q\}$ instead of the sequence of values itself. In our example, with a representation using three bits per sample, the bit rate of the system is equal to

$$R_b = 3F_c \quad \text{(bit/s)} \tag{5.4}$$

The inverse process (decoding) takes place at the receiver: the *bit-mapper* (BMAP) restores the quantized levels, and an interpolator filter yields an estimate of the analog signal.

The interpolator filter as a holder

Always referring to the sampling theorem, an ideal interpolator filter[3] has the following frequency response

$$\mathcal{G}_I(f) = \text{rect}\left(\frac{f}{F_c}\right) \tag{5.5}$$

Table 5.2 Encoder inverse bit-mapper.

Values $s_q(kT_c)$	Integer representation	Binary representation $c(k) = (c_2, c_1, c_0)$
$\mathcal{Q}_{-4}$	0	000
$\mathcal{Q}_{-3}$	1	001
$\mathcal{Q}_{-2}$	2	010
$\mathcal{Q}_{-1}$	3	011
$\mathcal{Q}_1$	4	100
$\mathcal{Q}_2$	5	101
$\mathcal{Q}_3$	6	110
$\mathcal{Q}_4$	7	111

[2] The notation adopted in (5.3) to define the set reflects the fact that in most cases the set of values assumed by s_q is symmetrical around the origin.

[3] From the Observation 1.7 on page 71, if $s_q(kT_c)$ is WSS, then the interpolated random process $s_q(t)$ is WSS and $E[s_q^2(t)] = E[s_q^2(kT_c)]$, whenever the gain of the interpolator filter is equal to one. As a result the signal-to-noise ratio in (5.1) becomes independent of t and can be computed using the samples of the processes, $\Lambda_q = E[s^2(kT_c)]/E[e_q^2(kT_c)]$.

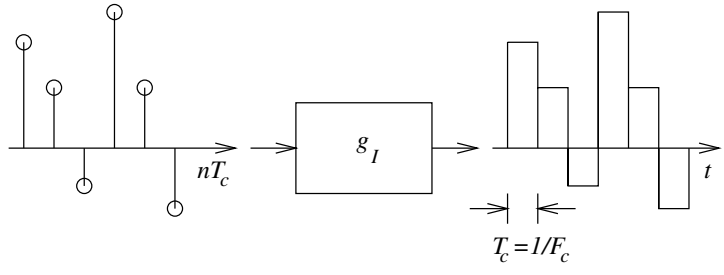


Figure 5.10. DAC interpolator as a holder.

Typically, however, the DAC employs a simple *holder* that holds the input values as illustrated in Figure 5.10. In this case

$$g_I(t) = \text{rect}\left(\frac{t - T_c/2}{T_c}\right) = \text{w}_{T_c}(t) \tag{5.6}$$

and

$$\mathcal{G}_I(f) = T_c \,\text{sinc}\left(\frac{f}{F_c}\right) e^{-2\pi f T_c/2} \tag{5.7}$$

Unless the sampling frequency has been chosen sufficiently higher than twice the bandwidth of $s(t)$, we see that the filter (5.7), besides not attenuating enough the images of $\tilde{s}_q(kT_c)$, introduces distortion in the passband of the desired signal.[4] A solution to this problem consists in introducing, before interpolation, a digital equalizer filter with a frequency response equal to $1/\,\text{sinc}(fT_c)$ in the passband of $s(t)$. Figure 5.11 illustrates the solution.

A simple digital equalizer filter is given by

$$G_{comp}(z) = -\frac{1}{16} + \frac{9}{8}z^{-1} - \frac{1}{16}z^{-2} \tag{5.8}$$

whose frequency response is given in Figure 5.12.

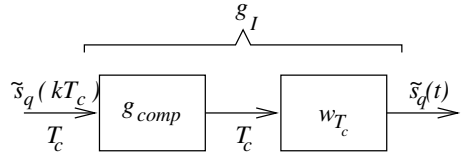


Figure 5.11. Holder filter preceded by a digital equalizer.

[4] In many applications, to simplify the analog interpolator filter, the signal before interpolation is oversampled: for example, by digital interpolation of the signal $\tilde{s}_q(kT_c)$ by at least a factor of 4.

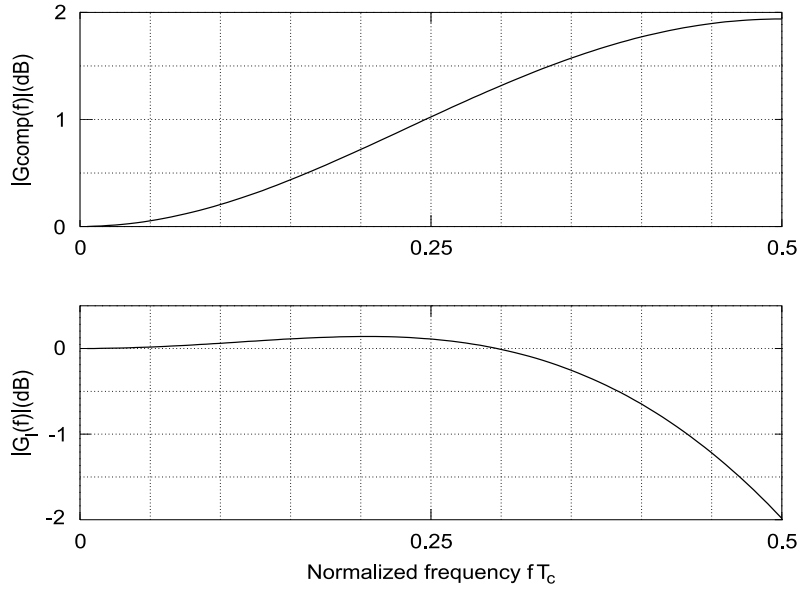


Figure 5.12. Frequency responses of a three-coefficient equalizer filter g_{comp} and of the overall filter $g_I = g_{comp} * w_{T_c}$.

An alternative solution is represented by an IIR filter with:

$$G_{comp}(z) = \frac{9/8}{1 + 1/8z^{-1}} \tag{5.9}$$

whose frequency response is given in Figure 5.13.

In the following sections, by the term DAC we mean a digital-to-analog converter with the aforementioned variations.

Sizing of the binary channel parameters

As will be further discussed in Section 6.2, in Figure 5.9 the binary channel is characterized by the encoder bit rate R_b. If B is the bandwidth of $s(t)$, the sample frequency F_c is such that

$$F_c = \frac{1}{T_c} \geq 2B \tag{5.10}$$

If $L = 2^b$ is the number of levels of the quantizer, then the encoder bit rate is equal to $R_b = bF_c$ bit/s.

Another important parameter of the binary channel is the bit error probability

$$P_{bit} = P[\hat{b}_\ell \neq b_\ell] \tag{5.11}$$

If an error occurs, the reconstructed binary representation $\tilde{c}(k)$ is different from $c(k)$: consequently the reconstructed level is $\tilde{s}_q(kT_c) \neq s_q(kT_c)$. In the case of a speech signal,

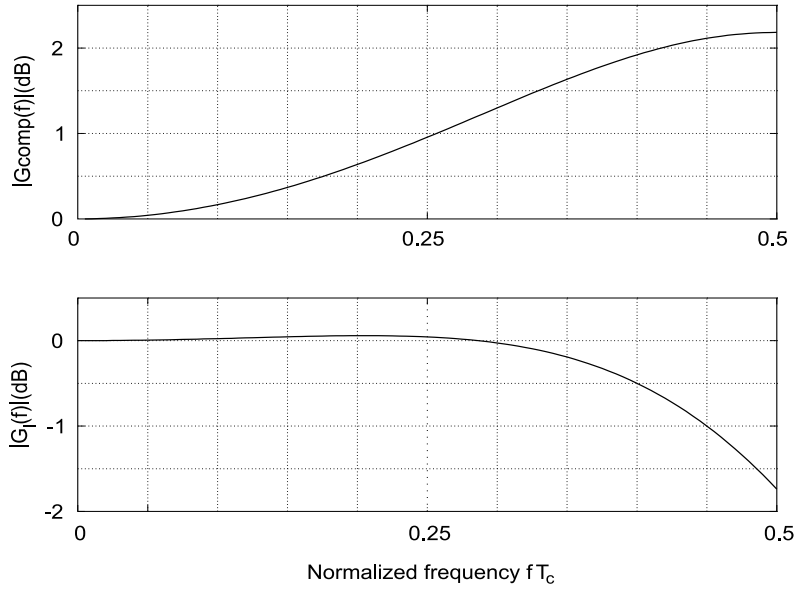


Figure 5.13. Frequency responses of an IIR equalizer filter g_{comp} and of the overall filter $g_I = g_{comp} * w_{T_c}$.

such an event is perceived by the ear as a fastidious impulse disturbance. For speech signals to have an acceptable quality at the receiver it must be $P_{bit} \leq 10^{-3}$.

5.1.2 Coding techniques and applications

At the output of an ADC, the PCM encoded samples, after suitable transformations, can be further quantized in order to reduce the bit rate. From [3], we list in Figure 5.14 various coding techniques, which are divided into three groups, that essentially exploit two elements:

- redundancy of speech,
- sensibility of the ear as a function of the frequency.

Waveform coding. *Waveform encoders* attempt to reproduce the waveform as closely as possible. This type of coding is applicable to any type of signal; two examples are the PCM and ADPCM schemes.

Coding by modeling. In this case coding is not related to signal samples, but to the parameters of the source that generates them. Assuming the *voiced/unvoiced* speech model, an example of a classical encoder (*vocoder*) is given in Figure 5.15, where a periodic excitation, or white noise segment, filtered by a suitable filter, yields a synthesized speech segment. A more sophisticated model uses a more articulated *multipulse* excitation.

Frequency-domain coding. In this case coding occurs after signal transformation to a domain different from time, usually frequency: examples are sub-band coding and transform coding.

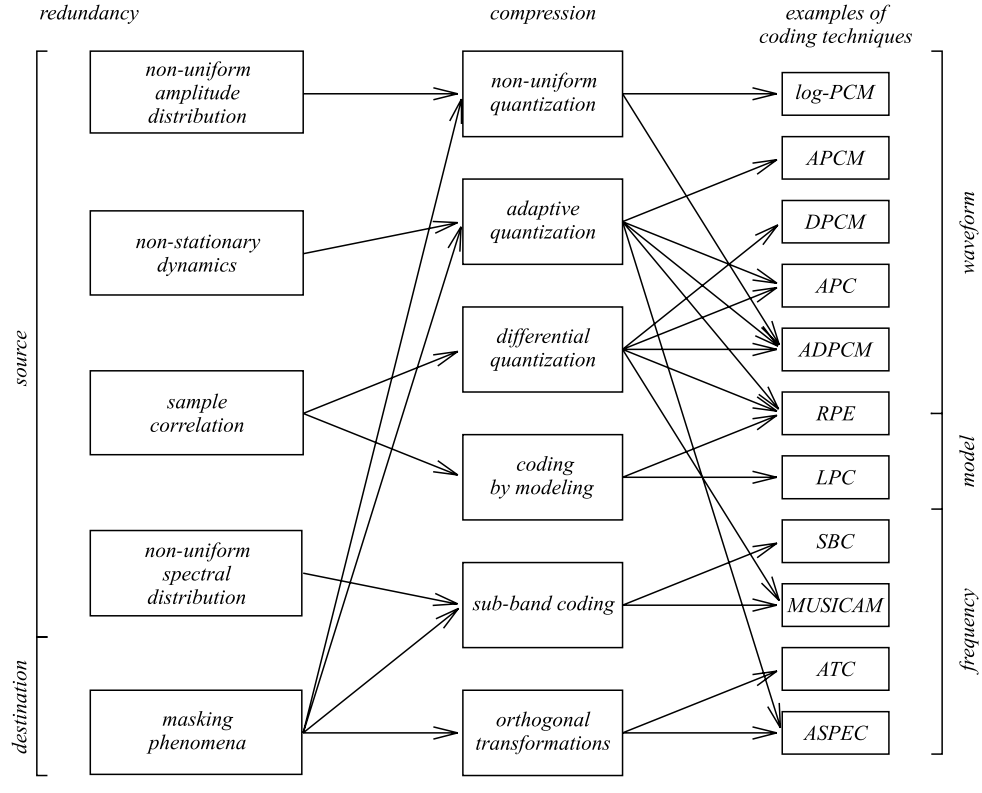


Figure 5.14. Characteristics exploited by the different coding techniques.

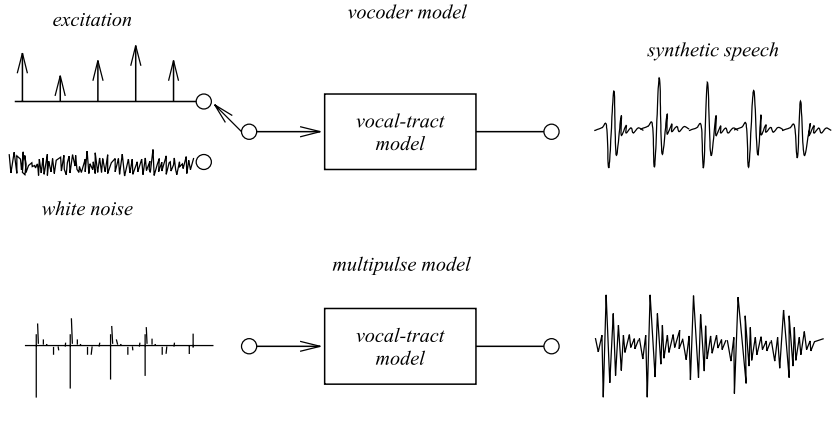


Figure 5.15. Vocoder and multipulse models for speech synthesis.

Table 5.3 Voice coding techniques.

Bit rate (kbit/s) 1.2 2.4 4.8 8.0 9.6 16 32 64	Algorithm		Year
———	Codebook excited LP	CELP	1984
———	Multipulse excited LP	MELP	1982
———	Vector quantization	VQ	1980
———	Time domain harmonic scaling	TDHS	1979
———	Adaptive transform coding	ATC	1977
———	Sub-band coding	SBC	1976
———	Residual excited LP	RELP	1975
———	Adaptive predictive coding	APC	1968
———	Formant vocoder	FOR-V	1971
———	Cepstral vocoder	CEP-V	1969
———	Channel vocoder	CHA-V	1967
———	Phase vocoder	PHA-V	1966
———	Linear prediction vocoder	LPC-V	1966
———	Adaptive differential PCM	ADPCM	
———	Differential PCM	DPCM	
———	Adaptive delta modulation	ADM	
———	Delta modulation	DM	
———	Pulse code modulation	PCM	

Various coding techniques are listed in Table 5.3. Table 5.4 illustrates the characteristics of a few systems, putting into evidence that for more sophisticated encoders the implementation complexity expressed in millions of instructions per second (MIPS), as well as the delay introduced by the encoder (*latency*), can be considerable.

The various coding techniques are different in quality and cost of implementation. With respect to the perceived quality, on a scale from poor to excellent, three categories of encoders perform as illustrated in Figure 5.16: obviously a higher implementation complexity is expected for encoders with low bit rate and good quality. We go from a bit rate in the range from 4.4 to 9.6 kbit/s for cellular radio systems, to a bit rate in the range from 16 to 64 kbit/s for transmission over the public network.

Generally a coding technique is strictly related to the application and depends on various factors:

- signal type (for example speech, music, voice-band data, signalling, etc.);
- maximum tolerable latency;
- implementation complexity.

In particular, speech encoder applications for bit rate in the range 4–16 kbit/s are:

- long distance and satellite transmission;
- digital mobile radio (cellular radio);

Table 5.4 Parameters of a few speech coders.

Coder	Bit rate (kbit/s)	Computational complexity (MIPS)	Latency (ms)
PCM	64	0.0	0
ADPCM	32	0.1	0
ASBC	16	1	25
MELP	8	10	35
CELP	4	100	35
LPC	2	1	35

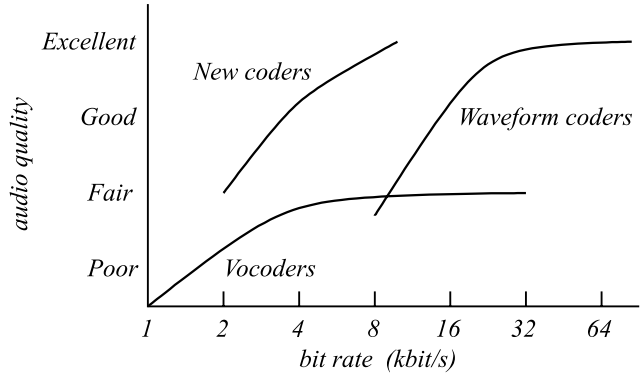


Figure 5.16. Audio quality vs. bit rate for three categories of encoders.

- modem transmission over the telephone channel (voice mail);
- speech storage for telephone services and speech encryption;
- packet networks with integrated speech and data.

5.2 Instantaneous quantization

5.2.1 Parameters of a quantizer

We consider a sample of a discrete-time random process $s(kT_c)$, obtained by sampling the continuous-time process $s(t)$ with rate F_c. To simplify the notation we choose $T_c = 1$, unless otherwise stated.

With reference to the scheme of Figure 5.17, for a quantizer with L output values we have:

- input signal $s(k) \in \Re$;
- quantized signal $s_q(k) \in \mathcal{A}_q = \{\mathcal{Q}_{-L/2}, \ldots, \mathcal{Q}_{-1}, \mathcal{Q}_1, \ldots, \mathcal{Q}_{L/2}\}$; the L values of the alphabet $\mathcal{A}_q$ are called *output levels*;

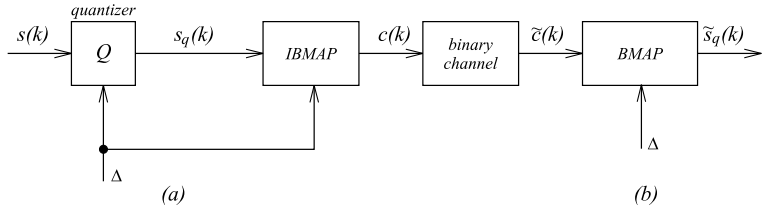


Figure 5.17. Quantization and mapping scheme: (a) encoder, (b) decoder.

- code word $c(k) \in \{0, 1, \ldots, L-1\}$, which represents the value of $s_q(k)$. The system with input $s(k)$ and output $c(k)$ constitutes a PCM encoder.

The quantizer can be described by the function

$$\mathcal{Q} : \Re \longrightarrow \mathcal{A}_q \tag{5.12}$$

For a given partition of the real axis in the intervals $\{R_i\}$, $i = -L/2, \ldots, -1, 1, \ldots, L/2$ such that $\Re = \bigcup_{i=-L/2,\, i\neq 0}^{L/2} R_i$, $R_i \bigcap R_j = \emptyset$ for $i \neq j$, (5.12) implies the following rule

$$\mathcal{Q}[s(k)] = s_q(k) = \mathcal{Q}_i \qquad \text{if } s(k) \in R_i \tag{5.13}$$

A common choice for the decision intervals R_i is given by:

$$\begin{cases} R_i = (\tau_i, \tau_{i+1}] & \text{for } i = -L/2, \ldots, -1 \\ R_i = (\tau_{i-1}, \tau_i] & \text{for } i = 1, \ldots, L/2 \end{cases} \tag{5.14}$$

where $\tau_{-L/2} = -\infty$ and $\tau_{L/2} = \infty$. We note that the *decision thresholds* $\{\tau_i\}$ are $L-1$, being $\tau_{-L/2}$ and $\tau_{L/2}$ assigned. The mapping rule (5.12) is called the *quantizer characteristic* and is illustrated in Figure 5.18 for $L = 8$ and $\tau_0 = 0$. The L values of $s_q(k)$ can be represented by integers $c(k) \in \{0, 1, \ldots, L-1\}$ or by a binary representation with $\lceil \log_2 L \rceil$ bits. For the quantizer characteristic of Figure 5.18, a binary representation is adopted that goes from 000 (the minimum level), to 111 (the maximum level); in this example the bit rate of the system is equal to $F_b = 3F_c$ bit/s.

Let

$$e_q(k) = s_q(k) - s(k) \tag{5.15}$$

be the quantization error. From the relation $s_q(k) = s(k) + e_q(k)$ we have that the quantized signal is affected by a certain error $e_q(k)$. We can formulate the problem as that of representing $s(k)$ with the minimum number of bits $\mathfrak{b}$, to minimize the system bit rate, and at the same time constraining the quantization error, so that a certain level of quality of the quantized signal is maintained.

Observation 5.1
In this chapter the notation $c(k)$ is used to indicate both an integer number and its vectorial binary representation (see (5.18)). Furthermore, in the context of vector quantization the elements of the set $\mathcal{A}_q$ are called code words.

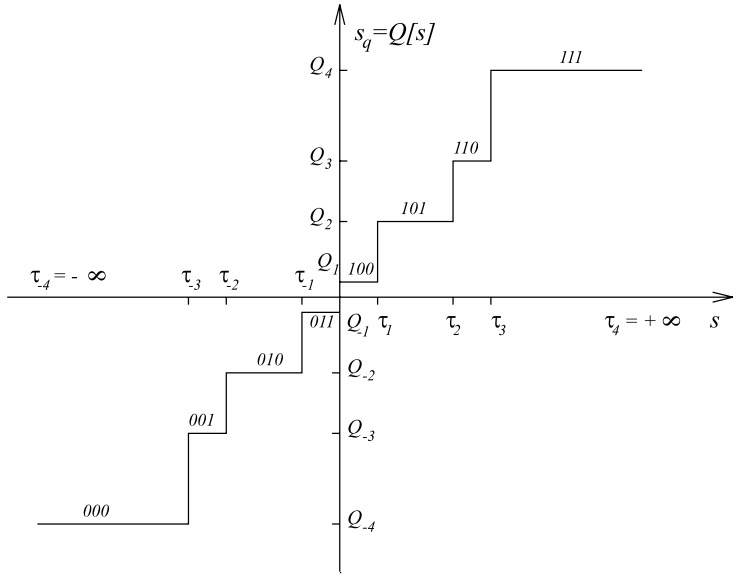


Figure 5.18. Three-bit quantizer characteristic.

5.2.2 Uniform quantizers

A quantizer with $L = 2^{\mathrm{b}}$ equally spaced output levels and decision thresholds is called *uniform*. For:

$$\begin{cases} \tau_{i+1} - \tau_i = \Delta & i = -L/2 + 1, \ldots, -1 \\ \tau_i - \tau_{i-1} = \Delta & i = 1, 2, \ldots, L/2 - 1 \end{cases} \tag{5.16}$$

$$\begin{cases} \mathcal{Q}_{i+1} - \mathcal{Q}_i = \Delta & i = -L/2, \ldots, -2 \\ \mathcal{Q}_1 - \mathcal{Q}_{-1} = \Delta & \\ \mathcal{Q}_i - \mathcal{Q}_{i-1} = \Delta & i = 2, \ldots, L/2 \end{cases} \tag{5.17}$$

where Δ is the *quantization step size*. Two types of characteristics are distinguished, *mid-tread* and *mid-riser*, depending on whether the zero output level belongs or not to $\mathcal{A}_q$.

Mid-riser characteristic. The quantizer characteristic is given in Figure 5.19 for $L = 8$: in this case the smallest value, in magnitude, assumed by $s_q(k)$ is $\Delta/2$, even for a very small input value s.

Let the binary representation of $c(k)$ be defined according to the following rule: the most significant bit of the binary representation of $c(k)$ denotes the sign (± 1) of the input value, whereas the remaining bits denote the amplitude.

Therefore adopting the binary vector representation

$$c(k) = [c_{\mathrm{b}-1}(k), \ldots, c_0(k)] \qquad c_j(k) \in \{0, 1\} \tag{5.18}$$

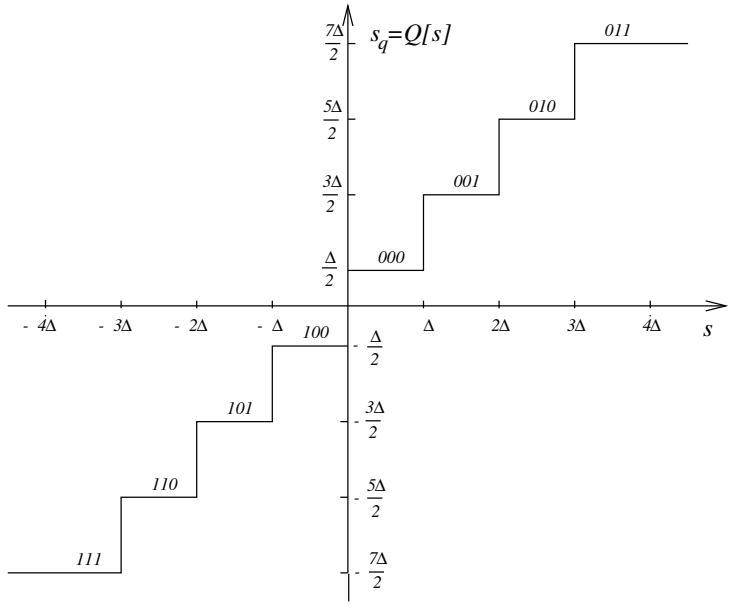


Figure 5.19. Uniform quantizer with mid-riser characteristic ($\mathtt{b} = 3$).

the relation between $s_q(k)$ and $c(k)$ is given by

$$s_q(k) = \Delta(1 - 2c_{\mathtt{b}-1}(k)) \sum_{j=0}^{\mathtt{b}-2} c_j(k)\, 2^j + \frac{\Delta}{2}(1 - 2c_{\mathtt{b}-1}(k)) \tag{5.19}$$

Mid-tread characteristic. The quantizer characteristic is shown in Figure 5.20. Zero is a value assumed by s_q. Let the binary representation of $c(k)$ be the two's complement representation of the level number. Then we have $s_q(k) = \Delta\, c(k)$. Note that the characteristic is asymmetric around zero, hence we may use $L - 1$ levels (giving up the minimum output level), or choose an implementation that can be slightly more complicated than in the case of a symmetric characteristic (see page 357).

Quantization error

We will refer to symmetrical quantizers, with mid-riser characteristic. An example with $L = 2^3 = 8$ levels is given in Figure 5.21: in this case the decision thresholds are $\tau_i = i\,\Delta$, $i = -L/2 + 1, \ldots, -1, 0, 1, \ldots, L/2 - 1$, with, as usual, $\tau_{-L/2} = -\infty$ and $\tau_{L/2} = \infty$. The output values are given by

$$\mathcal{Q}_i = \begin{cases} \left(i + \dfrac{1}{2}\right)\Delta & i = -L/2, \ldots, -1 \\ \left(i - \dfrac{1}{2}\right)\Delta & i = 1, \ldots, L/2 \end{cases} \tag{5.20}$$

Correspondingly the decision intervals are given by (5.14).

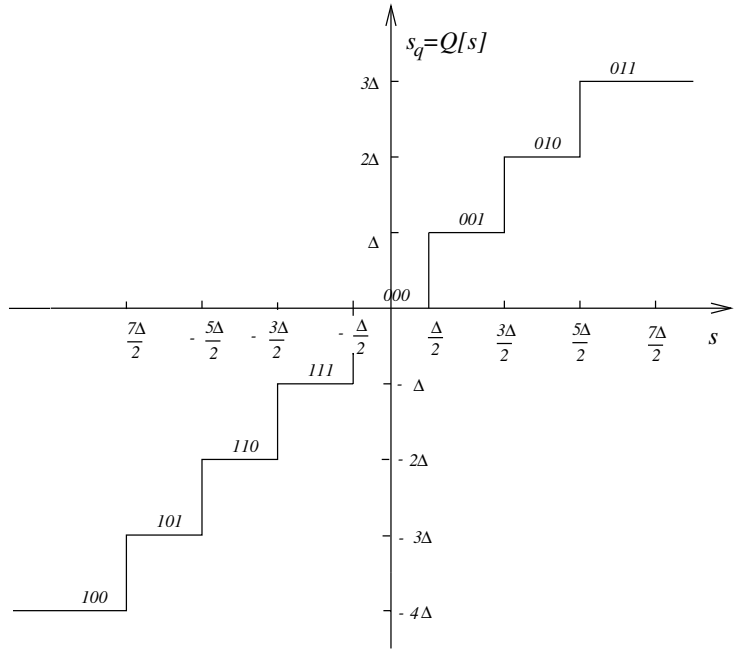


Figure 5.20. Uniform quantizer with mid-tread characteristic ($\mathtt{b} = 3$).

We note that if $s_q(k) = \mathcal{Q}_i$, then the $\mathtt{b} - 1$ least significant bits of $c(k)$ are given by the binary representation of $(|i| - 1)$, and $c(k)$ assumes amplitude values that go from 0 to $L/2 - 1 = 2^{\mathtt{b}-1} - 1$.

If for each value of s we compute the corresponding error $e_q = \mathcal{Q}(s) - s$, we obtain the quantization error characteristic of Figure 5.21. We define the quantizer *saturation value* as

$$\tau_{sat} = \tau_{(L/2)-1} + \Delta \tag{5.21}$$

that is shifted by Δ with respect to the last finite threshold value. Then we have

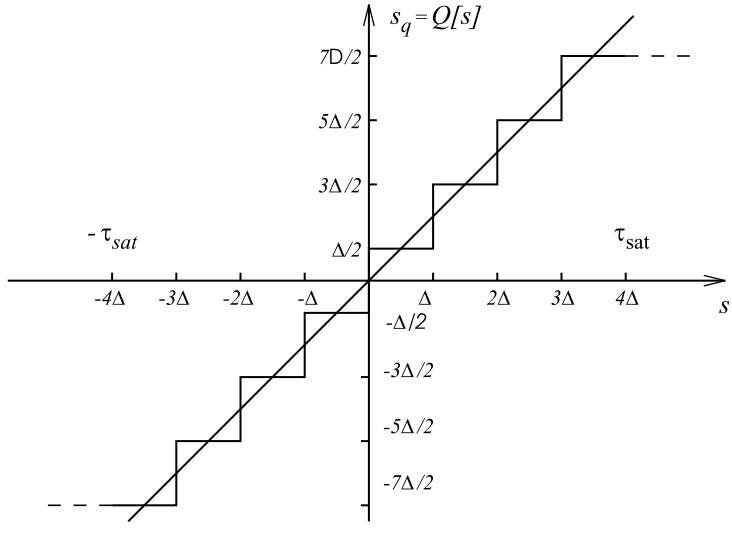
$$|e_q| \leq \frac{\Delta}{2} \qquad \text{for } |s| < \tau_{sat} \tag{5.22}$$

and

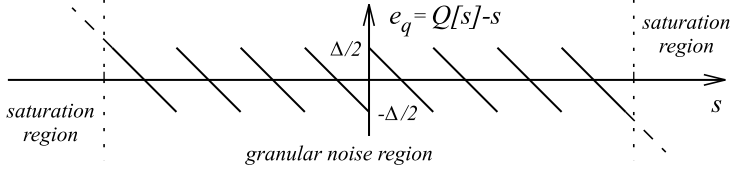
$$e_q = \begin{cases} \mathcal{Q}_{L/2} - s & \text{for } s > \tau_{sat} \\ \mathcal{Q}_{-L/2} - s & \text{for } s < -\tau_{sat} \end{cases} \tag{5.23}$$

Consequently, e_q may assume large values if $|s| > \tau_{sat}$. This observation suggests that the real axis be divided into two parts:

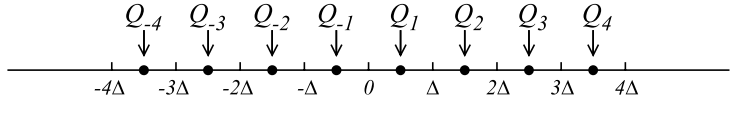
1. the region $s \in (-\infty, -\tau_{sat}) \cup (\tau_{sat}, +\infty)$, where e_q is called *saturation* or *overload error* (e_{sat});



(a) Uniform quantizer characteristic



(b) Quantization error characteristic



(c) Overall representation

Figure 5.21. Uniform quantizer ($b = 3$).

2. the region $s \in [-\tau_{sat}, \tau_{sat}]$, where e_q is called *granular error* (e_{gr}); the interval $[-\tau_{sat}, \tau_{sat}]$ is also called *quantizer range*.

It is often useful to compactly represent the quantizer characteristic in a single axis, as illustrated in Figure 5.21c, where the values of the decision thresholds are indicated by dashed lines, and the quantizer output values by dots.

Relation between Δ, $\mathtt{b}$ and τ_{sat}

The quantization step size Δ is chosen so that

$$2\tau_{sat} = L\Delta \tag{5.24}$$

Therefore, for $L = 2^{\mathtt{b}}$,

$$\Delta = \frac{2\tau_{sat}}{2^{\mathtt{b}}} \tag{5.25}$$

If $|s(k)| < \tau_{sat}$, observing (5.22) this choice guarantees that e_q is granular with amplitude in the range

$$-\frac{\Delta}{2} \leq e_q(k) \leq \frac{\Delta}{2} \tag{5.26}$$

If $|s(k)| > \tau_{sat}$ the saturation error can assume large values: therefore τ_{sat} must be chosen so that the probability of the event $|s(k)| > \tau_{sat}$ is small. For a fixed number of bits $\mathtt{b}$, and consequently for a fixed number of levels L, it is important to verify that, increasing τ_{sat}, Δ also increases and hence also the *granular error*; on the other hand, choosing a small Δ leads to a considerable *saturation error*. As a result, for each value of $\mathtt{b}$ there will be an optimum choice of τ_{sat} and hence of Δ. In any case, to decrease both errors we must increase $\mathtt{b}$ with consequent increase of the encoder bit rate.

Statistical description of the quantization noise

In Figure 5.22 we give an equivalent model of a quantizer where the quantization error is modeled as additive noise. Assuming (5.26) holds, that is for *granular* e_q, we make the following assumptions.

1. The quantization error is white,

$$E[e_q(k)e_q(k-n)] = \begin{cases} \mathtt{M}_{e_q} & n = 0 \\ 0 & n \neq 0 \end{cases} \tag{5.27}$$

2. It is uncorrelated with the input signal

$$E[s(k)e_q(k-n)] = 0 \qquad \forall n \tag{5.28}$$

3. It has a uniform distribution (see Figure 5.23):

$$p_{e_q}(a) = \frac{1}{\Delta} \qquad -\frac{\Delta}{2} \leq a \leq \frac{\Delta}{2} \tag{5.29}$$

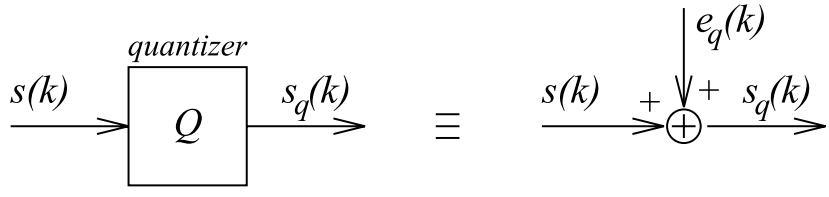


Figure 5.22. Equivalent model of a quantizer.

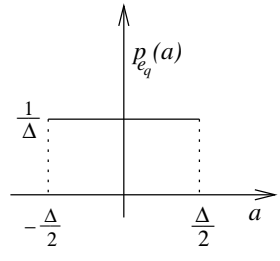


Figure 5.23. Probability density function of e_q.

We note that if $s(k)$ is a constant signal the above assumptions are not true; they hold in practice if $\{s(k)\}$ is described by a function that significantly deviates from a constant and Δ is adequately small, that is $\mathtt{b}$ is large. Figure 5.24 illustrates the quantization error for a 16-level quantized signal. The signal $e_q(t)$ is quite different from $s(t)$ and the above assumptions are plausible.

If the probability density function of the signal to quantize is known, letting g denote the function that relates s and e_q, that is $e_q = g(s)$, also called quantization error characteristic, the probability density function of the noise is obtained as an application of the theory of functions of a random variable, that yields

$$p_{e_q}(a) = \sum_{b \in g^{-1}(a)} \frac{p_s(b)}{|g'(b)|} \qquad -\frac{\Delta}{2} < a < \frac{\Delta}{2} \tag{5.30}$$

where $g^{-1}(\cdot)$ is the inverse of the error function, or equivalently the set of values of s corresponding to a given value of e_q. We note that in this case the slope of the function g is always equal to one, hence $g'(b) = 1$, and from (5.15) for $-\Delta/2 < a < \Delta/2$ we get

$$g^{-1}(a) = \left\{ \mathcal{Q}_i - a,\ i = -\frac{L}{2}, \ldots, -1, 1, \ldots, \frac{L}{2} \right\} \tag{5.31}$$

Finally,

$$p_{e_q}(a) = \sum_{i=-\frac{L}{2},\, i \neq 0}^{\frac{L}{2}} p_s(\mathcal{Q}_i - a) \qquad -\frac{\Delta}{2} < a < \frac{\Delta}{2} \tag{5.32}$$

It can be shown that, if Δ is small enough, the sum in (5.32) gives origin to a uniform function p_{e_q}, independently of the form of p_s.

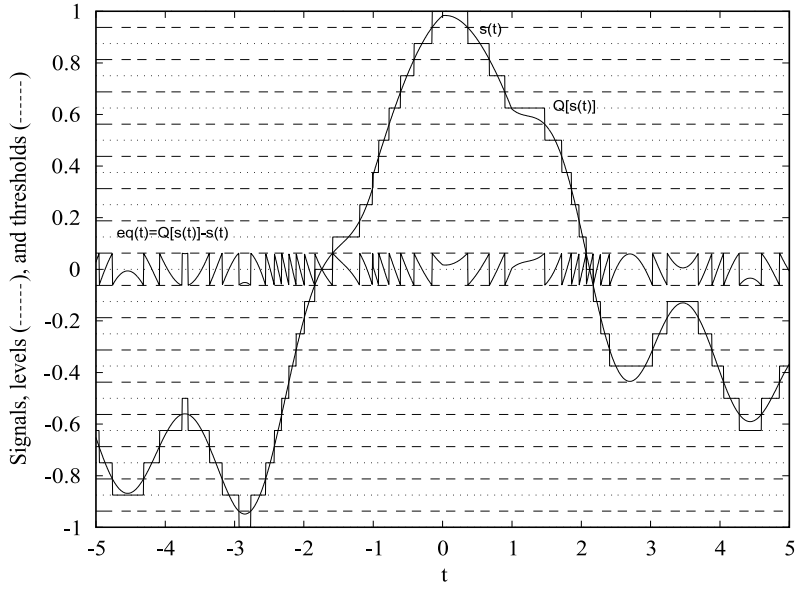


Figure 5.24. Quantization error, $L = 16$ levels.

Statistical power of the quantization error

With reference to the model of Figure 5.22, a measure of the quality of a quantizer is the signal-to-quantization error ratio:

$$\Lambda_q = \frac{E[s^2(k)]}{E[e_q^2(k)]} \tag{5.33}$$

Choosing τ_{sat} so that e_q is granular, from (5.29) we get

$$\mathtt{M}_{e_q} \simeq \mathtt{M}_{e_{gr}} \simeq \frac{\Delta^2}{12} \tag{5.34}$$

For an exact computation that includes also the saturation error we need to know the probability density function of s. The statistical power of e_q is given by

$$\begin{aligned} \mathtt{M}_{e_q} &= E[e_q^2(k)] = \int_{-\infty}^{+\infty} [\mathcal{Q}(a) - a]^2 p_s(a)\, da \\ &= \int_{-\tau_{sat}}^{\tau_{sat}} [\mathcal{Q}(a) - a]^2 p_s(a)\, da + \int_{-\infty}^{-\tau_{sat}} [\mathcal{Q}(a) - a]^2 p_s(a)\, da \\ &\quad + \int_{\tau_{sat}}^{\infty} [\mathcal{Q}(a) - a]^2 p_s(a)\, da\,. \end{aligned} \tag{5.35}$$

In (5.35) the first term is the statistical power of the granular error, $\mathtt{M}_{e_{gr}}$, and the other two terms express the statistical power of the saturation error, $\mathtt{M}_{e_{sat}}$. Let us assume that $p_s(a)$

is even and the characteristic is symmetrical, i.e. $\tau_{-i} = -\tau_i$ and $\mathcal{Q}_{-i} = -\mathcal{Q}_i$; then we get

$$\mathtt{M}_{e_{gr}} = 2\left\{\sum_{i=1}^{\frac{L}{2}-1}\int_{\tau_{i-1}}^{\tau_i}(\mathcal{Q}_i - a)^2 p_s(a)\,da + \int_{\tau_{\frac{L}{2}-1}}^{\tau_{sat}}(\mathcal{Q}_{\frac{L}{2}} - a)^2 p_s(a)\,da\right\} \tag{5.36}$$

$$\mathtt{M}_{e_{sat}} = 2\int_{\tau_{sat}}^{+\infty}(\mathcal{Q}_{\frac{L}{2}} - a)^2\, p_s(a)\,da \tag{5.37}$$

If the probability of saturation satisfies the relation $P[|s(k)| > \tau_{sat}] \ll 1$, then $\mathtt{M}_{e_{sat}} \simeq 0$; introducing the change of variable $b = \mathcal{Q}_i - a$, as $\tau_i = \mathcal{Q}_i + \Delta/2$ and $\tau_{i-1} = \mathcal{Q}_i - \Delta/2$, we have

$$\mathtt{M}_{e_q} \simeq \mathtt{M}_{e_{gr}} = 2\sum_{i=1}^{L/2}\int_{-\Delta/2}^{\Delta/2} b^2 p_s(\mathcal{Q}_i - b)\,db \tag{5.38}$$

If Δ is small enough, then

$$p_s(\mathcal{Q}_i - b) \simeq p_s(\mathcal{Q}_i) \quad \text{for } |b| \le \frac{\Delta}{2} \tag{5.39}$$

and assuming $2(\sum_{i=0}^{L/2} p_s(\mathcal{Q}_i)\Delta) \simeq \int_{-\infty}^{+\infty} p_s(b)\,db = 1$, we get

$$\mathtt{M}_{e_{gr}} = 2\left(\sum_{i=1}^{L/2} p_s(\mathcal{Q}_i)\Delta\right)\int_{-\Delta/2}^{\Delta/2}\frac{b^2}{\Delta}\,db \simeq \frac{\Delta^2}{12} \tag{5.40}$$

In conclusion, as in (5.34), we have

$$\mathtt{M}_{e_q} \simeq \frac{\Delta^2}{12} \tag{5.41}$$

assuming that τ_{sat} is large enough, so that the saturation error is negligible, and Δ is sufficiently small to verify (5.39).

Design of a uniform quantizer

Assuming the input $s(k)$ has zero mean and variance σ_s^2, and defining the parameter[5]

$$k_f = \frac{\sigma_s}{\tau_{sat}} \tag{5.42}$$

the procedure of designing a uniform quantizer consists of three steps.

1. Determine τ_{sat} so that the saturation probability is sufficiently small:

$$P_{sat} = P[|s(k)| > \tau_{sat}] \ll 1 \tag{5.43}$$

[5] Often the inverse $1/k_f = \tau_{sat}/\sigma_s$ is called *loading factor*.

For example, if $s(k) \in \mathcal{N}\left(0, \sigma_s^2\right)$, then[6]

$$P_{sat} = 2Q\left(\frac{\tau_{sat}}{\sigma_s}\right) = \begin{cases} 0.046 & \dfrac{\tau_{sat}}{\sigma_s} = 2 \\ 0.0027 & \dfrac{\tau_{sat}}{\sigma_s} = 3 \\ 0.000063 & \dfrac{\tau_{sat}}{\sigma_s} = 4 \end{cases}$$

2. Choose L so that the signal-to-quantization error ratio assumes a desired value

$$\Lambda_q = \frac{\mathtt{M}_s}{\mathtt{M}_{e_q}} \simeq \frac{\sigma_s^2}{\Delta^2/12} = 3k_f^2 L^2 \tag{5.44}$$

3. Given L and k_f, we obtain

$$\Delta = \frac{2\tau_{sat}}{L} = \frac{2\sigma_s}{k_f L} \tag{5.45}$$

Signal-to-quantization error ratio

For $L = 2^{\mathtt{b}}$, observing (5.44) we have the following result

$$(\Lambda_q)_{dB} \simeq 6.02\,\mathtt{b} + 4.77 + 20\log\left(\frac{\sigma_s}{\tau_{sat}}\right) \tag{5.46}$$

Recalling that this law considers only granular error, if we double the number of quantizer levels for a given loading factor, i.e. increase by one the number of bits $\mathtt{b}$, the signal-to-quantization error ratio increases by 6 dB.

Example 5.2.1
Let $s(k) \in \mathcal{U}[-s_{max}, s_{max}]$. Setting $\tau_{sat} = s_{max}$, we get

$$\frac{\tau_{sat}}{\sigma_s} = \frac{s_{max}}{\sigma_s} = \sqrt{3} \quad \Longrightarrow \quad (\Lambda_q)_{dB} = 6.02\,\mathtt{b} \tag{5.47}$$

Example 5.2.2
Let $s(k) = s_{max}\cos(2\pi f_0 T_c k + \varphi)$. Setting $\tau_{sat} = s_{max}$, we get

$$\frac{\tau_{sat}}{\sigma_s} = \frac{s_{max}}{\sigma_s} = \sqrt{2} \quad \Longrightarrow \quad (\Lambda_q)_{dB} = 6.02\,\mathtt{b} + 1.76 \tag{5.48}$$

Example 5.2.3
For $s(k)$ not limited in amplitude, and assuming P_{sat} negligible for $\tau_{sat} = 4\sigma_s$, we get

$$(\Lambda_q)_{dB} = 6.02\,\mathtt{b} - 7.2 \tag{5.49}$$

[6] The function Q is defined in Appendix 6.A.

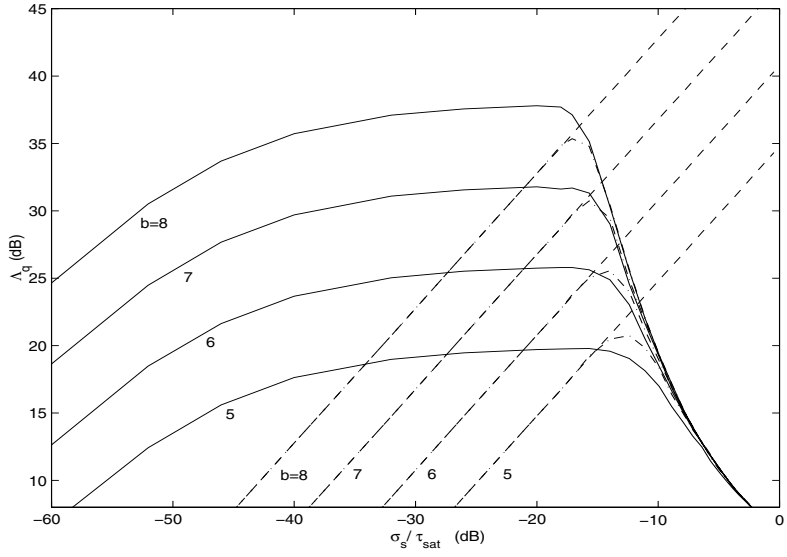


Figure 5.25. Signal-to-quantization error ratio versus σ_s/τ_{sat} of a uniform quantizer for granular noise only (dashed lines), for a Laplacian signal (dashed-dotted lines), and of a μ-law ($\mu = 255$) quantizer (continuous lines). The parameter b is the number of bits of the quantizer. The expression of Λ_q for granular noise only is given by (5.46) and (5.64) for a uniform and a μ-law quantizer, respectively.

The plot of Λ_q, given by (5.46), versus the statistical power of the input signal is illustrated in Figure 5.25 for various values of b. We note that for values of σ_s near τ_{sat} the approximation $M_{e_q} \simeq M_{e_{gr}}$ is no longer valid because $M_{e_{sat}}$ becomes non-negligible. For the computation of $M_{e_{sat}}$ we need to know the probability density function of s and apply (5.35). Assuming a Laplacian signal we obtain the curves also shown in Figure 5.25, that coincide with the curves given by (5.46) for $\sigma_s \ll \tau_{sat}$.

The optimization of Λ_q for the uniform quantization of a signal with a specified amplitude distribution yields the results given in Table 5.5 [4]. We note that for the more dispersive inputs the optimum value of Δ increases, and consequently the value of Λ_q decreases. We also note that the quantizers obtained by the optimization procedure and by the method on page 353 are in general different.

Example 5.2.4

For $s(k) \in \mathcal{N}\left(0, \sigma_s^2\right)$ and $b = 5$, observing Table 5.5 we have optimum performance for $\Delta/\sigma_s = 0.1881$, and consequently $\tau_{sat} = 2^{b-1}\Delta = 3.05\sigma_s$. As shown in Figure 5.26, the optimum value of Λ_q is obtained by determining the minimum of $(M_{e_{gr}} + M_{e_{sat}})/M_s$ as a function of σ_s/τ_{sat}. The optimum point depends on b: we have $\tau_{sat} = \ell\sigma_s$, where ℓ increases with b; in particular for $b = 3$ it turns out $\tau_{sat} = 2.3\,\sigma_s$, whereas for $b = 8$ we obtain $\tau_{sat} = 3.94\,\sigma_s$.

Table 5.5 Optimal quantization step size and maximum corresponding value Λ_q of a uniform quantizer for different $p_S(a)$ (**U**: uniform, **G**: Gaussian, **L**: Laplacian, **Γ**: gamma). [*From Jayant and Noll (1984).*]

b (bit/sample)	Δ_{opt}/σ_s				$\max(\Lambda_q)_{dB}$			
	$p_s(a)$				$p_s(a)$			
	U	**G**	**L**	**Γ**	**U**	**G**	**L**	**Γ**
1	1.7320	1.5956	1.4142	1.1547	6.02	4.40	3.01	1.76
2	0.8660	0.9957	1.0874	1.0660	12.04	9.25	7.07	4.95
3	0.4330	0.5860	0.7309	0.7957	18.06	14.27	11.44	8.78
4	0.2165	0.3352	0.4610	0.5400	24.08	19.38	15.96	13.00
5	0.1083	0.1881	0.2800	0.3459	30.10	24.57	20.60	17.49
6	0.0541	0.1041	0.1657	0.2130	36.12	29.83	25.36	22.16
7	0.0271	0.0569	0.0961	0.1273	42.14	35.13	30.23	26.99
8	0.0135	0.0308	0.0549	0.0743	48.17	40.34	35.14	31.89

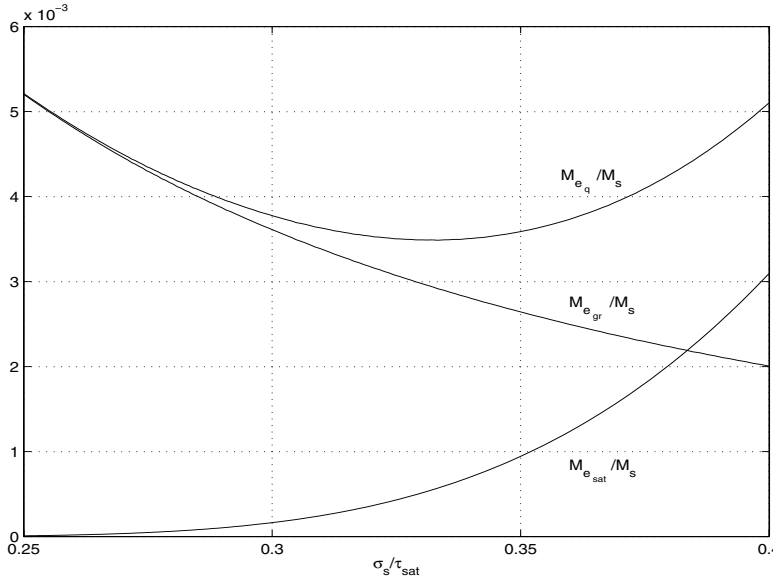


Figure 5.26. Determination of the optimum value of Λ_q for $\mathtt{b} = 5$ and $s(k) \in \mathcal{N}(0, \sigma_s^2)$.

We conclude this section observing that for a non-stationary signal, for example, a voice signal, setting $\tau_{sat} = 4\sigma_s$, where σ_s^2 is computed for a *voiced* spurt, yields $\Lambda_q \simeq 33\,\text{dB}$ for $\mathtt{b} = 7$, good enough for telephone communications. However, in an *unvoiced* spurt σ_s^2 can be reduced by 20–30 dB, and consequently Λ_q is degraded by an amount equivalent to 3–5 bit.

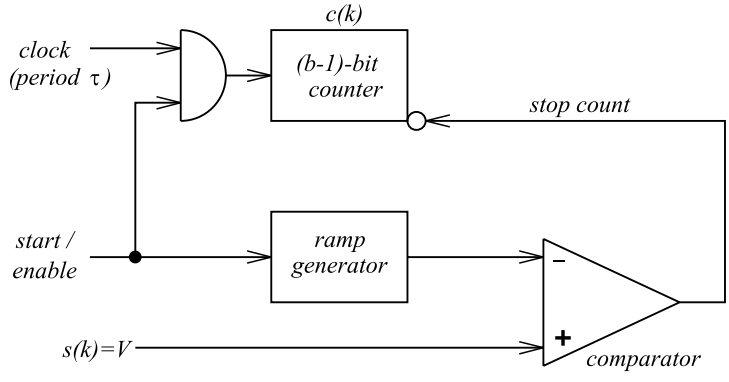


Figure 5.27. Uniform PCM encoder: encoding one level at a time.

Implementations of uniform PCM encoders

We now give three possible implementations of PCM encoders.

1. The first implementation encodes one level at a time and is illustrated in Figure 5.27. Set $V = |s(k)|$. The sign of $s(k)$ can be encoded as a separate bit. V is compared with the output signal of a ramp generator with slope Δ/τ, where τ is the clock period of a counter with $\mathtt{b}-1$ bits. Starting with the counter initialized to zero, when the generator output signal exceeds the level V, the number of clock periods elapsed from the start represents $c(k)$, which gives the PCM encoding of $|s(k)|$. For example, let us consider the case illustrated in Figure 5.28 for $\mathtt{b} = 3$:

$$\begin{aligned} &\text{if } V < \tau_1 \Rightarrow c(k) = 00 \quad \text{stop} \\ &\text{if } V < \tau_2 \Rightarrow c(k) = 01 \quad \text{stop} \end{aligned}$$

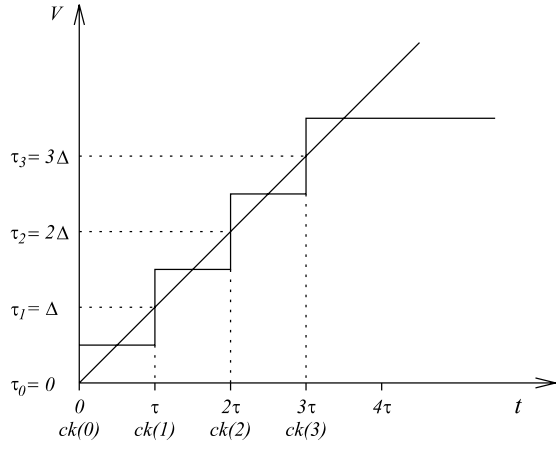


Figure 5.28. Example of encoding one level at a time for $\mathtt{b} = 3$.

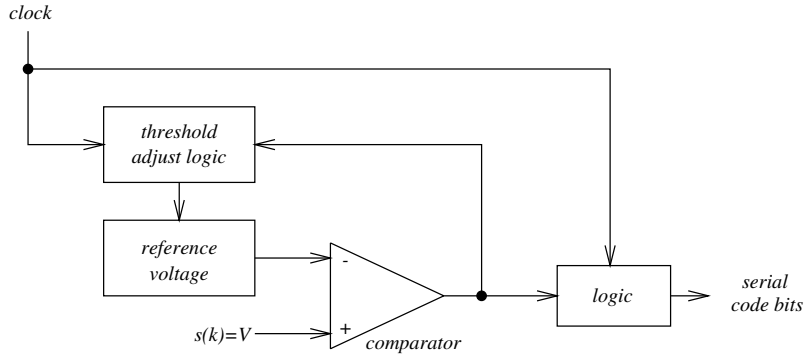


Figure 5.29. PCM encoder: encoding one bit at a time.

$$\text{if } V < \tau_3 \Rightarrow c(k) = 10 \qquad \text{stop}$$
$$\text{if } V > \tau_3 \Rightarrow c(k) = 11 \qquad \text{stop}$$

Generally the number of comparisons depends on V and it is at most equal to $2^{\mathtt{b}-1}$.

2. A second possible implementation, which encodes one bit at a time, is given in Figure 5.29. In this case $\mathtt{b}-1$ comparisons are made: it is as if we were to explore a complete binary tree whose $2^{\mathtt{b}-1}$ leaves represent the output levels. For example, for $\mathtt{b}=3$, neglecting the sign bit, the code word length is 2, and $c(k) = (c_1, c_0)$. To determine the bits c_0 and c_1 we can operate as follows:

$$\text{if } V < \tau_2 \Rightarrow c_1 = 0 \qquad \text{otherwise } c_1 = 1$$
$$\text{if } V < \tau_1 + c_1 2^1 \Delta \Rightarrow c_0 = 0 \qquad \text{otherwise } c_0 = 1$$

Only two comparisons are made, but the decision thresholds now depend on the choice of the previous bits.

3. The last implementation, which encodes one code word of $(\mathtt{b}-1)$ bit at a time, is given in Figure 5.30. In this scheme V is compared simultaneously with the $2^{\mathtt{b}-1}$ quantizer thresholds: the outcome of this comparison is a word of $2^{\mathtt{b}-1}$ bit formed by a sequence of "0" followed by a sequence of "1"; through a logic network this word is mapped to a binary word of $\mathtt{b}-1$ bits that yields the PCM encoding of $s(k)$. These encoders are called flash converters.

We conclude this section explaining that the acronym PCM stands for *pulse code modulation*. We waited until the end of the section to avoid confusion about the term *modulation*: in fact, PCM is not a modulation, but rather a coding method.

5.3 Non-uniform quantizers

There are two observations that suggest the choice of a non-uniform quantizer. The first refers to stationary signals with a non-uniform probability density function: for such signals

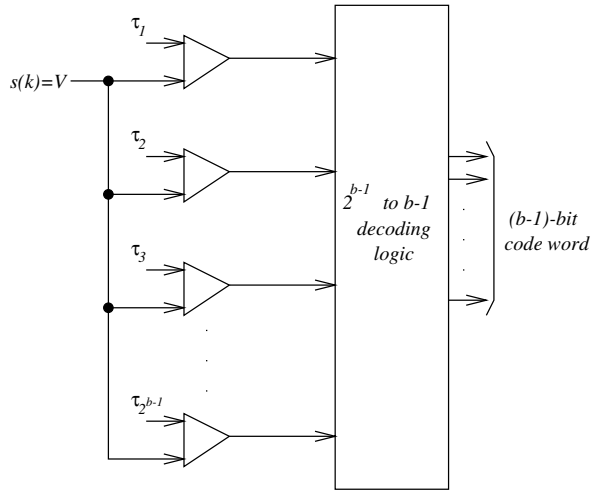


Figure 5.30. Flash converter: encoding one word at a time.

uniform quantizers are suboptimum. The second refers to non-stationary signals, e.g., speech, for which the ratio between instantaneous power (estimated over windows of tenths of milliseconds) and average power (estimated over the whole signal) can exhibit variations of several dB; moreover, the variation of the average power over different links is also of the order of 40 dB. Under these conditions a quantizer with non-uniform characteristics, as that depicted for example in Figure 5.31, is more effective because the signal-to-quantization error ratio Λ_q is almost independent of the instantaneous power. As also illustrated in Figure 5.31, for a non-uniform quantizer the quantization error is large if the signal is large, whereas it is small if the signal is small: as a result the ratio Λ_q tends to remain constant for a wide dynamic range of the input signal.

Three examples of implementation

1. The characteristic of Figure 5.31 can be implemented directly, for example, with the techniques illustrated in Figures 5.29 and 5.30.

2. As shown in Figure 5.32, a *compression* function may precede a uniform quantizer: at the decoder it is therefore necessary to have an *expansion* of the quantized signal.

3. The most popular method, depicted in Figure 5.33, employs a uniform quantizer having a large number of levels, with a step size equal to the minimum step size of the desired non-uniform characteristic. Encoding of the non-uniformly quantized signal y_q is obtained by a look-up table whose input is the uniformly quantized value x_q.

In Section 5.3.1 we will analyze in detail the last two methods.

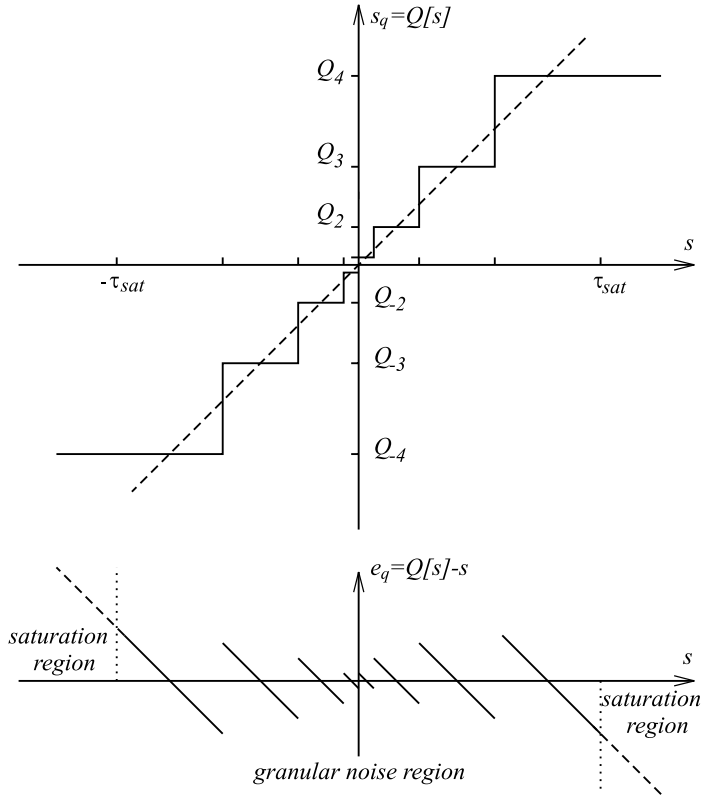


Figure 5.31. Non-uniform quantizer characteristic with $L = 8$ levels.

5.3.1 Companding techniques

Figure 5.32b illustrates in detail the principle of Figure 5.32a. The signal is first compressed through a non-linear function F, that yields the signal

$$y = F(s) \tag{5.50}$$

In Figure 5.32 we assume $\tau_{sat} = 1$. If $\tau_{sat} \neq 1$ we need to normalize s to τ_{sat}. The signal y is uniformly quantized and the code word given by the inverse bit mapper is transmitted. At the receiver the bit mapper gives y_q, that must be expanded to yield a quantized version of s

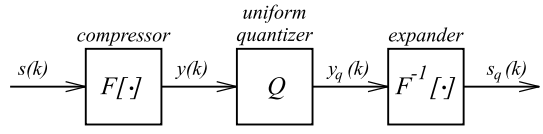
$$s_q = F^{-1}[Q[y]] \tag{5.51}$$

This quantization technique takes the name of *companding* from the steps of *compressing* and *expanding*.

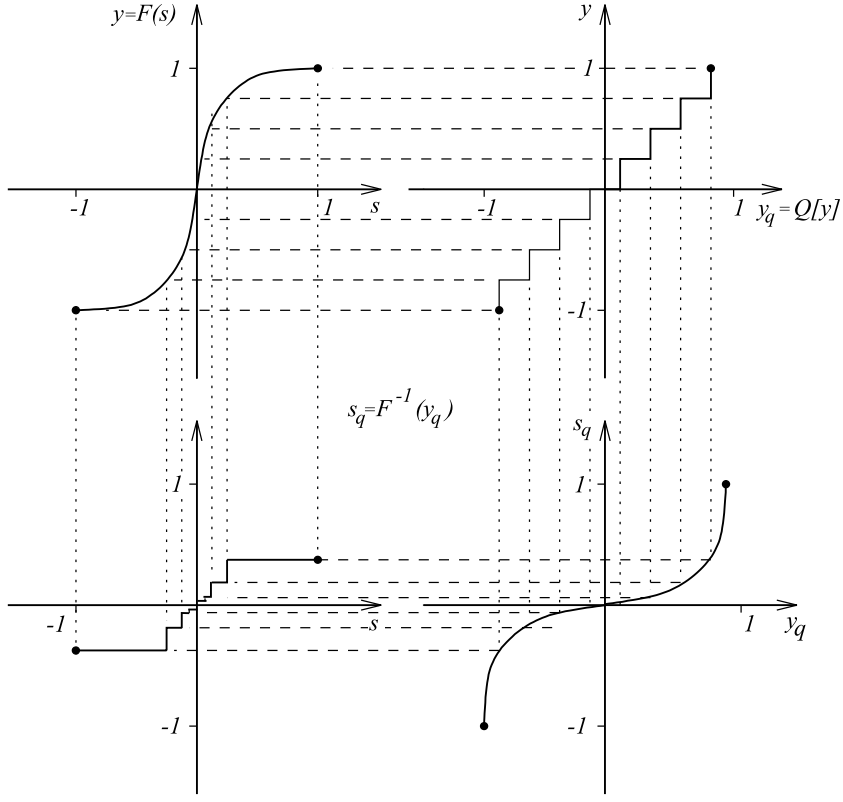
We find that the ideal characteristics of $F[\cdot]$ should be logarithmic,

$$F[s] = \ln s \tag{5.52}$$

We consider the two blocks shown in Figure 5.34.



(a)



(b)

Figure 5.32. (a) Use of a compression function F to implement a non-uniform quantizer; (b) non-uniform quantizer characteristic implemented by companding and uniform quantization. Here $\tau_{\text{sat}} = 1$ is assumed.

Encoding. Let

$$s(k) = e^{y(k)} \operatorname{sgn}[s(k)] \tag{5.53}$$

that is

$$y(k) = \ln |s(k)| \tag{5.54}$$

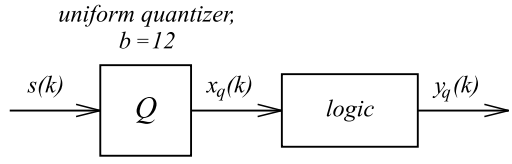


Figure 5.33. Non-uniform quantizer implemented digitally using a uniform quantizer with small step size followed by a look-up table.

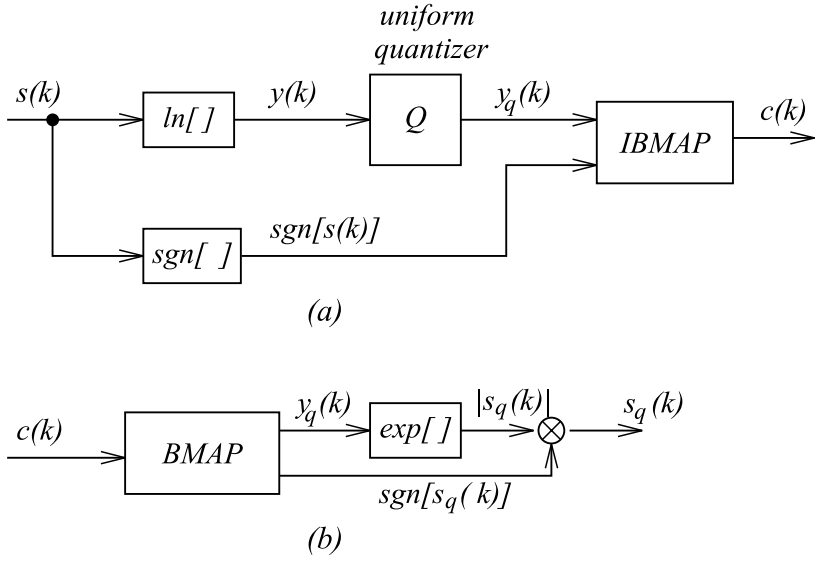


Figure 5.34. Non-uniform quantization by companding and uniform quantization: (a) PCM encoder, (b) decoder.

and assume the sign of the quantized signal is equal to that of $s(k)$. The quantization of $y(k)$ yields

$$y_q(k) = Q[y(k)] = \ln|s(k)| + e_q(k) \tag{5.55}$$

The value $c(k)$ is given by the inverse bit mapping of $y_q(k)$ and the sign of $s(k)$.

Decoder. Assuming $c(k)$ is correctly received, observing (5.55), the quantized version of $s(k)$ is given by

$$\begin{aligned} s_q(k) &= e^{y_q(k)}\,\mathrm{sgn}[s(k)] \\ &= |s(k)|\,\mathrm{sgn}[s(k)]e^{e_q(k)} \\ &= s(k)e^{e_q(k)} \end{aligned} \tag{5.56}$$

If $e_q \ll 1$, then

$$e^{e_q(k)} \simeq 1 + e_q(k) \tag{5.57}$$

and

$$s_q(k) = s(k) + e_q(k)s(k) \tag{5.58}$$

where $e_q(k)s(k)$ represents the output error of the system. As $e_q(k)$ is uncorrelated with the signal $\ln|s(k)|$, and hence with $s(k)$ (see (5.28)), we get

$$\Lambda_q = \frac{\mathtt{M}_s}{E[e_q^2(k)s^2(k)]} = \frac{1}{E[e_q^2(k)]} = \frac{1}{\mathtt{M}_{e_q}} \tag{5.59}$$

where from (5.41) we have that $\mathtt{M}_{e_q}$ depends only on the quantization step size Δ. Consequently Λ_q does *not* depend on $\mathtt{M}_s$.

We note that a logarithmic compression function generates a signal y with unbounded amplitude, thus an approximation of the logarithmic law is usually adopted.

Regulatory bodies have defined two compression functions:

1. *A-law* ($A = 87.56$). For $\tau_{sat} = 1$,

$$y = F[s] = \begin{cases} \dfrac{A|s|}{1+\ln(A)} & 0 \le |s| \le \dfrac{1}{A} \\ \dfrac{1+\ln(A|s|)}{1+\ln(A)} & \dfrac{1}{A} \le |s| \le 1 \end{cases} \tag{5.60}$$

This law, illustrated in Figure 5.35 for two values of A, is adopted in Europe. The sign is considered separately:

$$\operatorname{sgn}[y] = \operatorname{sgn}[s] \tag{5.61}$$

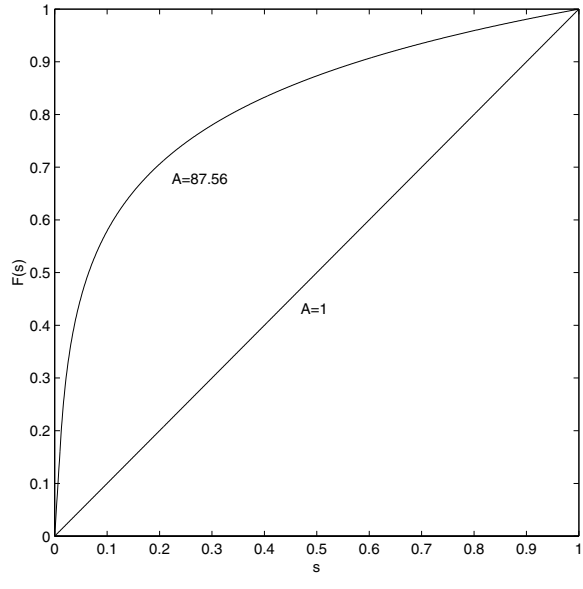


Figure 5.35. *A*-law.

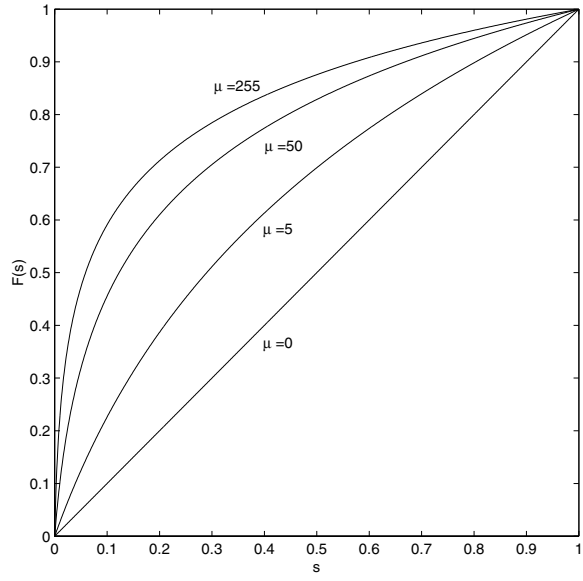


Figure 5.36. μ-law.

2. *μ-law* ($\mu = 255$). For $\tau_{sat} = 1$,

$$y = F[s] = \frac{\ln(1 + \mu|s|)}{\ln(1 + \mu)} \tag{5.62}$$

This law, illustrated in Figure 5.36 for four values of μ, is adopted in the United States and Canada. The compression increases for higher values of μ; the standard value of μ is equal to 255.
We note that, for $\mu\, s \gg 1$, we have

$$F[s] = \frac{\ln[\mu\, s]}{\ln(1 + \mu)} \tag{5.63}$$

as in the ideal case (5.54). Similar behavior is exhibited by (5.60).

Signal-to-quantization error ratio

Assuming the quantization error uniform within each decision interval, which is well verified for a uniform input in the interval $[-\tau_{sat}, \tau_{sat}]$, we can see that for μ-law, considering only the granular error, we have

$$(\Lambda_q)_{dB} = 6.02\mathrm{b} + 4.77 - 20\log_{10}[\ln(1 + \mu)] - 10\log_{10}\left\{1 + \left(\frac{\tau_{sat}}{\mu\sigma_s}\right)^2 + \sqrt{3}\left(\frac{\tau_{sat}}{\mu\sigma_s}\right)\right\} \tag{5.64}$$

Curves of Λ_q versus the statistical power of the input signal are plotted for $\mu = 255$ in Figure 5.25. Note that in the saturation region they coincide with the curves obtained for a

uniform quantizer with Laplacian input. We emphasize that also in this case Λ_q increases by 6 dB with the increase of $\mathtt{b}$ by one. We also note that, if $\mathtt{b} = 8$, $\Lambda_q \simeq 38$ dB for a wide range of values of σ_s. An effect not shown in Figure 5.25 is that, by increasing μ, the plot of Λ_q becomes "flatter", but the maximum value decreases.

Observation 5.2
In the standard *non-linear PCM*, a quantizer with 128 levels (7 bit/sample) is employed after the compression; including also the sign we have 8 bit/sample. For a sampling frequency of $F_c = 8$ kHz, this leads to a bit rate of the system equal to $R_b = 64$ kbit/s.

Digital compression

An alternative method to the compression-quantization scheme is illustrated by an example in Figure 5.37. The relation between $s(k)$ and $y_q(k)$ is obtained through a first multi-bit (5 in figure) quantization to generate x_q; then we have a mapping of the 5 bits of x_q to the 3 bits of y_q using the mapper (sign omitted) of Table 5.6.

For decoding, for each code word y_q we select only one code word x_q, which represents the reconstructed value s_q.

Using the standard compression laws, we need to approximate the compression functions by piecewise linear functions, as shown in Figure 5.39. For encoding, a mapper with 12-bit input and 8-bit output is given in Table 5.7. For decoding, we select for each compressed

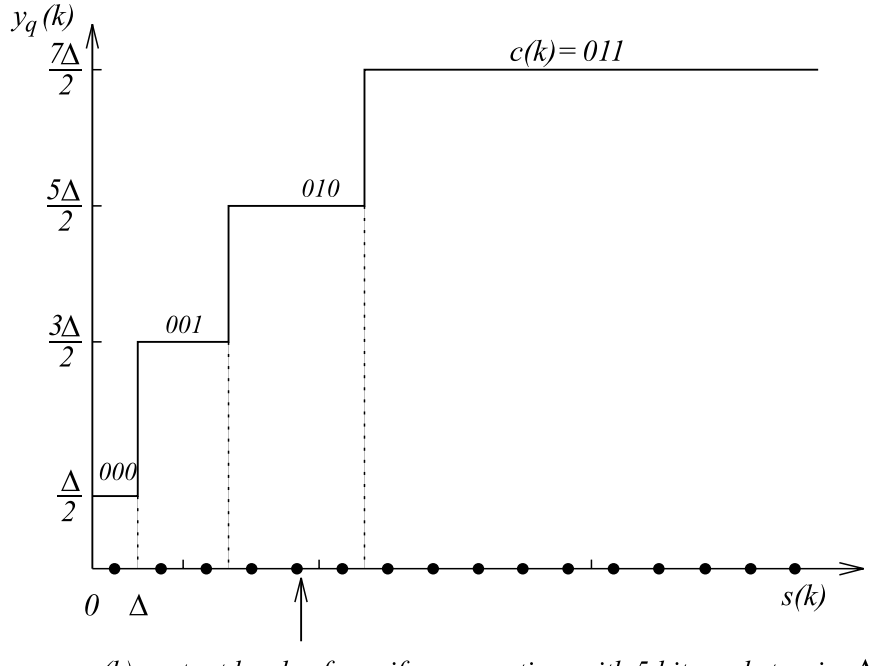


Figure 5.37. Distribution of quantization levels for a 3-bit μ-law quantizer with $\mu = 40$.

Table 5.6 Example of non-linear PCM from 4 to 2 bits (sign omitted).

Coding of x_q	Coding of y_q	Coding of s_q
0000	00	0000
0001	01	0001
0010		
0011		
0100	10	0100
0101		
1000		
1001		
1010		
1011		
1100	11	1011
1101		
1110		
1111		

code word a corresponding linear code word, as given in the third column of Table 5.7. In the literature there are other non-linear PCM tables, that differ in the compression law or in the accuracy of the codes [4].

Signal-to-quantization noise ratio mask

We conclude this section by giving in Figure 5.39 two masks that indicate the minimum tolerable values of Λ_q (dB) for an A-law quantizer ($A = 87.6$), $\tau_{sat} = 3.14\,\text{dBm}$, and $\mathfrak{b} = 8$ (sign included), as a function of σ_s (dBm) for input signals with Gaussian and sinusoidal distribution, respectively; these masks are useful to verify the quantizer performance.

5.3.2 Optimum quantizer in the MSE sense

Assuming we know the probability density function of the input signal $s(k)$, stationary with variance σ_s^2, we desire to determine the parameters of the non-uniform quantizer that optimizes Λ_q. The problem, illustrated in Figure 5.40, consists in choosing the decision thresholds

$$\left\{\tau_{-\left(\frac{L}{2}-1\right)}, \ldots, \tau_{-1}, \tau_0, \tau_1, \ldots, \tau_{\frac{L}{2}-1}\right\} \qquad \tau_{-\frac{L}{2}} = -\infty \quad \tau_{\frac{L}{2}} = +\infty \tag{5.65}$$

and the quantization levels

$$\{\mathcal{Q}_i\} \qquad i = -\frac{L}{2}, \ldots, -1, 1, \ldots, \frac{L}{2} \tag{5.66}$$

that minimize the statistical power of the error (minimum mean-square error criterion)

$$\mathtt{M}_{e_q} = E[(s_q(k) - s(k))^2] = E[(Q[s(k)] - s(k))^2] \tag{5.67}$$

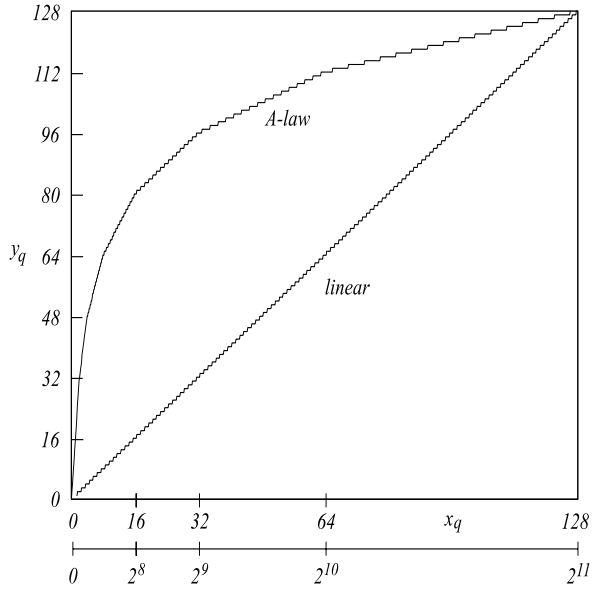


Figure 5.38. Piecewise linear approximation of the A-law compression function ($A = 87.6$). The 12-bit encoded input signals are mapped into 8-bit signals.

Table 5.7 Non linear PCM from 11 to 7 bits (sign omitted).

Linear code (x_q)	Compressed code (y_q)	Coding of s_q
1WXYZ------	111WXYZ	1WXYZ011111
01WXYZ-----	110WXYZ	01WXYZ01111
001WXYZ----	101WXYZ	001WXYZ0111
0001WXYZ---	100WXYZ	0001WXYZ011
00001WXYZ--	011WXYZ	00001WXYZ01
000001WXYZ-	010WXYZ	000001WXYZ0
0000001WXYZ	001WXYZ	0000001WXYZ
0000000WXYZ	000WXYZ	0000000WXYZ

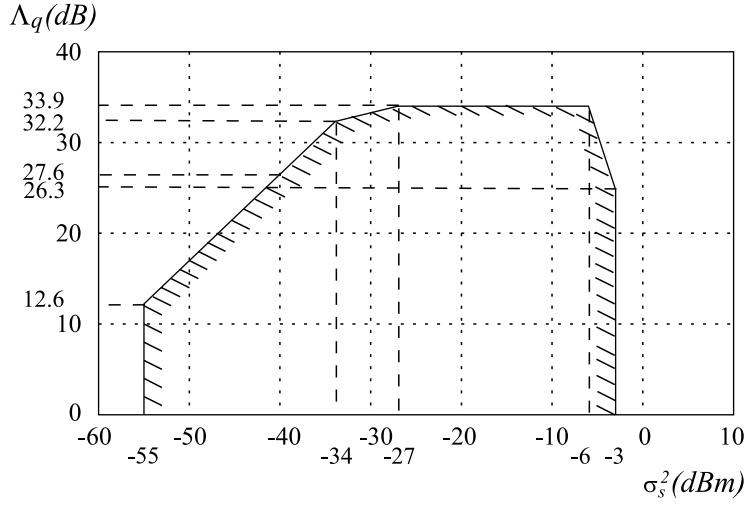
Assuming $p_s(a)$ even, because of the symmetry of the problem we can halve the number of variables to be determined by setting

$$\begin{cases} \tau_{-i} = -\tau_i & i = 1, \ldots, \dfrac{L}{2} - 1 \\ \tau_0 = 0 \end{cases} \tag{5.68}$$

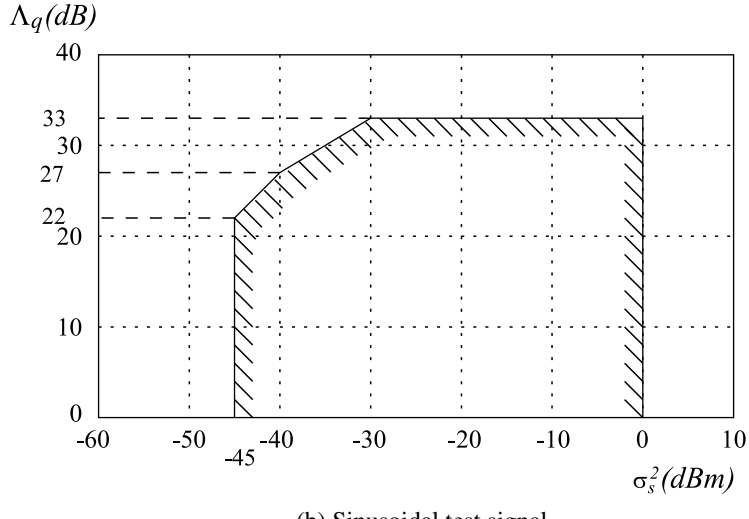
$$\mathcal{Q}_{-i} = -\mathcal{Q}_i \qquad i = 1, \ldots, \frac{L}{2} \tag{5.69}$$

and

$$\mathtt{M}_{e_q} = 2 \sum_{i=1}^{L/2} \int_{\tau_{i-1}}^{\tau_i} (\mathcal{Q}_i - a)^2 p_s(a)\, da \tag{5.70}$$



(a) Gaussian test signal



(b) Sinusoidal test signal

Figure 5.39. Λ_q versus σ_s^2 for an A-law quantizer ($A = 87.56$) and $\mathtt{b} = 8$.

Necessary but not sufficient conditions for minimizing (5.67) are

$$\frac{\partial \mathtt{M}_{e_q}}{\partial \tau_i} = 0 \qquad i = 1, \ldots, \frac{L}{2} - 1 \tag{5.71}$$

$$\frac{\partial \mathtt{M}_{e_q}}{\partial \mathcal{Q}_i} = 0 \qquad i = 1, \ldots, \frac{L}{2} \tag{5.72}$$

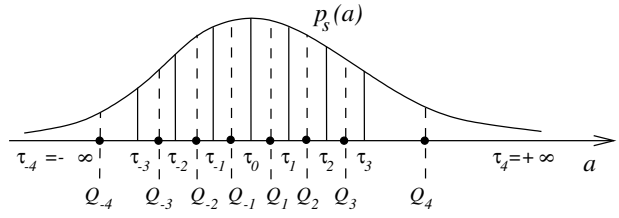


Figure 5.40. Decision thresholds and output levels for a particular $p_S(a)$ ($\mathfrak{b} = 3$).

From

$$\frac{1}{2}\frac{\partial \mathrm{M}_{e_q}}{\partial \tau_i} = (\mathcal{Q}_i - \tau_i)^2 p_s(\tau_i) - (\mathcal{Q}_{i+1} - \tau_i)^2 p_s(\tau_i) \tag{5.73}$$

(5.71) gives

$$p_s(\tau_i)[\mathcal{Q}_i^2 + \tau_i^2 - 2\mathcal{Q}_i\tau_i - \mathcal{Q}_{i+1}^2 - \tau_i^2 + 2\mathcal{Q}_{i+1}\tau_i] = 0 \tag{5.74}$$

that is

$$\tau_i = \frac{\mathcal{Q}_i + \mathcal{Q}_{i+1}}{2} \tag{5.75}$$

Conversely, the equation

$$\frac{1}{2}\frac{\partial \mathrm{M}_{e_q}}{\partial \mathcal{Q}_i} = 2\int_{\tau_{i-1}}^{\tau_i} (\mathcal{Q}_i - a)\, p_s(a)\, da = 0 \tag{5.76}$$

yields

$$\mathcal{Q}_i = \frac{\displaystyle\int_{\tau_{i-1}}^{\tau_i} a p_s(a)\, da}{\displaystyle\int_{\tau_{i-1}}^{\tau_i} p_s(a)\, da} \tag{5.77}$$

In other words, (5.75) establishes that the optimal threshold lies in the middle of the interval between two adjacent output values, and (5.77) sets $\mathcal{Q}_i$ as the centroid of $p_s(\cdot)$ in the interval $[\tau_{i-1}, \tau_i]$. These two rules are illustrated in Figure 5.41.

Max algorithm

We present now the Max algorithm to determine the decision thresholds and the optimum quantization levels.

1. Fixed $\mathcal{Q}_1$ "at random", we use (5.77) to get τ_1 from the integral equation

$$\mathcal{Q}_1 = \frac{\displaystyle\int_{\tau_0}^{\tau_1} a p_s(a)\, da}{\displaystyle\int_{\tau_0}^{\tau_1} p_s(a)\, da} \tag{5.78}$$

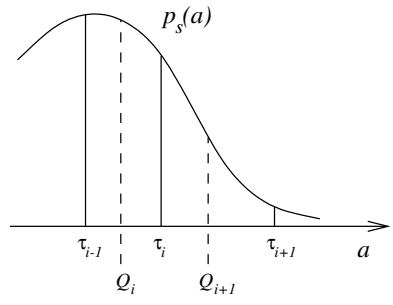


Figure 5.41. Optimum decision thresholds and output levels for a given $p_S(a)$.

2. From (5.75) we obtain $\mathcal{Q}_{i+1} = 2\tau_i + \mathcal{Q}_i$ for $i = 1$.

3. We use (5.77) to obtain τ_{i+1} by the equation

$$\mathcal{Q}_{i+1} = \frac{\displaystyle\int_{\tau_i}^{\tau_{i+1}} a p_s(a)\, da}{\displaystyle\int_{\tau_i}^{\tau_{i+1}} p_s(a)\, da} \tag{5.79}$$

The procedure is iterated for $i = 2, 3, \ldots, (L/2) - 2$. For $i = (L/2) - 1$ we obtain

$$\mathcal{Q}_{\frac{L}{2}} = 2\tau_{\frac{L}{2}-1} + \mathcal{Q}_{\frac{L}{2}-1} \tag{5.80}$$

Now, if $\tau_{L/2} = +\infty$ satisfies the last equation (5.77)

$$\mathcal{Q}_{\frac{L}{2}} = \frac{\displaystyle\int_{\tau_{\frac{L}{2}-1}}^{+\infty} a p_s(a)\, da}{\displaystyle\int_{\tau_{\frac{L}{2}-1}}^{+\infty} p_s(a)\, da} \tag{5.81}$$

then the parameters determined are optimum. Otherwise, if (5.81) is not satisfied we must change our choice of $\mathcal{Q}_1$ in step 1) and repeat the procedure.

Lloyd algorithm

This algorithm uses (5.75) and (5.77), but in a different order.

1. We set a relative error $\epsilon > 0$ and $\mathcal{D}_0 = \infty$.

2. We choose an initial partition of the positive real axis:

$$\mathcal{P}_1 = \{\tau_0, \tau_1, \ldots, \tau_{L/2} = +\infty\} \tag{5.82}$$

such that $\tau_0 = 0 < \tau_1 < \cdots < \tau_{L/2} = +\infty$.

3. We set the iteration index j at 1.
4. We obtain the optimum alphabet $\mathcal{A}_j = \{\mathcal{Q}_1, \ldots, \mathcal{Q}_{L/2}\}$ for the partition $\mathcal{P}_j$ using (5.77).
5. We evaluate the distortion associated with the choice of $\mathcal{P}_j$ and $\mathcal{A}_j$,

$$\mathcal{D}_j = E[e_q^2] = 2\sum_{i=1}^{L/2} \int_{\tau_{i-1}}^{\tau_i} (\mathcal{Q}_i - a)^2 p_s(a)\, da \tag{5.83}$$

6. If

$$\frac{\mathcal{D}_{j-1} - \mathcal{D}_j}{\mathcal{D}_j} < \epsilon \tag{5.84}$$

then we stop the procedure, otherwise we update the value of j: $j \leftarrow j + 1$.

7. We derive the optimum partition $\mathcal{P}_j = \{\tau_0, \tau_1, \ldots, \tau_{L/2} = +\infty\}$ for the alphabet $\mathcal{A}_{j-1}$ using (5.75).
8. We go back to step 4.

We observe that the sequence $\mathcal{D}_j > 0$ is non-increasing: hence the algorithm is converging, however, not necessarily to the absolute minimum, unless some assumptions are made about $p_s(\cdot)$.

Expression of Λ_q for a very fine quantization

For both algorithms it is important to initialize the various parameters near the optimum values. The considerations that follow have this objective, in addition to determining the optimum value of Λ_q for a non-uniform quantizer, at least approximately for a number of bits sufficiently high.

From (5.70), assuming that

$$p_s(a) \simeq p_s(\tau_{i-1}) \quad \text{for } \tau_{i-1} \le a < \tau_i \tag{5.85}$$

we have that

$$\begin{aligned} \mathtt{M}_{e_q} &= 2\sum_{i=1}^{L/2} \int_{\tau_{i-1}}^{\tau_i} (\mathcal{Q}_i - a)^2 p_s(a)\, da \\ &\simeq 2\sum_{i=1}^{L/2} p_s(\tau_{i-1}) \int_{\tau_{i-1}}^{\tau_i} (\mathcal{Q}_i - a)^2\, da \end{aligned} \tag{5.86}$$

If the $\mathcal{Q}_i$s are optimum, it must be

$$\frac{\partial \mathtt{M}_{e_q}}{\partial \mathcal{Q}_i} = 0 \qquad i = 1, \ldots, \frac{L}{2} \tag{5.87}$$

and

$$\mathcal{Q}_i = \frac{\int_{\tau_{i-1}}^{\tau_i} a\,da}{\int_{\tau_{i-1}}^{\tau_i} da} = \frac{\tau_i + \tau_{i-1}}{2} \tag{5.88}$$

Correspondingly, introducing the length of the i-th decision interval

$$(\Delta\tau_i) = \tau_i - \tau_{i-1} \qquad i = 1, \ldots, \frac{L}{2} \tag{5.89}$$

where $\tau_0 = 0$ and $\tau_{L/2} = +\infty$, it follows that

$$\mathtt{M}_{e_q} = 2\sum_{i=1}^{L/2} p_s(\tau_{i-1})\frac{(\Delta\tau_i)^3}{12} \tag{5.90}$$

It is now a matter of finding the minimum of (5.90) with respect to $(\Delta\tau_i)$, with the constraint that the decision intervals cover the whole positive axis; this is obtained by imposing that

$$2\sum_{i=1}^{L/2} p_s^{1/3}(\tau_{i-1})\,(\Delta\tau_i) \simeq 2\int_0^{+\infty} p_s^{1/3}(a)\,da = K \tag{5.91}$$

Using the Lagrange multiplier method, the cost function is

$$\min_{\lambda,\{\Delta\tau_i\}}\left[\mathtt{M}_{e_q} + \lambda\left(K - 2\sum_{i=1}^{L/2} p_s^{1/3}(\tau_{i-1})\,(\Delta\tau_i)\right)\right] \tag{5.92}$$

with $\mathtt{M}_{e_q}$ given by (5.90). By setting to zero the partial derivative of (5.92) with respect to $(\Delta\tau_i)$, we obtain

$$p_s(\tau_{i-1})\frac{(\Delta\tau_i)^2}{4} + \lambda(-p_s^{1/3}(\tau_{i-1})) = 0 \qquad i = 1, \ldots, \frac{L}{2} - 1 \tag{5.93}$$

that yields

$$(\Delta\tau_i) = 2\sqrt{\lambda}\; p_s^{-1/3}(\tau_{i-1}) \tag{5.94}$$

Substituting (5.94) in (5.91) yields

$$\sqrt{\lambda} = \frac{K}{2L} \tag{5.95}$$

hence

$$(\Delta\tau_i) = \frac{K}{L} p_s^{-1/3}(\tau_{i-1}) \qquad i = 1, \ldots, \frac{L}{2} - 1 \tag{5.96}$$

and the minimum value of $\mathtt{M}_{e_q}$ is given by

$$\mathtt{M}_{e_q,opt} = \frac{K^3}{12L^2} \tag{5.97}$$

For a quantizer optimized for a certain probability density function, and for a high number of levels $L = 2^{\mathtt{b}}$ (so that (5.85) holds), we have

$$\Lambda_q = \frac{\mathtt{M}_s}{\mathtt{M}_{e_q,opt}} = \frac{2^{2\mathtt{b}}}{f_f} \tag{5.98}$$

where f_f is a *form factor* related to the amplitude distribution of the normalized signal $\tilde{s}(k) = s(k)/\sigma_s$,

$$f_f = \frac{\tilde{K}^3}{12} \qquad \tilde{K} = \int_{-\infty}^{+\infty} p_{\tilde{s}}^{1/3}(a)\, da \tag{5.99}$$

In the Gaussian case, $s(k) \in \mathcal{N}(0, \sigma_s^2)$, $\tilde{s}(k) \in \mathcal{N}(0, 1)$, and $f_f = 2/(\sqrt{3}\pi)$.

Actually (5.96) indicates that the optimal thresholds are concentrated around the peak of the probability density; moreover, the optimum value of Λ_q, according to (5.98), follows the increment law of 6 dB per bit, as in the case of a quantizer granular error.

Observation 5.3

Equation (5.90) can be used to evaluate approximately $\mathtt{M}_{e_q}$ for a general quantizer characteristic, even of the A-law and μ-law types. In this case, from Figure 5.32, the quantization step size Δ of the uniform quantizer is related to the compression law according to the relation

$$\Delta = F(\tau_i) - F(\tau_{i-1}) \simeq (\Delta\tau_i)\, F'(\tau_{i-1}) \tag{5.100}$$

where F' is the derivative of F. Obviously (5.100) assumes that F' does not vary considerably in the interval $(\tau_{i-1}, \tau_i]$.

Substituting (5.100) in (5.90) we have

$$\mathtt{M}_{e_q} = 2 \sum_{i=1}^{L/2} p_s(\tau_{i-1}) \left[\frac{\Delta}{F'(\tau_{i-1})}\right]^2 \frac{(\Delta\tau_i)}{12} \tag{5.101}$$

For L sufficiently large, the intervals become small and we get

$$\mathtt{M}_{e_q} \simeq \frac{\Delta^2}{12}\, 2 \int_0^{\tau_{sat}} \frac{p_s(a)}{[F'(a)]^2}\, da \tag{5.102}$$

where Δ is related to a uniform quantizer parameters according to (5.25). It is left to the reader to show that for a uniform signal s in $[-\tau_{sat}, \tau_{sat}]$, quantized according to the μ-law, the ratio $\Lambda_q = \mathtt{M}_s/\mathtt{M}_{e_q}$ has the expression given in (5.64).

Performance of non-uniform quantizers

A quantizer takes the name uniform, Gaussian, Laplacian, or gamma, if it is optimized for input signals having the corresponding distribution. In Tables 5.8, 5.9, and 5.10 are given parameter values of three optimum quantizers obtained by the Max or Lloyd method, for Gaussian, Laplacian, and gamma input signal, respectively, and various numbers of levels [4]. Note that, even for a small number of levels, a more dispersive distribution, that is with longer tails, leads to less closely spaced thresholds and levels, and consequently to a decrease of Λ_q.

Concerning the increment of $(\Lambda_q)_{dB}$ according to the 6b law, we show in Figure 5.42 the deviation

$$(\Delta\Lambda_q)_{dB} = 6.02\,\mathtt{b} - \max\{\Lambda_q\}_{dB} \tag{5.103}$$

for both uniform and non-uniform quantizers [4]. The *optimum* value of Λ_q follows the 6b law only in the case of non-uniform quantizers for $\mathtt{b} \geq 4$. In the case of uniform quantizers, with increasing b the maximum of Λ_q occurs for a smaller ratio σ_s/τ_{sat} (due to the saturation error): this makes $\Delta\Lambda_q$ vary with b and in fact it increases.

Finally, we consider what happens if a quantizer, optimized for a specific input distribution, has a different type of input. For example, a uniform quantizer, best

Table 5.8 Optimum quantizers for a signal with Gaussian distribution ($\mathtt{m_S} = 0$, $\sigma_s^2 = 1$). [*From Jayant and Noll (1984).*]

	L							
	2		4		8		16	
i	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$
1	∞	0.798	0.453	0.982	0.501	0.245	0.258	0.128
2			∞	1.510	1.050	0.756	0.522	0.388
3					1.748	1.344	0.800	0.657
4					∞	2.152	1.099	0.942
5							1.437	1.256
6							1.844	1.618
7							2.401	2.069
8							∞	2.733
$\mathtt{M}_{e_q}$	0.363		0.117		0.0345		0.00955	
Λ_q (dB)	4.40		9.30		14.62		20.20	

Table 5.9 Optimum quantizers for a signal with Laplacian distribution ($m_s = 0$, $\sigma_s^2 = 1$). [*From Jayant and Noll (1984).*]

	L							
	2		4		8		16	
i	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$
1	∞	0.707	1.127	0.420	0.533	0.233	0.264	0.124
2			∞	1.834	1.253	0.833	0.567	0.405
3					2.380	1.673	0.920	0.729
4					∞	3.087	1.345	1.111
5							1.878	1.578
6							2.597	2.178
7							3.725	3.017
8							∞	4.432
M_{e_q}	0.500		0.1761		0.0545		0.0154	
Λ_q (dB)	3.01		7.54		12.64		18.12	

Table 5.10 Optimum quantizers for a signal with gamma distribution ($m_s = 0$, $\sigma_s^2 = 1$). [*From Jayant and Noll (1984).*]

	L							
	2		4		8		16	
i	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$	τ_i	$\mathcal{Q}_i$
1	∞	0.577	1.268	0.313	0.527	0.155	0.230	0.073
2			∞	2.223	1.478	0.899	0.591	0.387
3					3.089	2.057	1.051	0.795
4					∞	4.121	1.633	1.307
5							2.390	1.959
6							3.422	2.822
7							5.128	4.061
8							∞	6.195
M_{e_q}	0.6680		0.2326		0.0712		0.0196	
Λ_q (dB)	1.77		6.33		11.47		17.07	

for a uniform input, will have very low performance for an input signal with a very dispersive distribution; on the contrary, a non-uniform quantizer, optimized for a specific distribution, can have even higher performance for a less dispersive input signal.

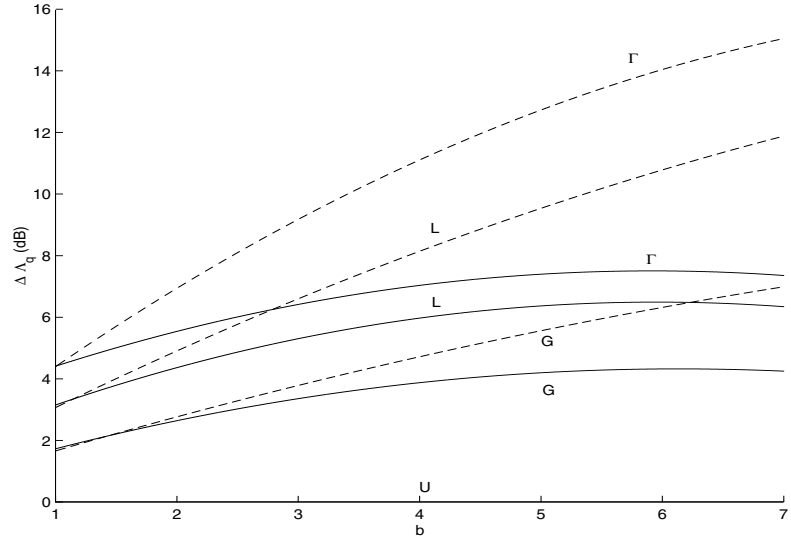


Figure 5.42. Performance comparison of uniform (dashed line) and non-uniform (continuous line) quantizers, optimized for a specific probability density function of the input signal. Input type: uniform (*U*), Laplacian (*L*), Gaussian (*G*) and gamma (Γ) [4]. [*From Jayant and Noll (1984).*]

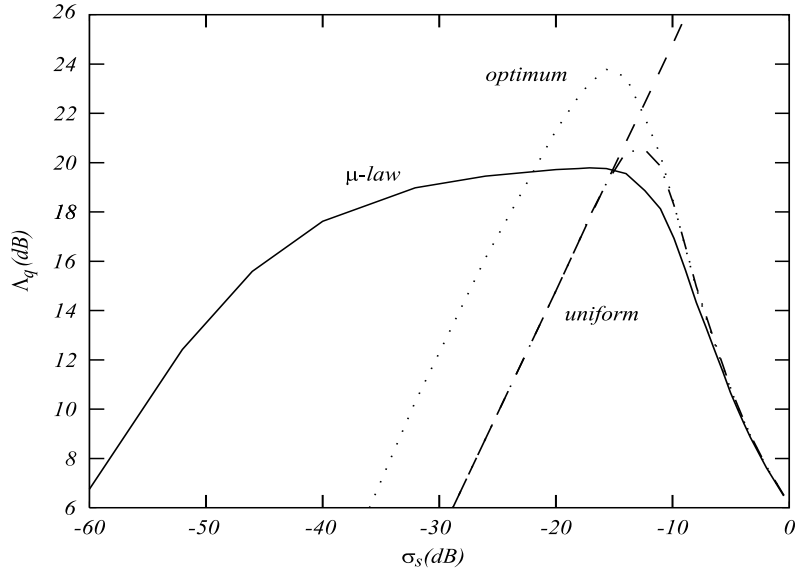


Figure 5.43. Comparison of the signal-to-quantization error ratio for uniform quantizer (dashed-dotted line), μ-law (continuous line) and optimum non-uniform quantizer (dotted line), for Laplacian input. All quantizers have 32 levels ($b = 5$) and are optimized for $\sigma_s = 1$.

The Γ quantizers have performance that is almost independent of the type of input. The performance also does not change for a wide range of the signal variance, as their characteristic is of logarithmic type (see Section 5.3.1).

A comparison between uniform and non-uniform quantizers with Laplacian input is given in Figure 5.43. All quantizers have 32 levels ($\mathfrak{b} = 5$) and are determined using: a) Table 5.5 for the uniform Laplacian type quantizer; b) Table 5.9 for the non-uniform Laplacian type quantizer; c) the μ ($\mu = 255$) compression law of Figure 5.36 with $\tau_{sat}/\sigma_s = 1$. We note that the optimum non-uniform quantizer gives best performance, even if this happens in a short range of values σ_s; for a decrease in the input statistical power, performance decreases according to the law $10 \log \mathtt{M}_s = 20 \log \sigma_s$ (dB), as we can see from (5.98). Only a logarithmic quantizer is independent of the input signal level.

5.4 Adaptive quantization

An alternative method to quantize a *non-stationary* signal consists in using an adaptive quantizer. The corresponding coding scheme, which has parameters that are adapted (over short periods) to the level of the input signal, is called an *adaptive PCM* or *APCM*.

General scheme

The overall scheme is given in Figure 5.44, where $\tilde{c}(k) \neq c(k)$ if errors are introduced by the binary channel. For a uniform quantizer, the idea is that of varying with time the quantization step size $\Delta(k)$ so that the quantizer characteristic adapts to the statistical power of the input signal.

If $\Delta(k)$ is the quantization step size at instant k, with reference to Figure 5.21 the quantizer characteristic is defined as

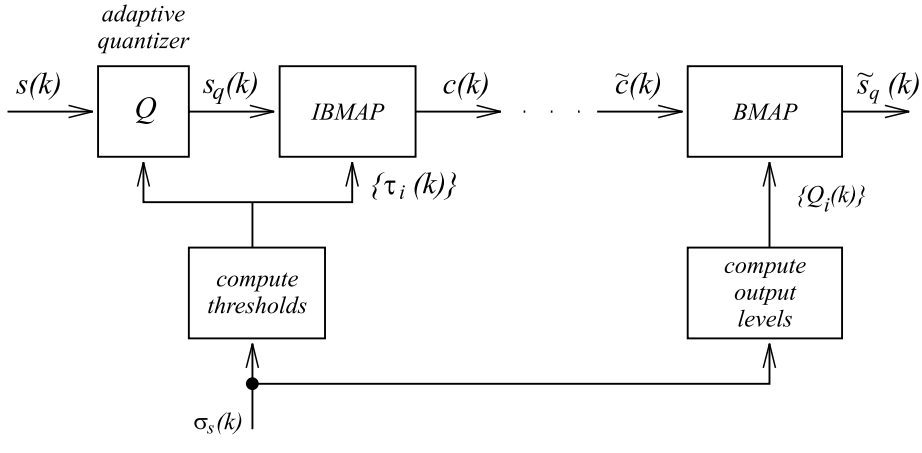


Figure 5.44. Adaptive quantization and mapping: general scheme.

$$\text{output levels:}\quad \mathcal{Q}_i(k) = \begin{cases} \left(i + \dfrac{1}{2}\right)\Delta(k) & i = -\dfrac{L}{2}, \ldots, -1 \\ \left(i - \dfrac{1}{2}\right)\Delta(k) & i = 1, \ldots, \dfrac{L}{2} \end{cases} \tag{5.104}$$

$$\text{thresholds:}\quad \tau_i(k) = i\,\Delta(k) \qquad i = -\frac{L}{2} + 1, \ldots, -1, 0, 1, \ldots, \frac{L}{2} - 1$$

If Δ_{opt} is the optimum value of Δ for a given amplitude distribution of the input signal assuming $\sigma_s = 1$ (see Table 5.5), and $\sigma_s(k)$ is the standard deviation of the signal at instant k, then we can use the following rule

$$\Delta(k) = \Delta_{opt}\sigma_s(k) \tag{5.105}$$

For a non-uniform quantizer, we need to change the levels and thresholds according to the relations:

$$\begin{aligned} \mathcal{Q}_i(k) &= \mathcal{Q}_{i,opt}\sigma_s(k) \\ \tau_i(k) &= \tau_{i,opt}\sigma_s(k) \end{aligned} \tag{5.106}$$

where $\{\mathcal{Q}_{i,opt}\}$ and $\{\tau_{i,opt}\}$ are given in Tables 5.8, 5.9, and 5.10 for various input amplitude distributions.

As illustrated in Figure 5.45, an alternative to the scheme of Figure 5.44 is the following: the quantizer is fixed and the input is scaled by an adaptive gain g, so that a signal $\{y(k)\}$ is generated with a constant statistical power, for example, $\sigma_y^2 = 1$. Therefore we let

$$g(k) = \frac{1}{\sigma_s(k)} \tag{5.107}$$

However, both methods require computing the statistical power σ_s^2 of the input signal. The adaptive quantizers are classified as:

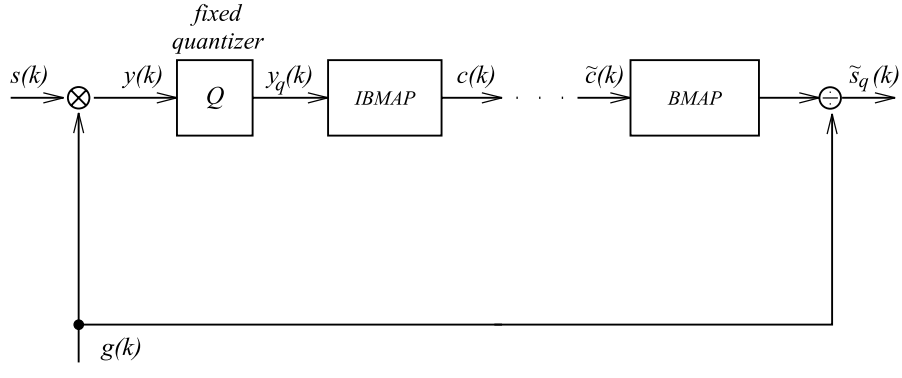


Figure 5.45. Adaptive gain, fixed quantization and mapping.

- *feedforward*, if σ_s is estimated by observing the signal $\{s(k)\}$ itself;
- *feedback*, if σ_s is estimated by observing $\{s_q(k) = \mathcal{Q}[s(k)]\}$ or $\{c(k)\}$, i.e. the signals at the output of the quantizer.

5.4.1 Feedforward adaptive quantizer

The feedforward methods for the two adaptive schemes of Figure 5.44 and Figure 5.45 are shown, respectively, in Figure 5.46 and Figure 5.47. The main difficulty in the two methods is that we need to quantize also the value of $\sigma_s(k)$ so that it can be coded and transmitted over a binary channel.

We emphasize that:

1. because of digital channel errors on both $c(k)$ and $(\sigma_s(k))_q$ (or $g_q(k)$) it may happen that $\tilde{s}_q(k) \neq s_q(k)$;
2. we need to determine the update frequency of $\sigma_s(k)$, that is what frequency is required to sample σ_s, and how many bits must be used to represent $(\sigma_s(k))_q$;
3. the system bit rate is now the sum of the bit rate of $c(k)$ and $(\sigma_s(k))_q$ (or $g_q(k)$).

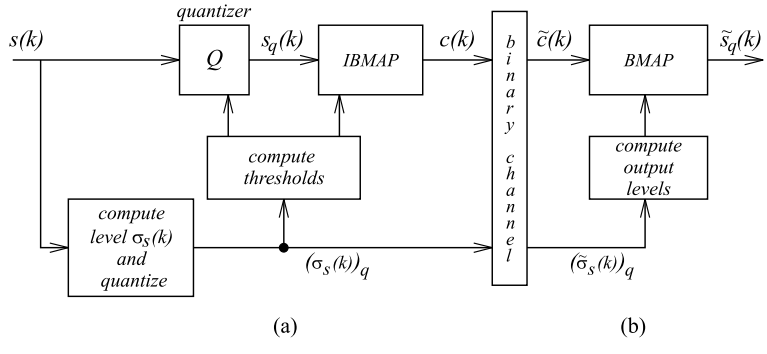


Figure 5.46. APCM scheme with feedforward adaptive quantizer: a) encoder, b) decoder.

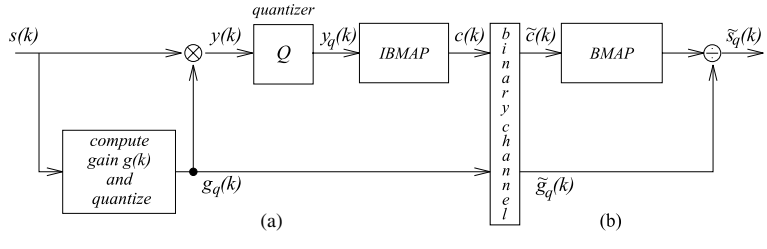


Figure 5.47. APCM scheme with feedforward adaptive gain and fixed quantizer: a) encoder, b) decoder.

The data sequence that represents $\{(\sigma_s(k))_q\}$ or $\{g_q(k)\}$ is called *side information*. Two methods to estimate $\sigma_s^2(k)$ are given in Section 1.11.1. For example, using a rectangular window of K samples, from (1.462) we have

$$\sigma_s^2(k-D) = \frac{1}{K} \sum_{n=k-(K-1)}^{k} s^2(n) \tag{5.108}$$

where D expresses a certain lead of the estimate with respect to the last available sample: typically $D = (K-1)/2$ or $D = K-1$. If $D = K-1$, K samples need to be stored in a buffer and then the average power must be computed: obviously, this introduces a latency in the coding system that is not always tolerable. Moreover, windows usually do not overlap, hence σ_s is updated every K samples.

For an exponential filter instead, from (1.468) we have

$$\sigma_s^2(k-D) = a\sigma_s^2(k-1-D) + (1-a)s^2(k) \tag{5.109}$$

Typically in this case we choose $D = 0$. To determine the update frequency of $\sigma_s^2(k)$, we recall that the 3 dB bandwidth of $\sigma_s^2(k)$ in (5.109) is equal to $B_\sigma = \frac{(1-a)}{(2\pi T_c)}$, for $a > 0.9$. Typically, however, we prefer to determine a from the equivalence (1.471) with the length of the rectangular window, that gives $a = 1 - \frac{1}{K-1}$: this means decimating, quantizing, and coding the values given by (5.109) every K instants. In Table 5.11 we give, for three values of a, the corresponding values of $K-1$ and B_σ for $1/T_c = 8$ kHz.

Performance

With the constraint that σ_s varies within a specific range, $\sigma_{min} \le \sigma_s \le \sigma_{max}$, in order to keep Λ_q relatively constant for a change of 40 dB in the input level, it must be

$$\frac{\sigma_{max}}{\sigma_{min}} \ge 100 \tag{5.110}$$

Actually σ_{min} controls the quantization error level for small input values (*idle noise*), whereas σ_{max} controls the saturation error level.

For speech signals sampled at 8 kHz, Table 5.12 shows the performance of different fixed and adaptive 8-level ($\mathfrak{b} = 3$) quantizers. The estimate of the signal power is obtained by a rectangular window with $D = K-1$; the decimation and quantization of $\sigma_s^2(k)$

Table 5.11 Time constant and bandwidth of a discrete-time exponential filter with parameter a and sampling frequency 8 kHz.

a	Time constant $K-1 = 1/(1-a)$ (samples)	Filter bandwidth $B_\sigma = (1-a)/(2\pi T_c)$ (Hz)
$1-2^{-5} = 0.9688$	32	40
$1-2^{-6} = 0.9844$	64	20
$1-2^{-7} = 0.9922$	128	10

Table 5.12 Performance comparison of fixed and adaptive quantizers for speech.

Speech $s(k)$ b=3	Λ_q (dB)		
	Non-adaptive	Adaptive $K = 128$ (16 ms)	Adaptive $K = 1024$ (128 ms)
non-uniform $\mathcal{Q}$			
μ law ($\mu = 100$, $\tau_{sat}/\sigma_s = 8$)	9.5	–	–
Gaussian ($\Lambda_{q,opt} = 14.6$ dB)	7.3	15	12.1
Laplacian ($\Lambda_{q,opt} = 12.6$ dB)	9.9	13.3	12.8
uniform $\mathcal{Q}$			
Gaussian ($\Lambda_{q,opt} = 14.3$ dB)	6.7	14.7	11.3
Laplacian ($\Lambda_{q,opt} = 11.4$ dB)	7.4	13.4	11.5

are not considered. Although $\mathtt{b} = 3$ is a small value to draw conclusions, we note that using an adaptive Gaussian quantizer with $K = 128$ we get 8 dB improvement over a non-adaptive quantizer. If $K \ll 128$ the side information becomes excessive, conversely there is a performance loss of 3 dB for $K = 1024$.

5.4.2 Feedback adaptive quantizers

As illustrated in Figure 5.48, the feedback method estimates σ_s from the knowledge of $\{s_q(k) = \mathcal{Q}[s(k)]\}$ or $\{c(k)\}$.

We make the following observations:

- there is no need to transmit $\sigma_s(k)$; therefore feedback methods do not require the transmission of side information;
- a transmission error on $c(k)$ affects not only the identification of the quantized level, but also the scaling factor $\sigma_s(k)$.

Concerning the estimate of σ_s, a possible method consists in applying (5.108) or (5.109), where $\{s(n)\}$ is substituted by $\{s_q(n)\}$. However, this signal is available only for $n \leq k-1$:

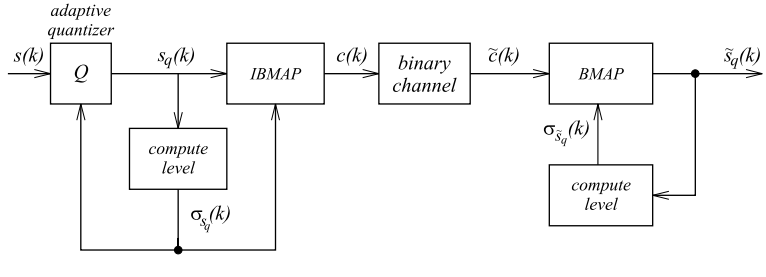


Figure 5.48. APCM scheme with feedforward adaptive quantizer.

this implies that the estimate (5.108) becomes $\sigma^2_{s_q}(k) = 1/K \sum_{n=(k-1)-(K-1)}^{k-1} s^2_q(n)$, with a lag of one sample. Likewise, the recursive estimate (5.109) becomes $\sigma^2_{s_q}(k) = a\sigma^2_{s_q}(k-1) + (1-a)s^2_q(k-1)$. Because of the lag in estimating the level of the input signal and the computational complexity of the method itself, we present now an alternative method to estimate σ_s adaptively.

Estimate of $\sigma_s(k)$

For an input with $\sigma_s = 1$ we compute the discrete amplitude distribution of the code words for a quantizer with 2^{b} levels and $|c(k)| \in \{1, 2, \ldots, L/2\}$. As illustrated in Figure 5.49 for $\mathrm{b} = 3$, let

$$\begin{cases} P[|c(k)| = 1] &= 2\displaystyle\int_{\tau_0=0}^{\tau_{opt,1}} p_s(a)\,da &= p_{c_1} \\ & \vdots & \vdots \\ P[|c(k)| = 4] &= 2\displaystyle\int_{\tau_{opt,3}}^{+\infty} p_s(a)\,da &= p_{c_4} \end{cases} \tag{5.111}$$

If σ_s changes suddenly, the distribution of $|c(k)|$ will be very different with respect to (5.111). For example, if $\sigma_s < 1$ it will be $P[|c(k)| = 1] \gg p_{c_1}$, while $P[|c(k)| = 4] \ll p_{c_4}$.

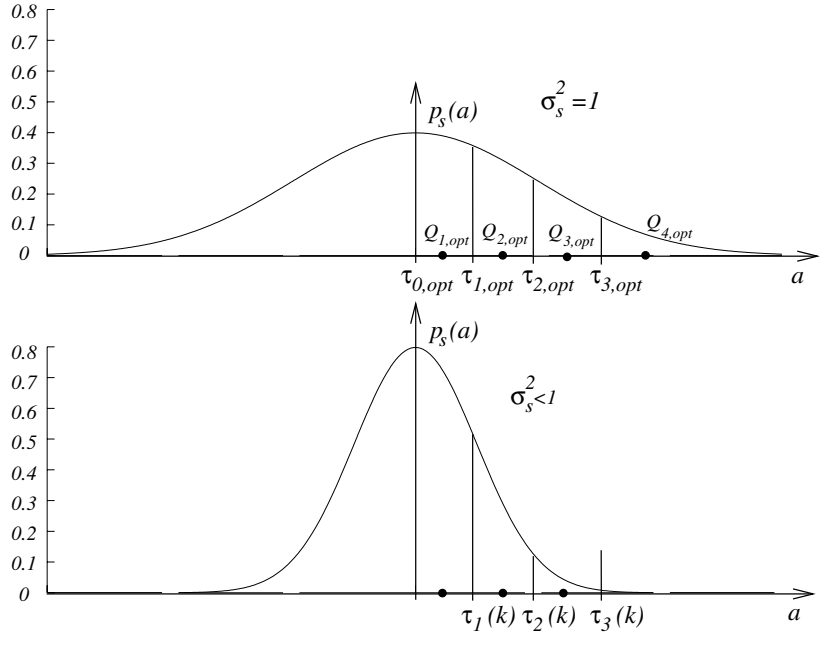


Figure 5.49. Output levels and optimum decision thresholds for Gaussian $s(k)$ with unit variance ($\mathrm{b} = 3$).

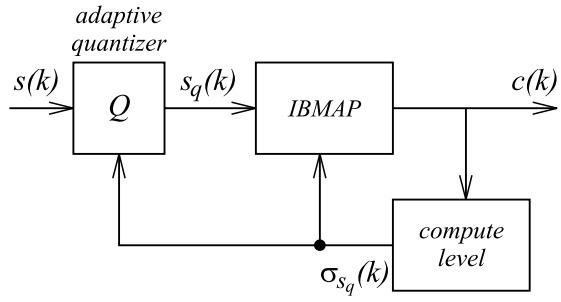


Figure 5.50. Adaptive quantizer where σ_s is estimated using the code words.

The objective is therefore that of changing $\sigma_{s_q}(k)$ so that the optimal distribution is obtained for $|c(k)|$. The algorithm proposed by Jayant [4], illustrated in Figure 5.50, is given by

$$\sigma_{s_q}(k) = \mathrm{p}[|c(k-1)|]\sigma_{s_q}(k-1) \tag{5.112}$$

in which $\{\mathrm{p}[i]\}$, $i = 1, \ldots, L/2$, are suitable parameters. For example, if it is $|c(k-1)| = 1$, then σ_{s_q} must decrease to reduce the quantizer range, thus $\mathrm{p}[1] < 1$; if instead it is $|c(k-1)| = L/2$, then σ_{s_q} must increase to extend the quantizer range, thus $\mathrm{p}[L/2] > 1$.

In practice, what we do is vary σ_{s_q} by small steps imposing bounds to the variations, that is:

$$\sigma_{min} \leq \sigma_{s_q}(k) \leq \sigma_{max} \tag{5.113}$$

The problem consists now in choosing the parameters $\{\mathrm{p}[i]\}$, $i = 1, \ldots, L/2$. Intuitively it should be

$$(\mathrm{p}[1])^{p_{c_1}} (\mathrm{p}[2])^{p_{c_2}} \ldots (\mathrm{p}[L/2])^{p_{c_{L/2}}} = 1 \tag{5.114}$$

In fact, from (5.112) it follows that

$$\ln \sigma_{s_q}(k) = \ln \mathrm{p}[|c(k-1)|] + \ln \sigma_{s_q}(k-1) \tag{5.115}$$

from which

$$E[\ln \sigma_{s_q}](k) = E[\ln \mathrm{p}[|c(k-1)|]] + E[\ln \sigma_{s_q}(k-1)] \tag{5.116}$$

In steady state we expect that $E[\ln \sigma_{s_q}(k)] = E[\ln \sigma_{s_q}(k-1)]$, therefore it must be

$$E[\ln \mathrm{p}[|c(k-1)|]] = \sum_{i=1}^{L/2} p_{c_i} \ln \mathrm{p}[i] = 0 \tag{5.117}$$

as in (5.114).

Based on numerous tests on speech signals, Jayant also gave the values of the parameters $\{\mathrm{p}[i]\}$, $i = 1, \ldots, L/2$. Let

$$\mathrm{q}(i) = \frac{2i-1}{L-1} \in \left\{ \frac{1}{L-1}, \frac{3}{L-1}, \ldots, 1 \right\} \tag{5.118}$$

In Figure 5.51 the values of $\{\mathrm{p}[i]\}$ are given in correspondence of $\{\mathrm{q}(i)\}$, $i = 1, \ldots, L/2$ [4]. For example, for $L = 8$ the values of $\{\mathrm{p}[i]\}$, $i = 1, \ldots, 4$, are in correspondence of the values of $\{\mathrm{q}(i)\} = \{1/7, 3/7, 5/7, 1\}$. Therefore $\mathrm{p}[1]$ is in the range from 0.8 to 0.9, and $\mathrm{p}[4]$ in the range from 1.8 to 2.9; we note that there is a large interval of possible values for $\mathrm{p}[i]$, especially if the index i is large.

Summarizing, at instant k, $\sigma_{s_q}(k)$ is known, and by (5.106) the decision thresholds $\{\tau_i(k)\}$ are also known. From the input sample $s(k)$, $c(k)$ is produced by the quantizer characteristic (see Figure 5.52). Then $\sigma_{s_q}(k+1)$ is computed by (5.112) and the thresholds are updated: the quantizer is now ready for the next sample $s(k+1)$.

At the receiver, the possible output values are also known from the knowledge of $\sigma_{s_q}(k)$ (see (5.106)). At the reception of $c(k)$ the index i in (5.106) is determined and, consequently, $s_q(k)$; in turn the receiver updates the value of $\sigma_{s_q}(k+1)$ by (5.112). Experimental measurements on speech signals indicate that this *feedback* adaptive scheme offers performance similar to that of a feedforward scheme. An advantage of the algorithm of Jayant is that it is sequential, thus it can adapt very quickly to changes in the mean signal level; on the other hand, it is strongly affected by the errors introduced by the binary channel.

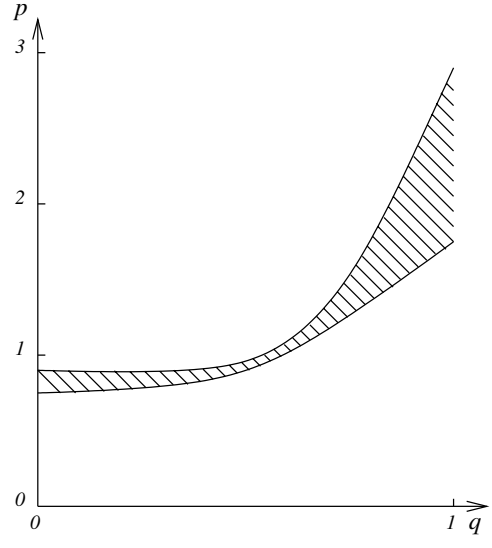


Figure 5.51. Interval of the multiplier parameters in the quantization of the speech signal as a function of the parameters $\{\mathrm{q}(i)\}$ [4]. [*From Jayant and Noll (1984).*]

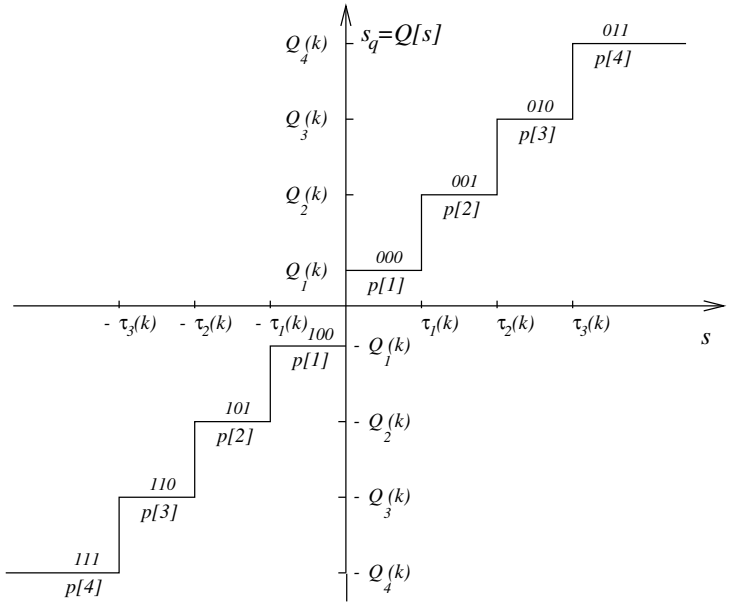


Figure 5.52. Input--output characteristic of a 3-bit adaptive quantizer. For each output level the PCM code word and the corresponding value of p are given.

5.5 Differential coding (DPCM)

The basic idea consists in quantizing the *prediction error* signal rather than the signal itself.[7] With reference to Figure 5.53, for a linear predictor with N coefficients, let $\hat{s}(k)$ be the *prediction signal*:[8]

$$\hat{s}(k) = \sum_{i=1}^{N} c_i s(k-i) \tag{5.119}$$

From (2.81) the *prediction error* is defined as

$$f(k) = s(k) - \hat{s}(k) \tag{5.120}$$

Considering the z-transform, let

$$C(z) = \sum_{i=1}^{N} c_i z^{-i} \tag{5.121}$$

[7] In the following sections, as well as in some schemes of the previous section on adaptive quantization, when processing of the input samples $\{s(k)\}$ is involved, it is desirable to perform the various operations in the digital domain on a linear PCM binary representation of the various samples, obtained by an ADC. Obviously, the finite number of bits of this preliminary quantization should not affect further processing. To avoid introducing a new signal, the preliminary conversion by an ADC is omitted in all our schemes.

[8] The considerations presented in this section are valid for any predictor, even non-linear predictors.

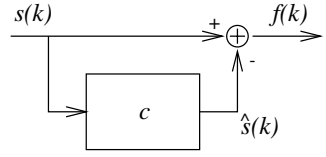


Figure 5.53. Computation of the prediction error signal $f(k)$.

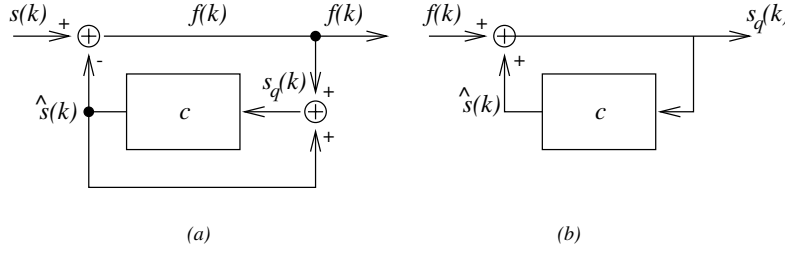


Figure 5.54. (a) Prediction error filter; (b) Inverse prediction error filter.

then

$$\hat{S}(z) = C(z)S(z) \tag{5.122}$$

and

$$F(z) = S(z)[1 - C(z)] \tag{5.123}$$

Recalling (2.81), $[1 - C(z)]$ is the *prediction error filter*. In the case $C(z) = z^{-1}$, that is for a predictor with a single coefficient equal to one, $f(k)$ coincides with the difference between two consecutive input samples.

It is interesting to re-arrange the scheme of Figure 5.53 in the equivalent scheme of Figure 5.54, where $s_q(k)$ is called the *reconstruction signal* and is given by

$$s_q(k) = \hat{s}(k) + f(k) \tag{5.124}$$

Figure 5.54a illustrates how the prediction error is obtained starting from input $s(k)$ and prediction $\hat{s}(k)$, Figure 5.54b shows how to obtain the reconstruction signal from $f(k)$ and $\hat{s}(k)$, according to (5.124). From (5.120) and (5.124), it is easy to prove that in the scheme of Figure 5.54 we have

$$s_q(k) = s(k) \tag{5.125}$$

that is the reconstructed signal coincides with the input signal.

We will now quantize the signal $\{f(k)\}$.

5.5.1 Configuration with feedback quantizer

With reference to the scheme of Figure 5.55, the following relations hold.

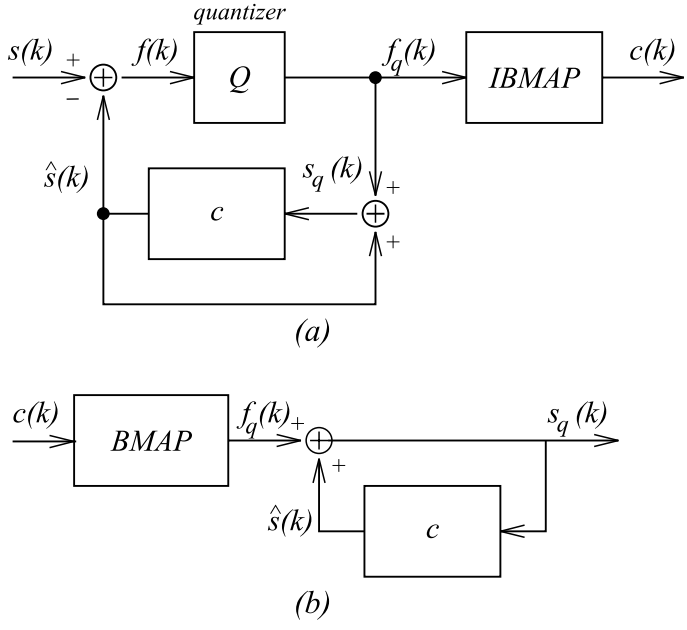


Figure 5.55. DPCM scheme with quantizer inserted in the feedback loop: (a) encoder, (b) decoder.

Encoder:

$$f(k) = s(k) - \hat{s}(k) \tag{5.126}$$

$$f_q(k) = \mathcal{Q}[f(k)] \tag{5.127}$$

$$s_q(k) = \hat{s}(k) + f_q(k) \tag{5.128}$$

$$\hat{s}(k+1) = \sum_{i=1}^{N} c_i s_q(k+1-i) \tag{5.129}$$

Decoder:

$$s_q(k) = \hat{s}(k) + f_q(k) \tag{5.130}$$

$$\hat{s}(k+1) = \sum_{i=1}^{N} c_i s_q(k+1-i) \tag{5.131}$$

In other words, the quantized prediction error is transmitted over the binary channel. Let

$$e_{q,f}(k) = f_q(k) - f(k) \tag{5.132}$$

be the quantization error and $\Lambda_{q,f}$ the signal-to-quantization error ratio

$$\Lambda_{q,f} = \frac{E[f^2(k)]}{E[e_{q,f}^2(k)]} \tag{5.133}$$

Recalling (5.98) we know that for an optimum quantizer, with the normalization by the standard deviation of $\{f(k)\}$, $\Lambda_{q,f}$ is only a function of the number of bits and of the probability density function of $\{f(k)\}$.

From (5.128) and (5.132), using (5.126), we have

$$s_q(k) = \hat{s}(k) + f(k) + e_{q,f}(k) = s(k) + e_{q,f}(k) \tag{5.134}$$

To summarize, the reconstruction signal is different from the input signal, $s_q(k) \neq s(k)$, and the reconstruction error (or noise) depends on the quantization of $f(k)$, not of $s(k)$. Consequently, if $\mathtt{M}_f < \mathtt{M}_s$ then also $\mathtt{M}_{e_{q,f}} < \mathtt{M}_{e_q}$ and the DPCM scheme presents an advantage over PCM. Observing (5.134) the signal-to-noise ratio is given by

$$\Lambda_q = \frac{\mathtt{M}_s}{\mathtt{M}_{e_{q,f}}} = \frac{\mathtt{M}_s}{\mathtt{M}_f}\frac{\mathtt{M}_f}{\mathtt{M}_{e_{q,f}}} \tag{5.135}$$

Given

$$G_p = \frac{\mathtt{M}_s}{\mathtt{M}_f} \tag{5.136}$$

called *prediction gain*, it follows that

$$\Lambda_q = G_p \Lambda_{q,f} \tag{5.137}$$

where observing (5.133) $\Lambda_{q,f}$ depends on the number of quantizer levels, which in turn determine the transmission bit rate, whereas G_p depends on the predictor complexity and on the correlation sequence of the input $\{s(k)\}$. We observe that the input to the filter that yields $\hat{s}(k)$ in (5.129) is $\{s_q(k)\}$ and not $\{s(k)\}$; this will cause a deterioration of G_p with respect to the ideal case $\{s_q(k) = s(k)\}$. This decrease will be more prominent the larger $\{e_{q,f}\}$ will be with respect to $\{s(k)\}$.

If we ignore the dependence of G_p on $\{e_{q,f}(k)\}$, (5.137) shows that to obtain a given Λ_q we can use a quantizer with a few levels, provided the input $\{s(k)\}$ is highly predictable. Therefore G_p can be sufficiently high also for a predictor with a reduced complexity.

For the quantizer, assuming the distribution of $\{f(k)\}$ is known, $\Lambda_{q,f}$ is maximized by selecting the thresholds and the output values according to the techniques given in Section 5.3. In particular the statistical power of $\{f(k)\}$, useful in scaling the quantizer characteristic, can be derived from (5.136), assuming known $\mathtt{M}_s$ and G_p,

$$\mathtt{M}_f = \frac{\mathtt{M}_s}{G_p} \tag{5.138}$$

Regarding the predictor, once the number of coefficients N is fixed, we need to determine the coefficients $\{c_i\}$, $i = 1, \ldots, N$, that minimize $\mathtt{M}_f$. For example in the case $N = 1$, recalling (2.91), the optimum value of c_1 is given by $\rho(1)$, the correlation coefficient of the input signal at lag 1. Then we have

$$G_p = \frac{1}{1 - \rho^2(1)} \tag{5.139}$$

ignoring the effect of the quantizer, that is for $\{s_q(k) = s(k)\}$.

Table 5.13 Prediction gain with $N = 1$ for three values of $\rho(1)$.

$c_1 = \rho(1)$	$G_p = 1/(1 - \rho^2(1))$ (dB)
0.85	5.6
0.90	7.2
0.95	10.1

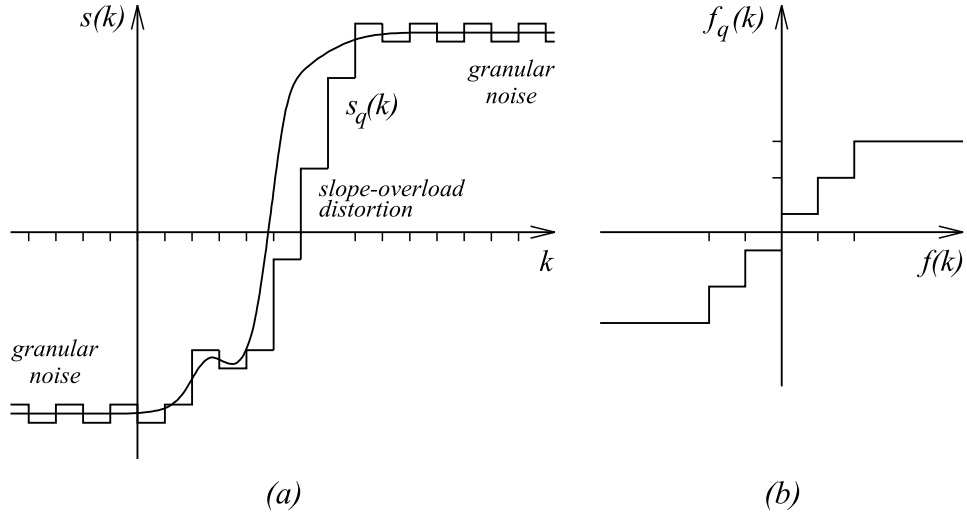


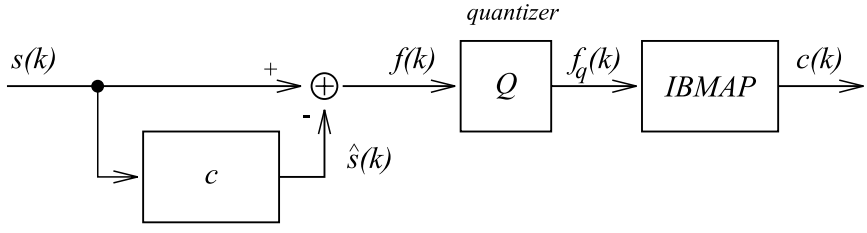
Figure 5.56. (a) Reconstruction signal for a DPCM, with (b) a 6 level quantizer.

We give in Table 5.13 the values of G_p for three values of $\rho(1)$. We note that, for an input having $\rho(1) = 0.85$, a simple predictor with one coefficient yields a prediction gain equivalent to one bit of the quantizer: consequently, given the total Λ_q, the DPCM scheme allows us to use a transmission bit rate lower than that of PCM. Evidently, for an input with $\rho(1) = 0$ there is no advantage in using the DPCM scheme.

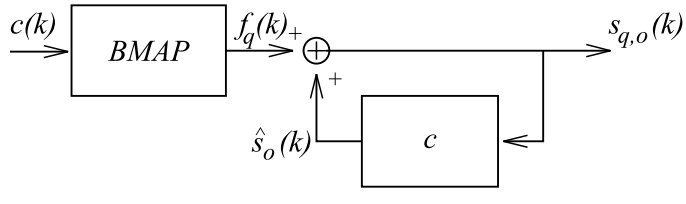
For a simple predictor with $N = 1$ and $c_1 = 1$, hence $\hat{s}(k) = s_q(k-1)$, Figure 5.56a illustrates the behavior of the reconstruction signal after DPCM with the six-level quantizer shown in Figure 5.56b. We note that the minimum level of the quantizer still determines the statistical power of the granular noise in $\{s_q(k)\}$; the maximum level of the quantizer is instead related to the *slope overload distortion* in the sense that if $\mathcal{Q}_{L/2}$ is not sufficiently large, as shown in Figure 5.56a, the output signal cannot follow the rapid changes of the input signal. In the specific case, being $\mathcal{Q}_{L/2}/T_c < \max|s(k) - s(k-1)|$, $\{s_q(k)\}$ presents a slope different from that of $\{s(k)\}$ in the instants of maximum variation.

5.5.2 Alternative configuration

If we use few quantization levels, the predictor of the scheme of Figure 5.55, having as input $\{s_q(k)\}$ instead of $\{s(k)\}$, can give poor performance because of the large quantization



(a) Encoder



(b) Decoder

Figure 5.57. DPCM scheme with quantizer inserted *after* the feedback loop: a) encoder, b) decoder.

noise present in $\{s_q(k)\}$. An alternative consists in using the scheme of Figure 5.57, where the following relations hold.

Encoder:

$$f(k) = s(k) - \hat{s}(k) \tag{5.140}$$

$$f_q(k) = \mathcal{Q}[f(k)] \tag{5.141}$$

$$s_q(k) = s(k) \tag{5.142}$$

$$\hat{s}(k+1) = \sum_{i=1}^{N} c_i s(k+1-i) \tag{5.143}$$

Decoder:

$$s_{q,o}(k) = f_q(k) + \hat{s}_o(k) \tag{5.144}$$

$$\hat{s}_o(k+1) = \sum_{i=1}^{N} c_i s_{q,o}(k+1-i) \tag{5.145}$$

At the encoder, $\hat{s}(k)$ is obtained from the input signal without errors. However, the prediction signal reconstructed at the decoder is $\hat{s}_o(k) \neq \hat{s}(k)$. In fact, from (5.144) and

(5.145), even if by chance $\hat{s}_o(i) = \hat{s}(i)$, for $i \leq k-1$, as $f_q(k-1) \neq f(k-1)$ then $s_{q,o}(k-1) \neq s(k-1)$, and consequently $\hat{s}_o(k) \neq \hat{s}(k)$.

A difficulty of the scheme is that, depending on the function $C(z)$, the difference between the prediction signals, $\hat{s}_o(k) - \hat{s}(k)$, may be non-negligible. As a result the output

$$\begin{aligned} s_{q,o}(k) &= \hat{s}_o(k) + f(k) + e_{q,f}(k) \\ &= \hat{s}_o(k) + s(k) - \hat{s}(k) + e_{q,f}(k) \\ &= s(k) + [\hat{s}_o(k) - \hat{s}(k)] + e_{q,f}(k) \end{aligned} \tag{5.146}$$

can assume values that are quite different from $s(k)$.

Observation 5.4
Note that the same problem mentioned above may occur also in the scheme of Figure 5.55 because of errors introduced by the binary channel, though to a lesser extent as compared to the scheme of Figure 5.57, as the signal $\{f_q(k)\}$ at the encoder is affected by a smaller disturbance. For both configurations, however, the inverse prediction error filter must suppress the propagation of such errors in a short time interval. This is difficult to achieve if the transfer function $1/[1 - C(z)]$ has poles near the unit circle, and consequently the impulse response is very long.

5.5.3 Expression of the optimum coefficients

For linear predictors, the prediction signal $\hat{s}(k)$ is given by

$$\hat{s}(k) = \sum_{i=1}^{N} c_i s_q(k-i) \tag{5.147}$$

where $s_q(k)$ is the reconstruction signal, which in the case of a feedback quantizer system is given by $s_q(k) = s(k) + e_{q,f}(k)$. For the design of the predictor, we choose the coefficients $\{c_i\}$ that minimize the statistical power of the prediction error,

$$\mathtt{M}_f = E[(s(k) - \hat{s}(k))^2] \tag{5.148}$$

We introduce the following vectors and matrices.

Vector of prediction coefficients

$$\mathbf{c} = [c_1, \ldots, c_N]^T \tag{5.149}$$

Vector of correlation coefficients of $\{s(k)\}$

$$\boldsymbol{\rho} = [\rho(1), \ldots, \rho(N)]^T \tag{5.150}$$

where $\rho(i)$ is defined in (1.540).

Correlation matrix of s_q, normalized by $\mathtt{M}_s$[9]

$$\mathbf{\Gamma} = \begin{bmatrix} \left(1+\dfrac{1}{\Lambda_q}\right) & \rho(1) & \dots & \rho(N-1) \\ \rho(1) & \left(1+\dfrac{1}{\Lambda_q}\right) & \dots & \rho(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ \rho(N-1) & \dots & \rho(1) & \left(1+\dfrac{1}{\Lambda_q}\right) \end{bmatrix} \tag{5.153}$$

Recalling the analysis of Section 2.2, the optimum prediction coefficients are given by the matrix equation (2.78)

$$\mathbf{\Gamma}\mathbf{c}_{opt} = \boldsymbol{\rho} \tag{5.154}$$

The corresponding minimum value of $\mathtt{M}_f$ is obtained from (2.79),

$$\mathtt{M}_f = \mathtt{M}_s(1 - \mathbf{c}_{opt}^T \boldsymbol{\rho}) \tag{5.155}$$

The difficulty of this formulation is that to determine the solution we need to know the value of Λ_q (see (5.153)). We may consider the solution with the *quantizer omitted*, hence $\Lambda_q = \infty$, and $\mathbf{\Gamma}$ depends only on the second order statistic of $\{s(k)\}$. In this case some efficient algorithms to determine $\mathbf{c}$ and $\mathtt{M}_f$ in (5.154) and (5.155) are given in Sections 2.2.1 and 2.2.2.

Effects due to the presence of the quantizer

Observing (5.155), the prediction gain is given by

$$G_p = \frac{\mathtt{M}_s}{\mathtt{M}_f} = \frac{1}{1 - \mathbf{c}_{opt}^T \boldsymbol{\rho}} \tag{5.156}$$

In general it is very difficult to analyze the effects of $\{e_{q,f}(k)\}$ on G_p, except in the case $N = 1$, for which (5.154) becomes

$$c_{opt,1}\left(1+\frac{1}{\Lambda_q}\right) = \rho(1) \tag{5.157}$$

Then

$$c_{opt,1} = \frac{\rho(1)}{1 + (1/\Lambda_q)} \tag{5.158}$$

---

[8] Assuming for $\{s(k)\}$ and $\{e_{q,f}(k)\}$ the correlations are expressed by (5.27) and (5.28), we get

$$\mathtt{r}_{s_q}(n) = \mathtt{r}_s(n) + \mathtt{M}_{e_{q,f}}\,\delta_n \tag{5.151}$$

Dividing by $\mathtt{M}_s$ we obtain

$$\frac{\mathtt{r}_{s_q}(n)}{\mathtt{M}_s} = \rho(n) + \Lambda_q^{-1}\,\delta_n \tag{5.152}$$

and

$$G_p = \frac{1}{1 - c_{opt,1}\rho(1)} = \frac{1}{1 - \dfrac{\rho^2(1)}{1 + 1/\Lambda_q}} \tag{5.159}$$

The above relations show that if Λ_q is small, that is, if the system is very noisy, then $c_{opt,1}$ is small and G_p tends to 1. Only for $\Lambda_q = \infty$ it is $c_{opt,1} = \rho(1)$.

It may occasionally happen that a suboptimum value is assigned to c_1: we will try to evaluate the corresponding value of G_p. For $N = 1$ and any c_1 it is

$$\begin{aligned} \mathtt{M}_f &= E[(s(k) - c_1(s(k-1) + e_{q,f}(k-1)))^2] \\ &\simeq \mathtt{M}_s(1 - 2c_1\rho(1) + c_1^2) + c_1^2\,\mathtt{M}_{e_{q,f}} \end{aligned} \tag{5.160}$$

As from (5.135) it follows $\mathtt{M}_{e_{q,f}} = \mathtt{M}_s/\Lambda_q$, where from (5.137) $\Lambda_q = G_p\Lambda_{q,f}$, observing (5.160) we obtain

$$G_p = \frac{1 - (c_1^2/\Lambda_{q,f})}{1 - 2c_1\rho(1) + c_1^2} \tag{5.161}$$

hence $\Lambda_{q,f}$ depends only on the number of quantizer levels.

Note that (5.161) allows the computation of the optimum value of c_1 for a predictor with $N = 1$ in the presence of the quantizer: however, the expression is complicate and will not be given here. Rather we will derive G_p for two values of c_1.

1. For $c_1 = \rho(1)$ we have

$$G_p = \frac{1}{1 - \rho^2(1)}\left(1 - \frac{\rho^2(1)}{\Lambda_{q,f}}\right) \tag{5.162}$$

   where the factor $(1 - \rho^2(1)/\Lambda_{q,f})$ is due only to the presence of the quantizer.

2. For $c_1 = 1$ we have

$$G_p = \frac{1}{2(1 - \rho(1))}\left(1 - \frac{1}{\Lambda_{q,f}}\right) \tag{5.163}$$

We note that the choice $c_1 = 1$ leads to a simple implementation of the predictor: however, this choice results in $G_p > 1$ only if $\rho(1) > 1/2$.

Various experiments with speech have demonstrated that for *very long observations*, of the order of one second, the prediction gain for a fixed predictor is between 5 and 7 dB, and saturates for $N \geq 2$; in fact, speech is a non-stationary signal and adaptive predictors should be used.

5.5.4 Adaptive predictors

In *adaptive differential* PCM (ADPCM), the predictor is time-varying. Therefore we have

$$\hat{s}(k) = \sum_{i=1}^{N} c_i(k)s_q(k-i) \tag{5.164}$$

The vector $\mathbf{c} = [c_1, \ldots, c_N]^T$ is chosen to minimize $\mathtt{M}_f$ over short intervals within which the signal $\{s(k)\}$ is quasi-stationary. Speech signals have slowly-varying spectral characteristics and can be assumed as stationary over intervals of the order of 5–25 ms.

Also for ADPCM two strategies emerge.

Adaptive feedforward predictors

The general scheme is illustrated in Figure 5.58. We consider an observation window for the signal $\{s(k)\}$ of K samples. Based on these samples the input autocorrelation function is estimated up to lag N using (1.478); then we solve the system of equations (5.154) to obtain the coefficients $\mathbf{c}$ and the statistical power of the prediction error.

These quantities, after being appropriately quantized for finite precision representation, give the parameters of the predictor $\mathbf{c}_q$ and of the quantizer $(\sigma_f)_q$: the system is now ready to encode the samples of the observation window in sequence. The digital representation of $\{f_q(k)\}$, together with the quantized parameters of the system, must be sent to the receiver to reconstruct the signal.

For the next K samples of $\{s(k)\}$ the procedure is repeated.

In general, for speech we choose $KT_c \simeq$ 10–20 ms, and $N \simeq 10$. Not considering the computation time, this system introduces a minimum delay from $\{s(k)\}$ to the decoder output $\{\tilde{s}_q(k)\}$ equal to K samples.

The performance improvement obtained by using an adaptive scheme is illustrated in Figure 5.59. In particular, for speech signals sampled at 8 kHz, the power measured on windows of 128 samples is shown in Figure 5.59a: we note that the speech level exhibits a dynamic range of 30–40 dB and rapidly changes value. The prediction gain in the absence of the quantizer is shown for a fixed predictor with $N = 3$ and an adaptive predictor with $N = 10$ in Figure 5.59b and in Figure 5.59c, respectively. The fixed predictor is determined by considering the statistic of the whole signal, thus within certain windows the prediction gain is even less than 1. The adaptive predictor is estimated at every window by the feedforward method and yields $G_p > 1$, even for unvoiced spurts that present small correlation. We note that, for some voiced spurts, G_p can reach the value of 20–30 dB.

Sequential adaptive feedback predictors

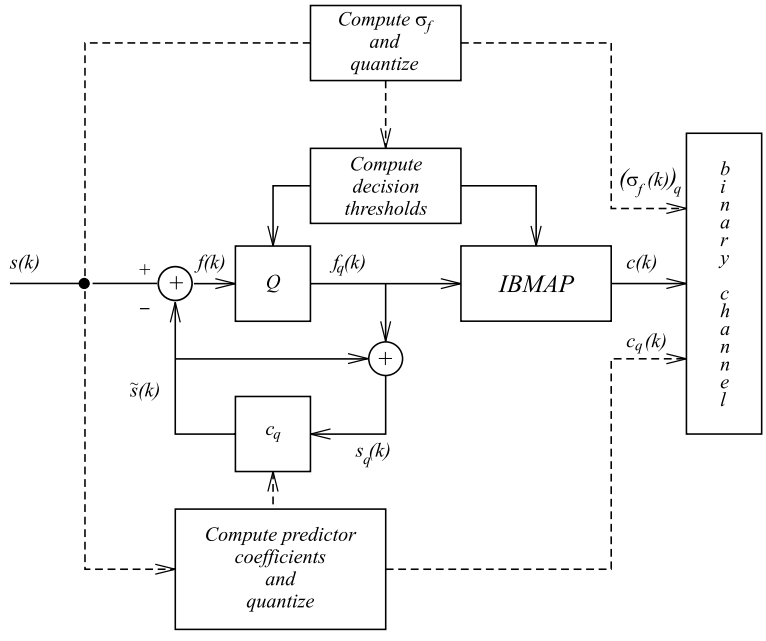
Also for adaptive feedback predictors we could observe $\{s_q(i)\}$ for a window of K samples and apply the same procedure as the feedforward method; however, the observation is now available only for instants $i < k$, and consequently this method is not suitable to track rapid changes of the input statistic. An alternative could be that of estimating *at every instant* k the correlation of $\{s_q(i)\}$, $i < k$, and calculate $\mathbf{c}(k)$; however, this method requires too many computations.

Another simple alternative, of the sequential adaptive type, is illustrated in Figure 5.60, where the predictor is adapted by the LMS algorithm (see Section 3.1.2). Defining

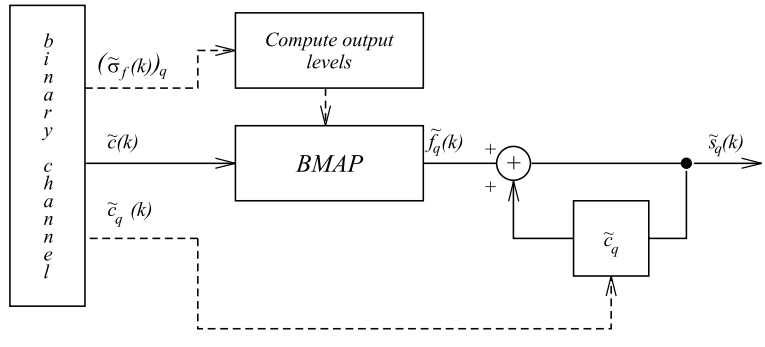
$$\mathbf{s}_q(k) = [s_q(k-1), \ldots, s_q(k-N)] \tag{5.165}$$

coefficient adaptation is given by

$$\mathbf{c}(k+1) = \mu_1 \mathbf{c}(k) + \mu\ f_q(k)\mathbf{s}_q(k) \qquad 0 < \mu < \frac{2}{N\mathtt{r}^2_{s_q}(0)} \tag{5.166}$$



(a)



(b)

Figure 5.58. ADPCM scheme with feedforward adaptation of both predictor and quantizer: (a) encoder, (b) decoder.

where $\mu_1 \leq 1$ controls the stability of the decoder if, because of binary channel errors, it occasionally happens $\tilde{f}_q(k) \neq f_q(k)$.

Table 5.14 gives the algorithm, while Figure 5.61 illustrates the implementation; in decoding the same equations are used with $\tilde{f}_q(k)$ in place of $f_q(k)$ and therefore $\tilde{s}_q(k)$ in place of $s_q(k)$.

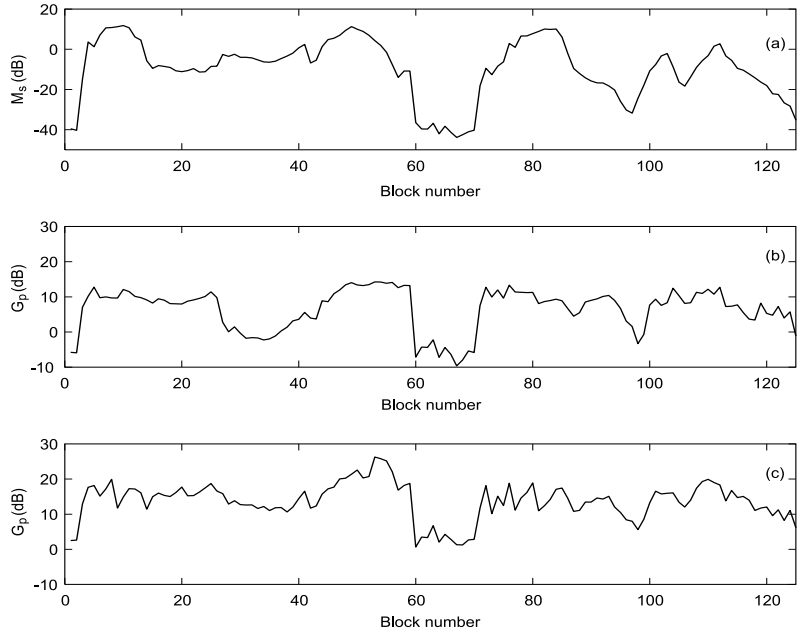


Figure 5.59. (a) Speech level measured on windows of 128 samples, and corresponding prediction gain G_P for: (b) a fixed predictor ($N = 3$), (c) an adaptive predictor ($N = 10$). For these measurements the quantizer was removed.

Table 5.14 Adaptation equations of the LMS adaptive predictor.

Initialization	
	$\mathbf{c}(0) = \mathbf{0}$
	$\mathbf{s}_q(0) = \mathbf{0}$ (or $\mathbf{s}_q(0) = \mathbf{s}(0)$)
	$\hat{s}(0) = 0$
For $k = 0, 1, \ldots$	
	$f(k) = s(k) - \hat{s}(k)$
	$f_q(k) = \mathcal{Q}[f(k)]$
	$\mathbf{c}(k+1) = \mu_1 \mathbf{c}(k) + \mu\ f_q(k)\mathbf{s}_q(k)$
	$s_q(k) = \hat{s}(k) + f_q(k)$
	$\hat{s}(k+1) = \mathbf{c}^T(k+1)\mathbf{s}_q(k+1)$

Observation 5.5
For a stationary input, as for example a modem signal, the LMS adaptive prediction can be used to easily determine the predictor coefficients; once the convergence is reached, however, it is better to switch off the adaptation. This observation does not apply to speech, which presents characteristics that may change very rapidly with time.

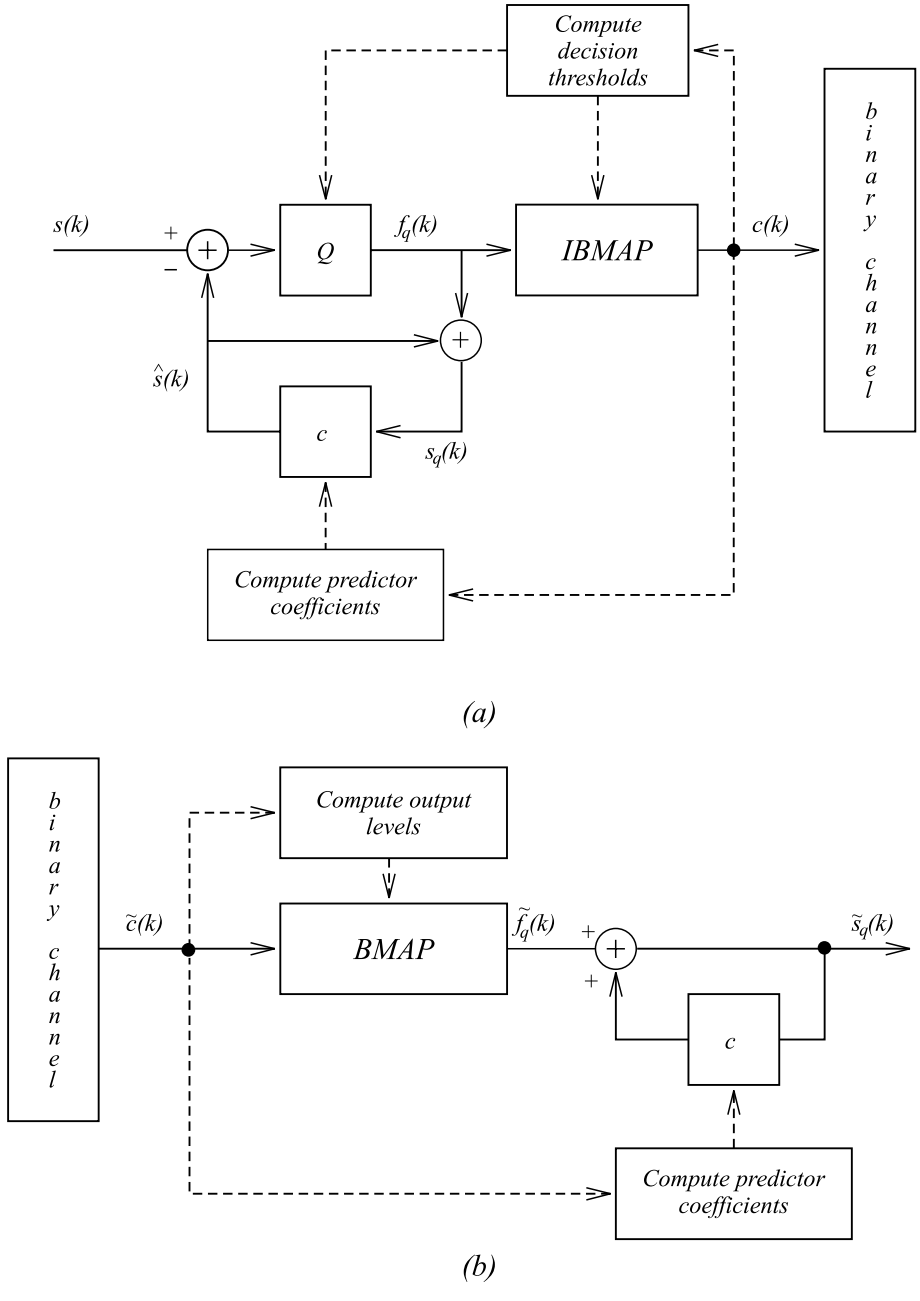


Figure 5.60. ADPCM scheme with feedback adaptation for both predictor and quantizer: (a) encoder, (b) decoder.

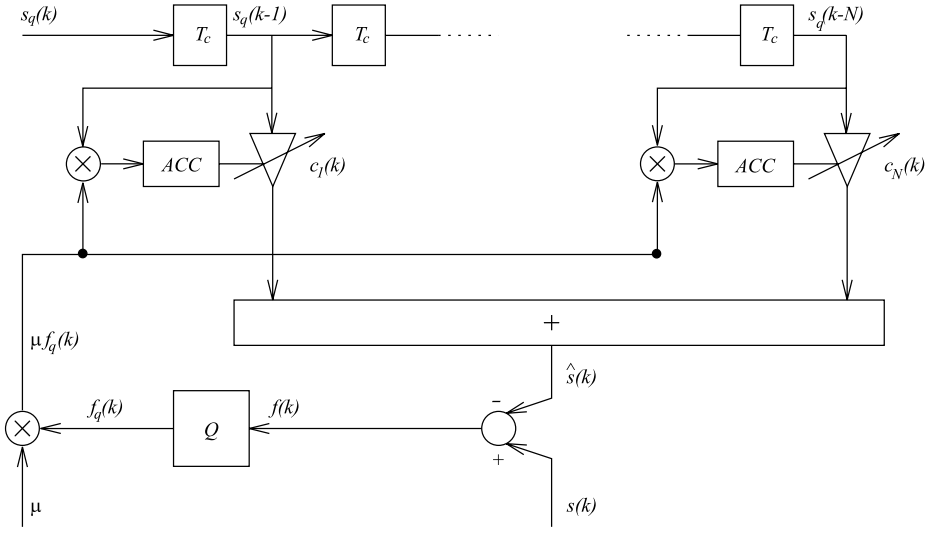


Figure 5.61. LMS adaptive predictor.

Performance

Objective and subjective experiments conducted on speech signals sampled at 8 kHz have indicated that adopting ADPCM rather than PCM leads to a saving of 2 to 3 bits in encoding: for example, a 5-bit ADPCM scheme yields the same quality as a 7-bit PCM. Obviously in both cases the quantizer is non-uniform.

5.5.5 Alternative structures for the predictor

In this section we omit the quantizer and we analyze alternative structures for the predictor. For further study on various signal models (AR, MA and ARMA), we refer to Section 1.12.

All-pole predictor

The predictor considered in the previous section implies an inverse prediction error filter or *synthesis filter* (see Figure 2.9) with transfer function

$$H(z) = \frac{S(z)}{F(z)} = \frac{1}{1 - C(z)} \tag{5.167}$$

which has only poles (neglecting zeros at the origin). The predictor refers to an input model whose samples are given by

$$s(k) = -\sum_{i=1}^{N} \mathrm{a}_i s(k-i) + w(k) \tag{5.168}$$

We consider two cases.

1. $\{w(k)\}$ is *white noise*. Then (5.168) implies an AR(N) model for the input. The optimum predictor that minimizes $E[(s(k) - \hat{s}(k))^2]$ is $C(z) = -\sum_{n=1}^{N} \mathtt{a}_n z^{-n}$ and it yields $f(k) = w(k)$.

2. $\{w(k)\}$ is a *periodic sequence of impulses*,

$$w(k) = A \sum_{n=-\infty}^{+\infty} \delta_{k-nP} \tag{5.169}$$

with $P \gg N$. In this case (5.168) implies that also $\{s(k)\}$ is a periodic signal of period P. The optimum predictor is still $C(z) = -\sum_{n=1}^{N} \mathtt{a}_n z^{-n}$ and it yields $f(k) = w(k)$, equal to a periodic sequence of impulses.

We observe that the two cases model in a simplified manner the input to an all-pole filter whose output is *unvoiced* (case 1) or *voiced* (case 2). The coding scheme that makes use of an *all-pole predictor* is called *linear predictive coding* (LPC) and the prediction error $\{f(k)\}$ is called LPC *residual*.

All-zero predictor

For an MA input model, the prediction signal is given by

$$\hat{s}(k) = \sum_{i=1}^{q} b_i f(k-i) \tag{5.170}$$

Correspondingly from (5.124), for $s_q(k) = s(k)$, the *synthesis filter* has a FIR all zero transfer function

$$H(z) = 1 + \sum_{i=1}^{q} b_i z^{-i} \tag{5.171}$$

Incidentally we note that an approximate LMS adaptation of the coefficients $\{b_i\}$ is given by

$$b_i(k+1) = b_i(k) + \mu_b f_q(k) f_q(k-i) \qquad i = 1, \ldots, q \tag{5.172}$$

Pole-zero predictor

The general case refers to an ARMA(p, q) input model. In this case

$$\hat{s}(k) = \sum_{i=1}^{p} c_i s(k-i) + \sum_{i=1}^{q} b_i f(k-i) \tag{5.173}$$

Correspondingly, we have

$$H(z) = \frac{1 + \sum_{i=1}^{q} b_i z^{-i}}{1 - \sum_{i=1}^{p} c_i z^{-i}} \tag{5.174}$$

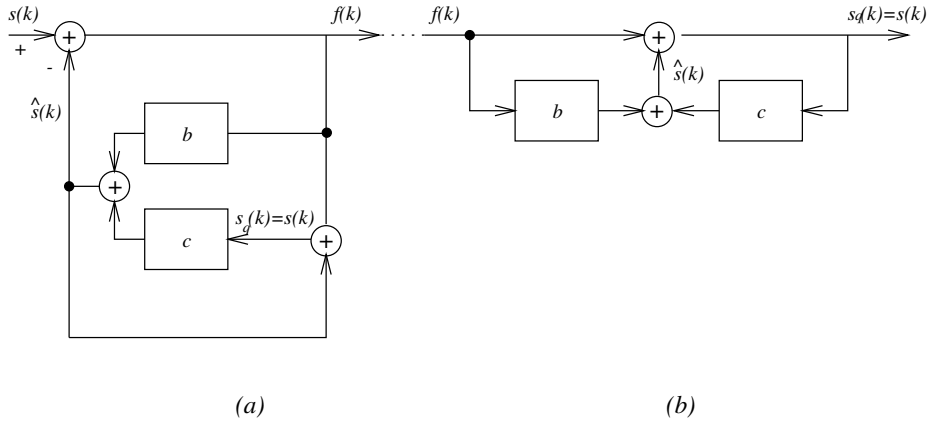


Figure 5.62. Pole-zero predictor: (a) analysis, (b) synthesis.

The equations (5.173) and (5.174) are illustrated in Figure 5.62. For the LMS adaptation of the coefficients in (5.173), we refer to (5.166) for the coefficients $\{c_i\}$ and to (5.172) for the coefficients $\{b_i\}$. This configuration was adopted by the ADPCM G.721 standard at 32 kbit/s (see Table 5.16), that uses an LMS adaptive predictor with 2 poles and 4 zeros; the 5 bit quantizer is adapted by the Jayant scheme.

Pitch predictor

An alternative structure exploits the quasi-periodic behavior, of period P, of voiced spurts of speech. In this case it is convenient to use the estimate

$$\hat{s}(k) = \hat{s}_\ell(k) + \hat{s}_s(k) \tag{5.175}$$

where

$$\hat{s}_\ell(k) = \beta s(k - P) \tag{5.176}$$

is the long-term estimate. In (5.176) β is the *pitch gain*, and P is the *pitch period* expressed in number of samples. Let $\{f_\ell(k)\}$ be the corresponding prediction error

$$f_\ell(k) = s(k) - \beta s(k - P) \tag{5.177}$$

Then, in (5.175),

$$\hat{s}_s(k) = \sum_{i=1}^{N} c_i f_\ell(k - i) \tag{5.178}$$

is the short-term estimate. The prediction error

$$f(k) = s(k) - \hat{s}(k) = f_\ell(k) - \sum_{i=1}^{N} c_i f_\ell(k - i) \tag{5.179}$$

is related to the input $\{s(k)\}$ as shown in the scheme of Figure 5.63.

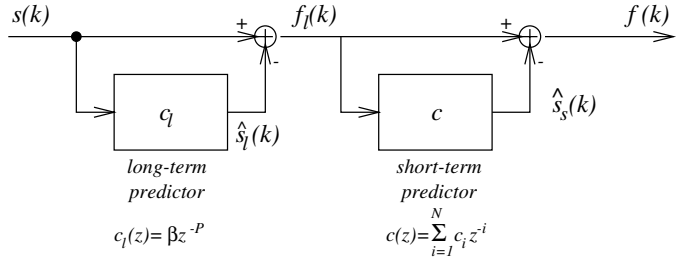


Figure 5.63. Cascade of a long-term predictor with an all-pole predictor.

APC

Because P is usually in the range from 40 to 120 samples, for speech signals sampled at 8 kHz the whole predictor in (5.179) is in fact an all-pole type with a very high order N. The subdivision into two terms, even if not optimum, has the advantage of allowing a very simple computation of the various parameters.

1. Computation of long-term predictor through minimization of the cost function

$$\min_{\beta,P} E[f_\ell(k)^2] = E[(s(k) - \beta s(k-P))^2] \tag{5.180}$$

It follows:[10]

$$P = \arg\max_{n\neq 0} \rho(n) \tag{5.181}$$

where $\rho(n)$ represents the correlation coefficient of $\{s(k)\}$ at lag n, and

$$\beta = \rho(P) \tag{5.182}$$

2. Determination of the short-term predictor through minimization of the cost function:

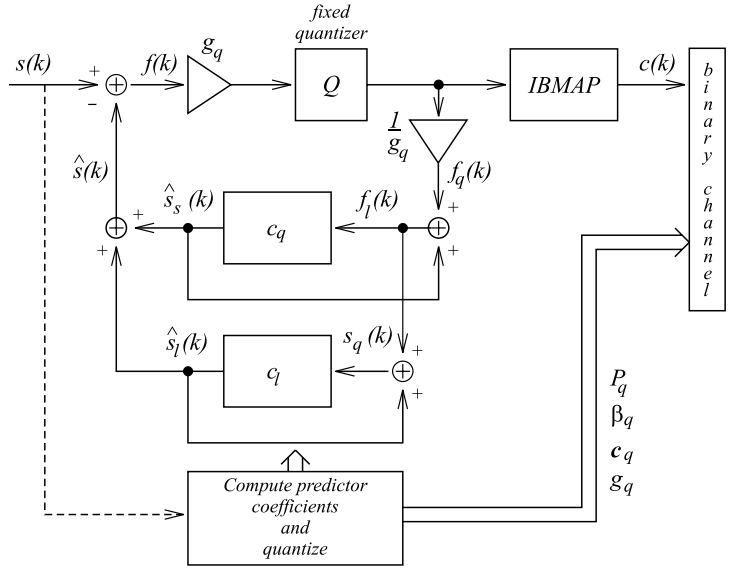
$$\min_{\mathbf{c}} E[f^2(k)] \tag{5.183}$$

From the estimate of the autocorrelation sequence of $\{s(k)\}$, once P and β are determined, the autocorrelation sequence of $\{f_\ell(k)\}$ is easily computed. Then the coefficients $\{c_i\}$ of the long-term predictor can be obtained by solving a system of equations similar to (5.154), where $\mathbf{\Gamma}$ and $\boldsymbol{\rho}$ depend on the autocorrelation coefficients of $\{f_\ell(k)\}$.

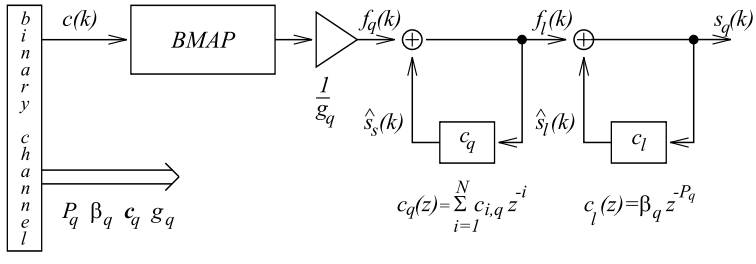
Experimental measurements, initially conducted by Atal [5], have demonstrated that adapting the various coefficients by the *feedforward* method every 5 ms we get high quality speech reproduction with an overall bit rate of only 10 kbit/s, using only a one-bit quantizer.

The encoder and decoder schemes are given in Figure 5.64: they form the *adaptive predictive coding* (APC) scheme which differs from the DPCM for the inclusion of the *long-term predictor*.

[10] For the adopted notation see Footnote 3 on page 441.



(a) Encoder



(b) Decoder

Figure 5.64. Adaptive predictive coding scheme.

For voiced speech, the improvement given by the long-term predictor to lowering the LPC residual is shown in Figure 5.65. Without the long-term predictor the LPC residual presents a peak at every pitch period P; as shown in Figure 5.65c, these peaks are removed by the long-term predictor. The frequency-domain representation of the three signals in Figure 5.65 is given in Figure 5.66. We note that plots in Figures 5.66a and 5.66b exhibit some spectral lines, due to the periodic behavior of the corresponding signals in the time domain, whereas these lines are attenuated in the plot of Figure 5.66c.

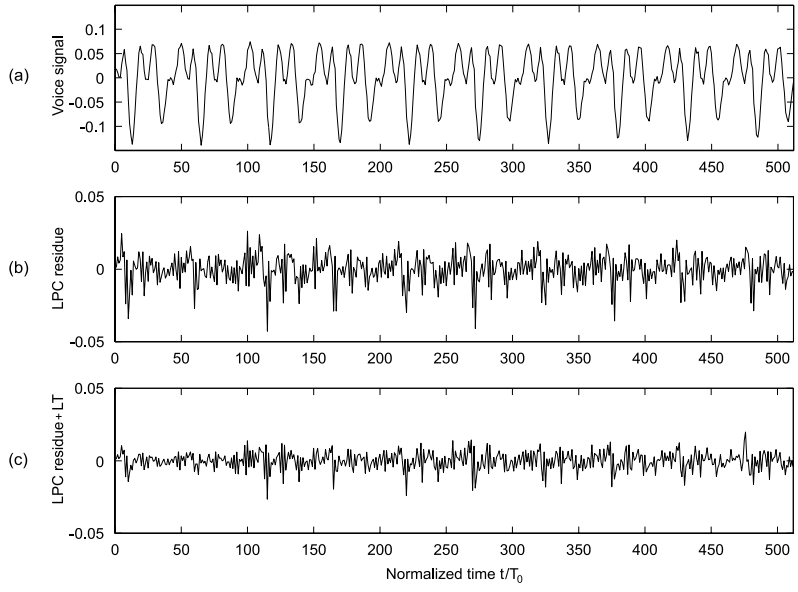


Figure 5.65. (a) Voiced speech, (b) LPC residual, (c) LPC residual with long-term predictor.

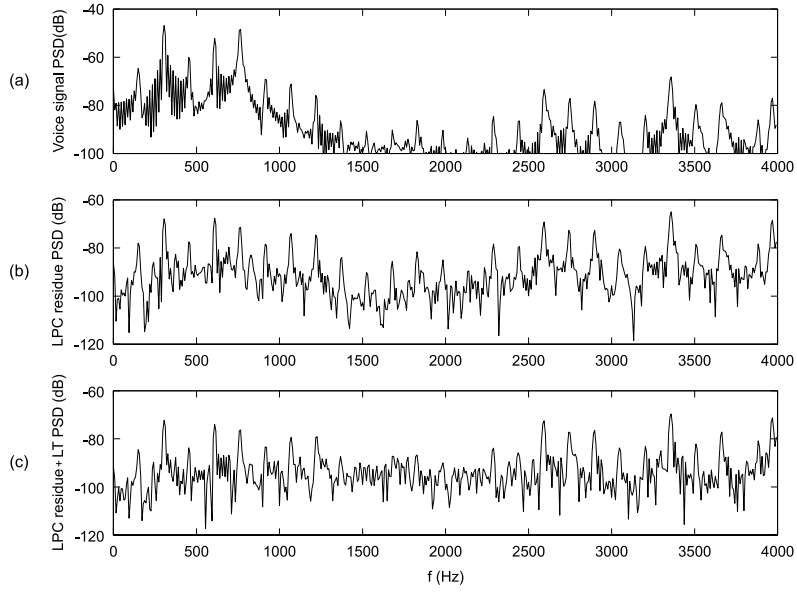


Figure 5.66. DFT of signals of Figure 5.65: (a) voiced speech, (b) LPC residual, (c) LPC residual with long-term predictor.

Other long-term predictors with 2 or 3 coefficients have been proposed: although they are more effective, the determination of their parameters is much more complicate than the approach (5.180). There are also numerous methods that are more robust and effective than (5.181) to determine the pitch period P. To avoid this very laborious computation, all-pole predictors have been proposed with more than 50 coefficients, thus partly assimilating the long-term predictor in the overall predictor.

Two improvements with respect to the basic APC scheme are outlined in the following observations [3].

Observation 5.6
From the standpoint of perception, it is important to have a signal-to-noise ratio that is constant in the frequency domain: this yields the so-called *spectral shaping of the error*, obtained by filtering the residual error so that it is reduced at frequencies where the signal has low energy and enhanced at frequencies where the signal has high energy.

Observation 5.7
In APC, parameters associated with the prediction coefficients $\{c_i\}$ $i = 1, \ldots, N$, are normally sent: for example reflection coefficients (PARCOR), *area functions* or *line spectrum pairs* (LSP).

5.6 Delta modulation

5.6.1 Oversampling and quantization error

For an input signal $s(t)$, $t \in \Re$, WSS random process with bandwidth B, let the sampling period be

$$T_c = \frac{T_0}{F_0} \tag{5.184}$$

where

$$T_0 = \frac{1}{2B} \tag{5.185}$$

and F_0 is the *oversampling factor*. Let

$$x(k) = s(kT_c) \tag{5.186}$$

be the sampled version of $s(t)$, with autocorrelation

$$\mathtt{r}_x(n) = \mathtt{r}_s(nT_c) = \mathtt{r}_s\left(n\frac{T_0}{F_0}\right) \tag{5.187}$$

and power spectral density

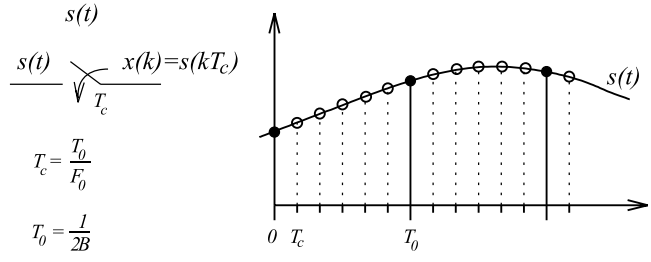
$$\mathcal{P}_x(f) = \sum_{\ell=-\infty}^{+\infty} \mathcal{P}_s\left(f - \ell\frac{F_0}{T_0}\right) \tag{5.188}$$

Equation (5.188) is obtained from (5.187) using (1.90) and the definition (1.247).

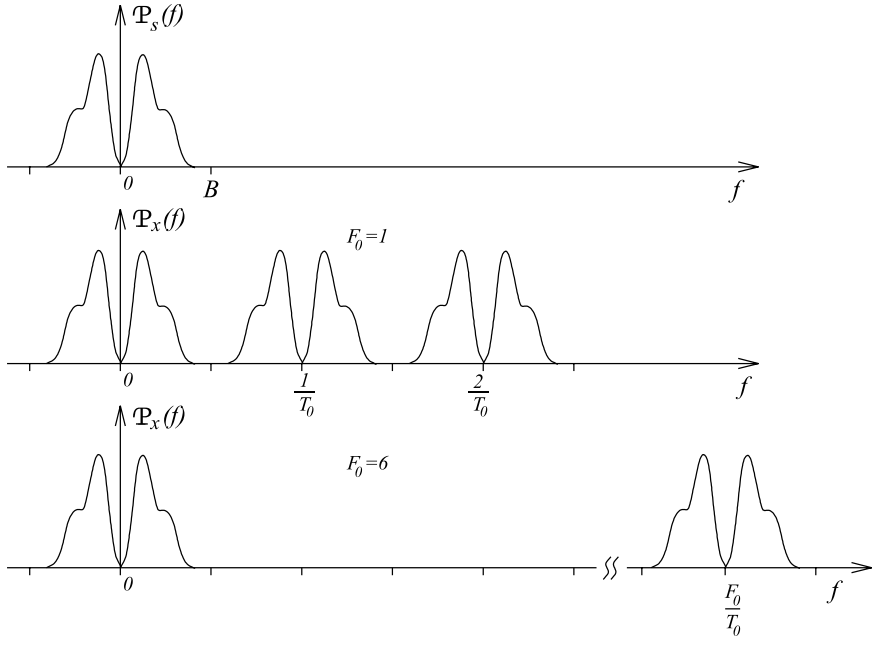
Figure 5.67 shows the effect of oversampling on $\{x(k)\}$. In particular, from (5.187) and from Figure 5.67a we note that by increasing F_0 the samples $\{x(k)\}$ become more correlated; moreover, from (5.188) we have that the spectrum of $\{x(k)\}$ presents images that are more spaced apart from each other.

Let us now quantize $\{x(k)\}$. With reference to Figure 5.68, let $x_q(k)$ be the quantized signal and $e_q(k)$ the corresponding quantization error

$$x_q(k) = x(k) + e_q(k) \tag{5.189}$$



(a) Time domain signal



(b) Power spectral density

Figure 5.67. Effects of oversampling for two values of the oversampling factor F_0.

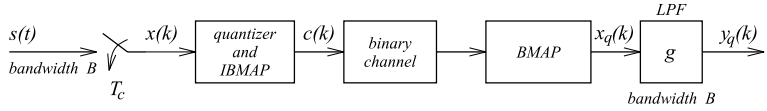


Figure 5.68. General scheme.

For $\{e_q(k)\}$ white with statistical power $\mathtt{M}_{e_q}$, depending only on the number of quantizer levels (see (5.44) and (5.98)), we have

$$\mathcal{P}_{e_q}(f) = \mathtt{M}_{e_q} \frac{T_0}{F_0} \tag{5.190}$$

Consequently, by increasing F_0 the PSD of e_q decreases in amplitude. Moreover, from (5.189), for the non-correlation assumption between x and e_q, we have

$$\mathcal{P}_{x_q}(f) = \mathcal{P}_x(f) + \mathcal{P}_{e_q}(f) \tag{5.191}$$

From (5.189), by filtering $\{x_q(k)\}$ with an ideal lowpass filter g having bandwidth B and unit gain, at the output we will have

$$y_q(k) = x(k) + e_{q,o}(k) \tag{5.192}$$

where $e_{q,o}(k) = e_q * g(k)$ has PSD given by

$$\mathcal{P}_{e_{q,o}}(f) = \mathtt{M}_{e_q} \frac{T_0}{F_0} \operatorname{rect} \frac{f}{2B} \tag{5.193}$$

Then

$$\mathtt{M}_{e_{q,o}} = \frac{\mathtt{M}_{e_q}}{F_0} \tag{5.194}$$

and, with reference to (5.192), the effective signal-to-noise ratio is given by

$$\Lambda_{q,o} = \Lambda_q F_0 \tag{5.195}$$

where $\Lambda_q = \mathtt{M}_x / \mathtt{M}_{e_q}$ is to the signal-to-noise ratio after the quantizer.

In conclusion, under the assumption that the quantization noise is white, oversampling improves performance by a factor F_0. However, the encoder bit rate,

$$R_b = \mathtt{b} \frac{1}{T_c} = \mathtt{b} \frac{F_0}{T_0} \tag{5.196}$$

increases proportionally to F_0: for example, $F_0 = 4$ improves Λ_q by 6 dB, but at the expense of quadrupling the bit rate. Therefore oversampling by large factors is used only locally before PCM or in *compact disk* (CD) applications to simplify the analog interpolation filter at the receiver.

5.6.2 Linear delta modulation (LDM)

Linear delta modulation is a DPCM scheme with oversampled input signal,

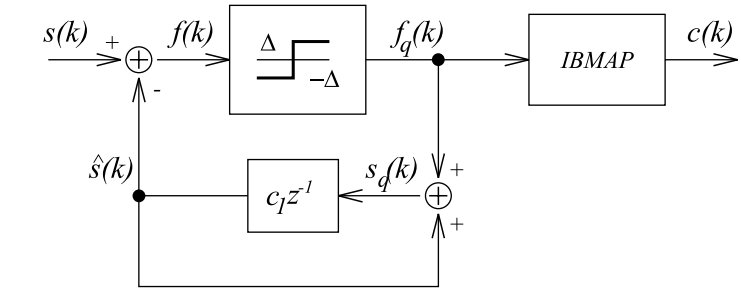
$$\frac{1}{T_c} \gg 2B \tag{5.197}$$

and a quantizer with only two levels ($\mathtt{b} = 1$). Then the encoder bit rate is equal to the sampling rate,

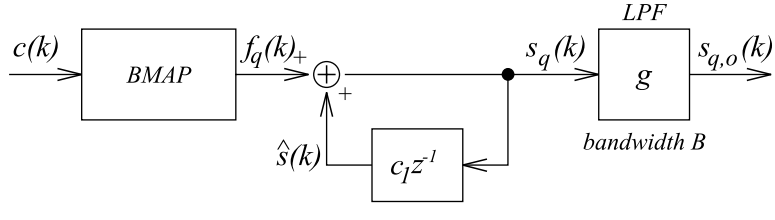
$$R_b = \frac{1}{T_c} \text{ (bit/s)} \tag{5.198}$$

The high value of F_0 implies a high predictability of the input sequence: therefore a predictor with a few coefficients gives a high prediction gain and the quantizer can be reduced to the simplest case of $\mathtt{b} = 1$. We note, moreover, that one-bit code words eliminate the need for framing of the code words at the transmitter and at the receiver, thus simplifying the overall system.

For a predictor with only one coefficient c_1, the coding scheme, which is illustrated in Figure 5.69, is called a *linear delta modulator* (LDM). The following relations hold.



(a) Encoder



(b) Decoder

Figure 5.69. LDM coding scheme.

Encoder:

$$f(k) = s(k) - \hat{s}(k) \tag{5.199}$$

$$f_q(k) = \begin{cases} \Delta \ (c(k) = 1) & \text{if } f(k) \geq 0 \\ -\Delta \ (c(k) = 0) & \text{if } f(k) < 0 \end{cases} \tag{5.200}$$

$$\hat{s}(k+1) = c_1\Big[\hat{s}(k) + f_q(k)\Big] \tag{5.201}$$

Decoder:

$$f_q(k) = \begin{cases} \Delta & c(k) = 1 \\ -\Delta & c(k) = 0 \end{cases} \tag{5.202}$$

$$s_q(k) = c_1 s_q(k-1) + f_q(k) \tag{5.203}$$

$$s_{q,o}(k) = (s_q * g)(k) \tag{5.204}$$

The system is based on three parameters ($1/T_c$, Δ, and c_1), that are appropriately selected. For example the choice of $c_1 = 1$ considerably simplifies (5.201) and (5.203), that become simple accumulator expressions. However, typically we set $c_1 \leq 1$ so that random transmission errors do not propagate indefinitely in the reconstruction signal (5.203).

LDM implementation

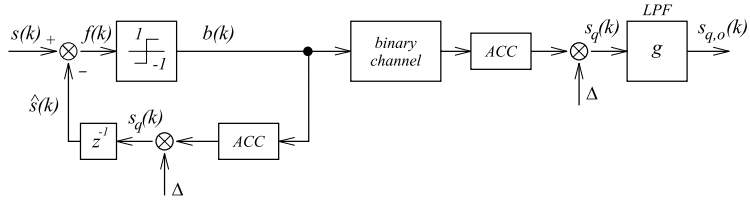
Digital implementation. An implementation of the scheme of Figure 5.69 for $c_1 = 1$ is given in Figure 5.70a. Letting $\mathtt{b}(k) = \text{sgn}[f(k)]$, this implementation involves the accumulation of $\{b(i)\}$, $i \leq k$, by an *up-down counter* (ACC). The accumulated value is proportional to $s_q(k)$.

Mixed analog-digital implementation. An alternative to the previous scheme, which requires carrying out the operation $f(k) = s(k) - \hat{s}(k)$ in the digital domain, is obtained by placing a DAC after the accumulator and carrying out the comparison in the analog domain, as illustrated in Figure 5.70b. Note that the decoder consists simply of an accumulator followed by a DAC, which performs the function of the filter g, to eliminate the out-of-band noise, and of the gain Δ of the quantizer step size.

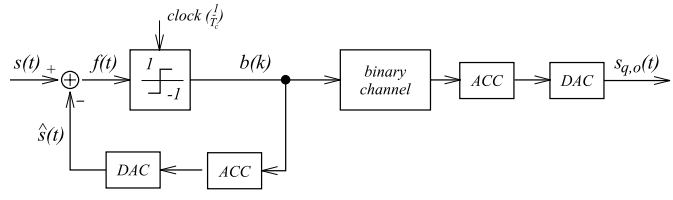
Analog implementation. In many applications it is convenient to implement the analog accumulator by an integrator: thus we obtain the implementation of Figure 5.70c, where the DAC is often a simple holder. At the receiver, the integrator has a bandwidth B equal to that of the input signal.

Choice of system parameters

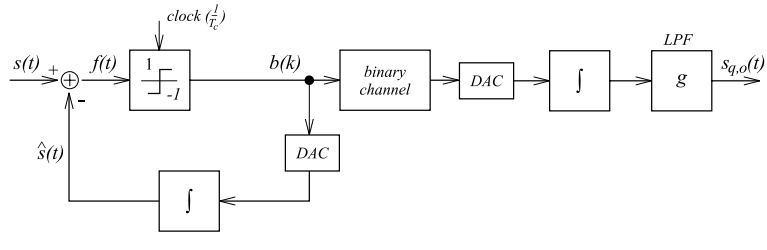
With reference to Figure 5.70a and Figure 5.71, we will now establish a relation between the various parameters of an LDM. As for the DPCM scheme, also for the LDM we need to choose a small Δ to obtain low granular noise; to get instead a small slope overload



(a) Digital implementation



(b) Mixed digital-analog implementation



(c) Analog implementation

Figure 5.70. LDM implementations.

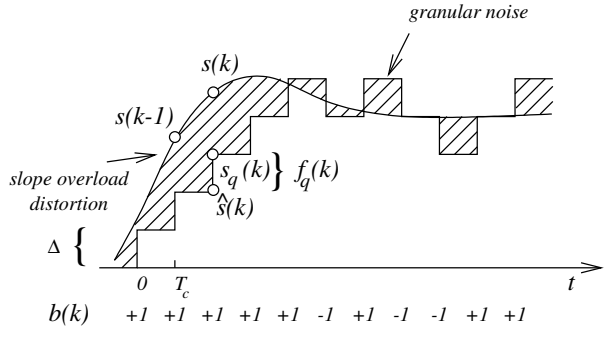


Figure 5.71. Graphic representation of LDM.

distortion a large Δ is needed, so that

$$\frac{\Delta}{T_c} \geq \max_t \left| \frac{\mathrm{d}}{\mathrm{dt}} s(t) \right| \tag{5.205}$$

and the reconstruction signal $\{s_q(k)\}$ can follow very rapid changes of the input signal. In other words, for a given value of $\max_t |(\mathrm{d}/\mathrm{dt})s(t)|$, if we reduce Δ to decrease the granular noise we must also reduce T_c to limit the slope overload distortion; as a consequence we have that LDM requires a very high oversampling factor F_0 to give satisfactory performance. We note that doubling the sampling rate we obtain an increment in $\Lambda_{q,o}$ of 9 dB: 3 dB are due to filtering of the out-of-band noise and 6 dB to the reduction of the granular noise, as Δ can be halved.

In speech applications with a bandwidth of about 3 kHz, to have a $\Lambda_{q,o}$ of approximately 35 dB we need $F_0 \geq 33$, which requires a sampling rate of the order of 200 kHz. The optimum value of Δ is given approximately by

$$\Delta_{opt} = \sqrt{2\mathtt{M}_s(1 - \rho_x(1))}\ln(2F_0) \tag{5.206}$$

where $\rho_x(1) = \mathtt{r}_s(T_c)/\mathtt{r}_s(0)$.

5.6.3 Adaptive delta modulation (ADM)

To reduce both granular noise and slope overload distortion, the only possibility in the LDM is to reduce T_c. An alternative is represented by an adaptive scheme for the step size Δ, as shown in Figure 5.72.

In particular, the Jayant algorithm uses the following relation

$$\Delta(k) = \mathtt{p}\Delta(k-1) \tag{5.207}$$

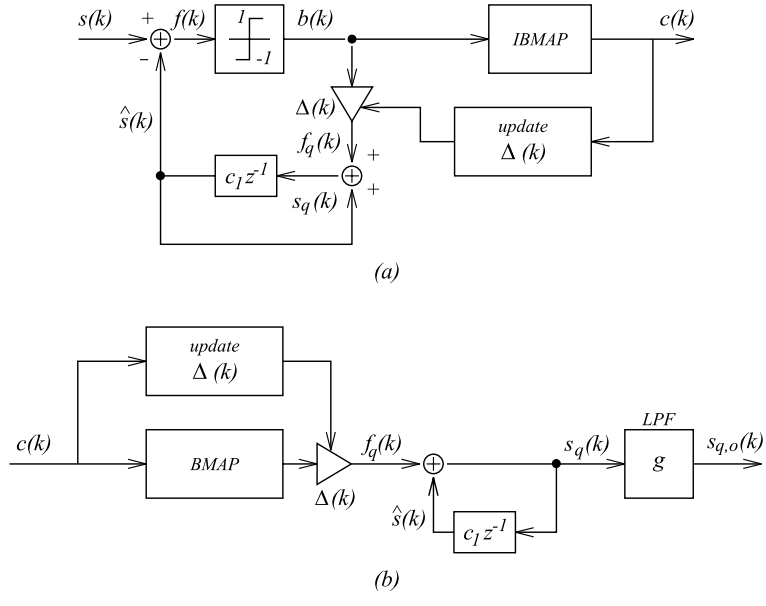


Figure 5.72. ADM coding scheme: (a) encoder, (b) decoder.

where

$$\mathrm{p} = \begin{cases} \mathrm{p}_o > 1 & \text{if } c(k) = c(k-1) \text{ (\textit{slope overload})} \\ \mathrm{p}_g < 1 & \text{if } c(k) \neq c(k-1) \text{ (\textit{granular noise})} \end{cases} \tag{5.208}$$

We note that in this scheme $\Delta(k)$ also depends on $c(k)$. The following relations between the signals of Figure 5.72 hold.

Encoder:

$$f(k) = s(k) - \hat{s}(k) \tag{5.209}$$

$$b(k) = \operatorname{sgn} f(k) \tag{5.210}$$

$$\Delta(k) = \mathrm{p}\Delta(k-1) \tag{5.211}$$

$$f_q(k) = \Delta(k)b(k) \tag{5.212}$$

$$\hat{s}(k+1) = c_1[\hat{s}(k) + f_q(k)] \tag{5.213}$$

Decoder:

$$\Delta(k) = \mathrm{p}\Delta(k-1) \tag{5.214}$$

$$f_q(k) = \Delta(k)b(k) \tag{5.215}$$

$$s_q(k) = c_1 s_q(k-1) + f_q(k) \tag{5.216}$$

$$s_{q,o}(k) = s_q * g(k) \tag{5.217}$$

Typical values for p_o and p_g are given by $1.25 < \mathrm{p}_o < 2$ and $\mathrm{p}_o\mathrm{p}_g \leq 1$. A graphic representation of ADM encoding is shown in Figure 5.73.

Experiments on speech signals show that by doubling the sampling rate in the ADM we get an improvement of 10 dB in Λ_q. In some applications ADM encoding is preferred to PCM because of its simple implementation, in spite of the higher bit rate.

Continuously variable slope delta modulation (CVSDM)

An alternative to the adaptation (5.207) is given by the equation

$$\Delta(k) = \begin{cases} \alpha\Delta(k-1) + D_2 & \text{if } c(k) = c(k-1) = c(k-2) \text{ (\textit{slope overload})} \\ \alpha\Delta(k-1) + D_1 & \text{otherwise} \end{cases} \tag{5.218}$$

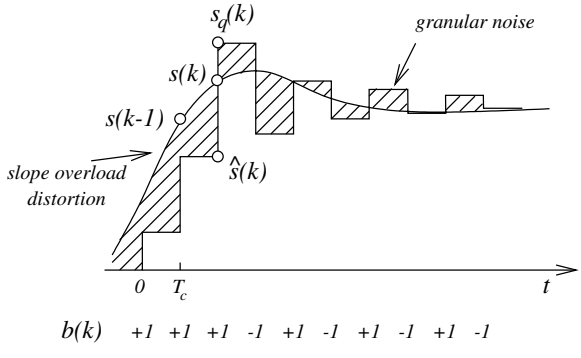


Figure 5.73. Graphic representation of ADM.

where $0 < \alpha \leq 1$, and D_1 and D_2 are suitable positive parameters with $D_2 \gg D_1$. The value α controls the speed of adaptation. The main difficulty of this scheme is the sensitivity of $\{s_q(k)\}$ to transmission errors, especially for $\alpha = 1$; choosing $\alpha < 1$ mitigates the effects of transmission errors, at the expense of worse performance.

ADM with second-order predictors

With the aim of improving system performance, in some cases second-order predictors are used, where

$$\hat{s}(k) = c_1 s_q(k-1) + c_2 s_q(k-2) \tag{5.219}$$

The transfer function of the synthesis filter is given by

$$H(z) = \frac{1}{1 - c_1 z^{-1} - c_2 z^{-2}} \tag{5.220}$$

The function $H(z)$ can be split into the product of two first-order terms,

$$H(z) = \frac{1}{(1 - \mathsf{p}_1 z^{-1})(1 - \mathsf{p}_2 z^{-1})} \tag{5.221}$$

If p_1 and p_2 are real with $0 < \mathsf{p}_1, \mathsf{p}_2 \leq 1$, the decoder is equivalent to the cascade of two leaky integrators. The problem now is to determine the slope overload condition from the sequence $\{c(k)\}$.

5.6.4 PCM encoder via LDM

We consider an alternative scheme to the three implementations of Section 5.2.2 to generate a linear PCM encoded signal, that employs the LDM implementation of Figure 5.70c, as illustrated in Figure 5.74. It is sufficient to accumulate $\{b(k)\}$ to obtain a PCM representation of the input $s(t)$; using a decimator filter, that is a lowpass filter followed by a downsampler, we obtain the PCM output signal sampled at the minimum rate $1/T_0$. Disregarding the filtering effect on noise, that brings a gain of $10 \log_{10} F_0$ dB, we observe that to generate a PCM signal $\{c_{PCM}(k)\}$ with an accuracy of b bits, the oversampling factor F_0 must be at most equal to 2^{b} and, in general,

$$1 \ll F_0 \leq 2^{\mathsf{b}} \tag{5.222}$$

For example, for $\mathsf{b} = 8$ and $1/T_0 = 8$ kHz, this means $1/T_c = F_0/T_0 \simeq 2$ MHz.

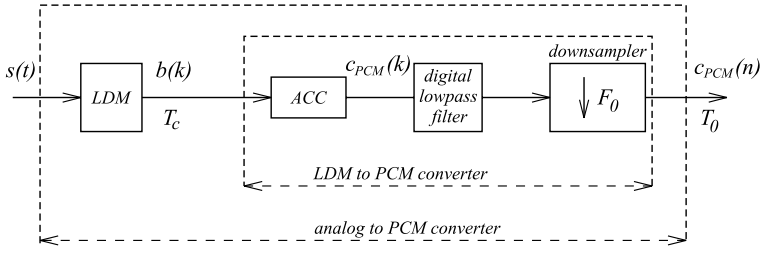
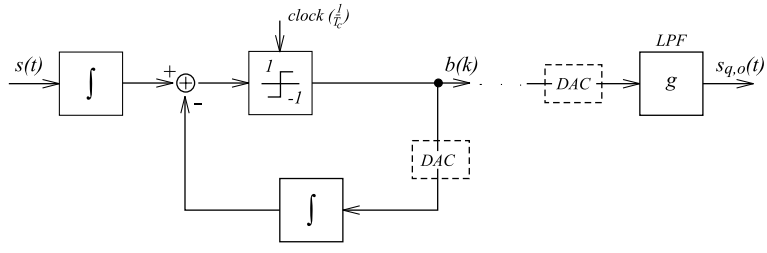
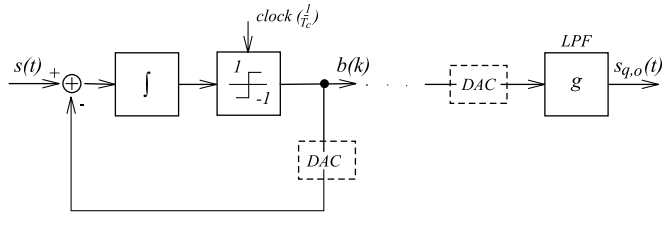


Figure 5.74. Linear PCM encoder via LDM.



(a) Derivation from an LDM scheme with pre-emphasis.



(b) Final scheme.

Figure 5.75. ΣDM coding scheme.

5.6.5 Sigma delta modulation (ΣDM)

With reference to the scheme of Figure 5.70c, to enhance the low frequency components of speech signals we can insert a pre-emphasis integrator before the LDM encoder. A differentiator then has to be inserted at the LDM decoder, which simplifies the LDM integrator: therefore the decoder becomes a simple lowpass filter. Thus we get the general scheme of Figure 5.75a and the simplified scheme of Figure 5.75b: we note that the DACs are simple holders of binary signals.

It is interesting to observe the simplicity of the ΣDM implementation. Moreover, recalling that the spectrum of quantization noise in PCM and LDM is flat, ΣDM presents the advantage that the noise is colored and for the most part is found outside the passband of the desired signal. Therefore it can be removed to a large extent by a simple lowpass filter.

One of the most frequent applications of ΣDM is in linear PCM encoders where, similarly to the scheme of Figure 5.74, it is sufficient to employ a ΣDM followed by a digital decimator filter with input the binary signal $\{b(k)\}$. Note that, with respect to the scheme of Figure 5.74, the accumulator has been removed.

5.7 Coding by modeling

In the coding schemes investigated so far, PCM, DPCM, and their variations, the objective is to reproduce at the decoder a waveform that is as close as possible to the input signal. We now take a different approach and, given the general coding scheme of Figure 5.76,

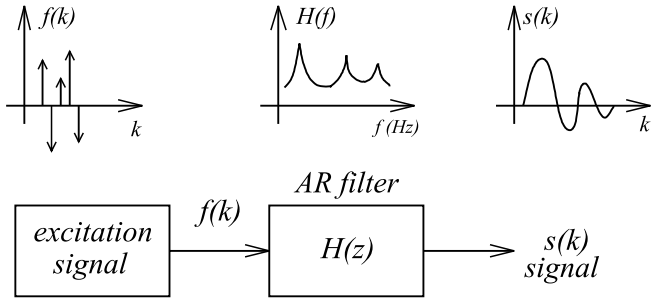


Figure 5.76. Basic scheme of coding by modeling.

the source $\{s(k)\}$, for example speech, is modeled by an AR(N) linear system

$$H(z) = \frac{\sigma}{1 - \sum_{i=1}^{N} c_i z^{-i}} \tag{5.223}$$

with input $\{f(k)\}$.

In (5.223) the coefficients $\{c_i\}$ and σ are obtained by the prediction algorithms of Section 2.2. In particular, as we will assume that $\{f(k)\}$ has unit statistical power, the standard deviation in (5.223) is given by

$$\sigma = \sqrt{J_N} \tag{5.224}$$

where J_N is the statistical power of the prediction error.

The difference among the various coding schemes consists in the form of excitation. Three examples follow.

Regular pulse excited (RPE). The excitation signal consists of a train of undersampled impulses, derived from the residual signal.

Multipulse LP (MELP). The excitation signal consists of a certain number of impulses with suitable amplitude and lag.

Codebook excited linear prediction (CELP). The excitation signal is selected by a collection of possible waveforms stored in a table.

We will now analyze in detail some coding schemes. For further study we refer the reader to [3, 6].

Vocoder or LPC

The general scheme for the conventional LPC, known also as the LPC *vocoder*, is illustrated in Figure 5.77. At the encoder, the signal is classified as voiced or unvoiced, and the LCP parameters are extracted, together with the pitch period P for the voiced case; however,

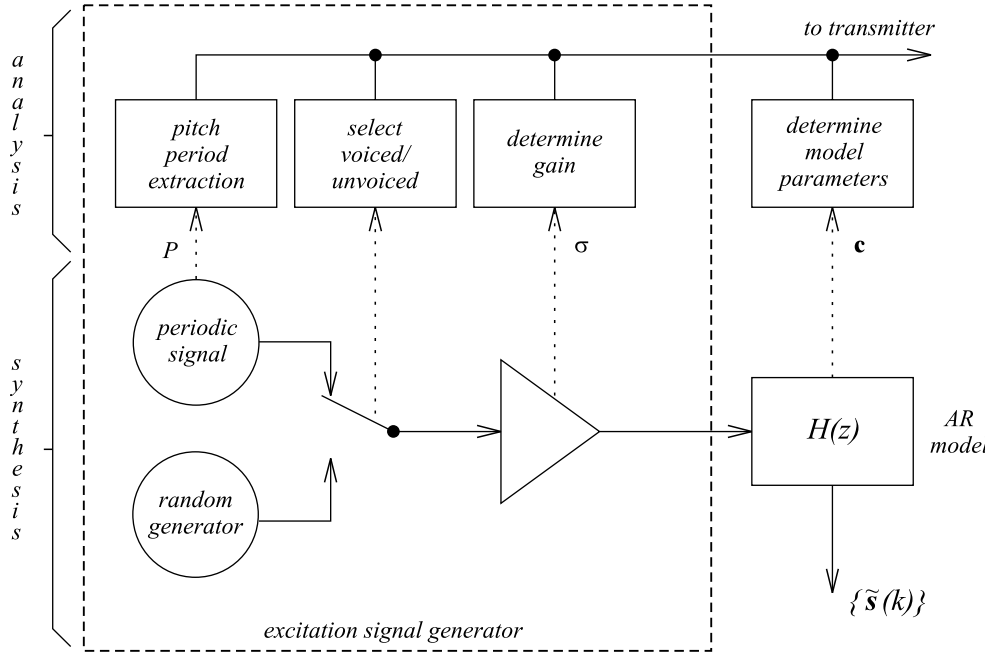


Figure 5.77. Vocoder or LPC scheme.

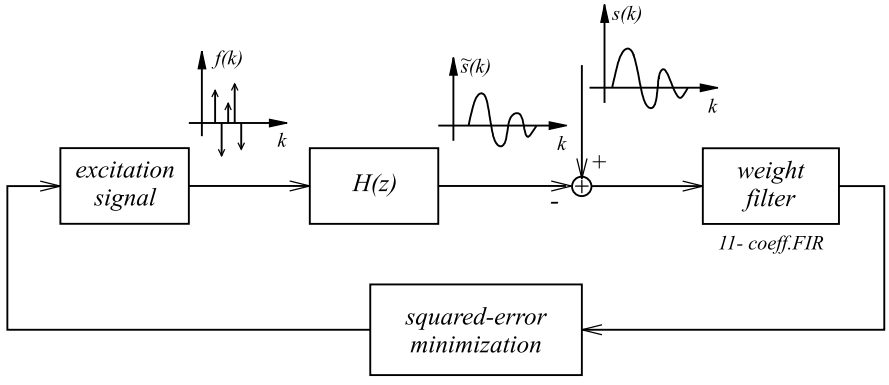
the prediction residual error is not transmitted. At the decoder, for the voiced case a train of impulses with period P is produced, whereas for the unvoiced case white noise is produced. The excitation is then filtered by the AR filter to generate the reconstruction signal.

In an early LPC scheme for military radio applications (LPC-10), the input signal sampled at 8 kHz is segmented into blocks of 180 samples. For the analysis of the LPC parameters the covariance method is used; overall 54 bits per block are needed with a bit rate of 2400 bit/s.

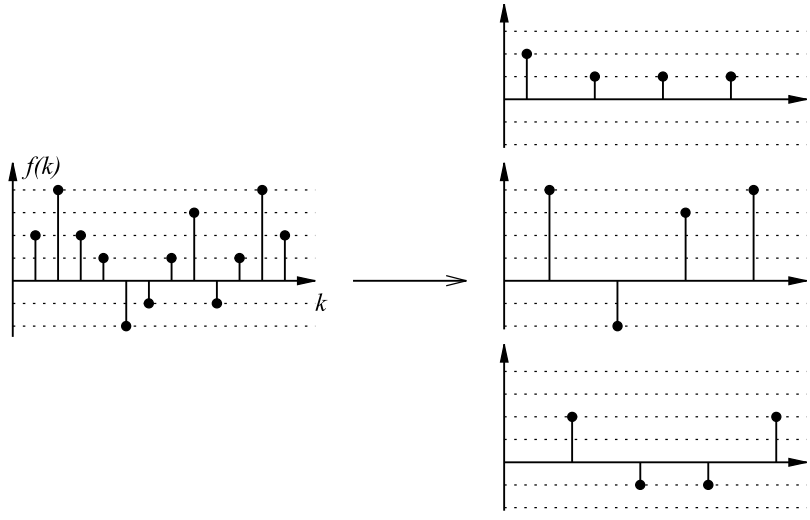
RPE coding

The RPE coding scheme, illustrated in Figure 5.78a, is a particular case of *residual excited LP* (RELP) coding in which the excitation is obtained by downsampling the prediction residual error by a factor of 3, as shown in Figure 5.78b; the excitation sequence is then quantized using a 3-bit adaptive non-uniform quantizer. The choice of the best of the three subsequences (actually four are used in practice) is made by the *analysis-by-synthesis* (ABS) approach, where all the excitations are tried: the best is that which produces the output "closest" to the original signal.

The standard ETSI for GSM (06.10) includes also a long-term predictor as the one of Figure 5.64. The bit rate is 13 kbit/s, with a latency lower than 80 ms, operating with blocks of 160 samples.



(a) Basic scheme.



(b) Residual error decimation in three subsequences.

Figure 5.78. RPE coding scheme.

CELP coding

As shown in Figure 5.79, the excitations belong to a codebook obtained in a "random" way, or by *vector quantization* (see Section 5.8) of the residual signal. The choice of the excitation (index of the codebook) is made by the ABS approach, trying to minimize the output of the weighting filter; also in this case the predictor includes a long term component.

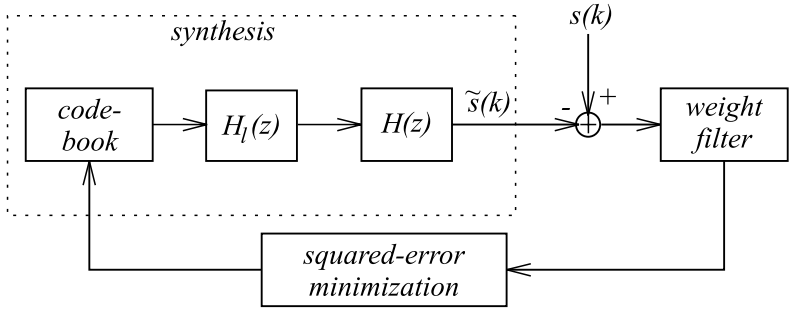


Figure 5.79. CELP coding scheme.

Multipulse coding

It is similar to CELP coding, with the difference that the minimization procedure is used to determine the position and amplitude of a specific number of impulses. The analysis procedure is less complex than that of the CELP scheme.

5.8 Vector quantization (VQ)

Vector quantization (VQ) is introduced as a natural extension of the scalar quantization (SQ) concept. However, using multidimensional signals opens the way to many techniques and applications that are not found in the scalar case [7, 8].

The basic concept is that of associating with an input vector $\mathbf{s} = [s_1, \ldots, s_N]^T$, generic sample of a vector random process $\mathbf{s}(k)$, a reproduction vector $\mathbf{s}_q = \mathbf{Q}[\mathbf{s}]$ chosen from a finite set of L elements (*code vectors*), $\mathcal{A} = \{\mathbf{Q}_1, \ldots, \mathbf{Q}_L\}$, called codebook, so that a given distortion measure $d(\mathbf{s}, \mathbf{Q}[\mathbf{s}])$ is minimized.

Figure 5.80 exemplifies the encoder and decoder functions of a VQ scheme. The encoder computes the distortion associated with the representation of the input vector $\mathbf{s}$ by each

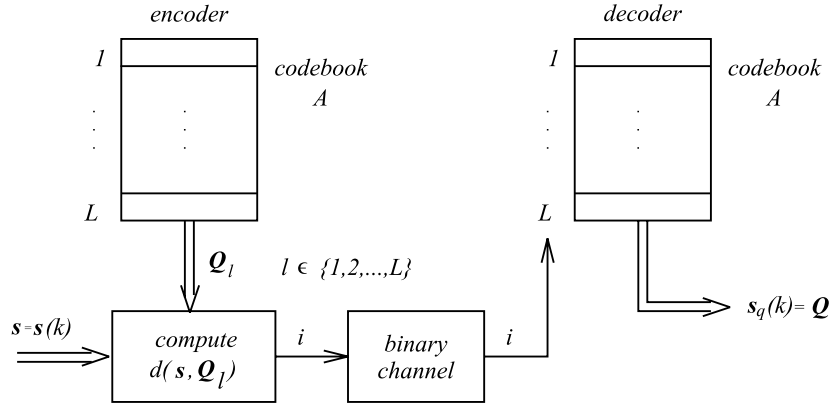


Figure 5.80. Block diagram of a vector quantizer.

reproduction vector of $\mathcal{A}$ and decides for the vector $\mathbf{Q}_i$ of the codebook $\mathcal{A}$ that minimizes it; the decoder associates the vector $\mathbf{Q}_i$ to the index i received. We note that the information transmitted over the digital channel identifies the code vector $\mathbf{Q}_i$: therefore it depends only on the codebook size L and not on N, dimension of the code vectors.

An example of input vector $\mathbf{s}$ is obtained by considering N samples at a time of a speech signal, $\mathbf{s}(k) = [s(kNT_c), \ldots, s((kN - N + 1)T_c)]^T$, or the N LPC coefficients, $\mathbf{s}(k) = [c_1(k), \ldots, c_N(k)]^T$, associated with an observation window of a signal.

5.8.1 Characterization of VQ

Considering the general case of complex-valued signals, a vector quantizer is characterized by

- *Source or input vector* $\mathbf{s} = [s_1, s_2, \ldots, s_N]^T \in \mathbb{C}^N$.
- *Codebook* $\mathcal{A} = \{\mathbf{Q}_i\}$, $i = 1, \ldots, L$, where $\mathbf{Q}_i \in \mathbb{C}^N$ is a code vector.
- *Distortion measure* $d(\mathbf{s}, \mathbf{Q}_i)$.
- *Quantization rule* (minimum distortion)

$$\mathbf{Q} : \mathbb{C}^N \to \mathcal{A} \quad \text{with} \quad \mathbf{Q}_i = \mathbf{Q}[\mathbf{s}] \quad \text{if } i = \arg\min_{\ell} d(\mathbf{s}, \mathbf{Q}_\ell) \tag{5.225}$$

Definition 5.1 (Partition of the source space)
The equivalence relation

$$Q = \{(\mathbf{s}_1, \mathbf{s}_2) : \mathbf{Q}[\mathbf{s}_1] = \mathbf{Q}[\mathbf{s}_2]\} \tag{5.226}$$

which associates input vector pairs having the same reproduction vector, identifies a partition $\mathcal{R} = \{R_1, \ldots, R_L\}$ of the source space $\mathbb{C}^N$, whose elements are the sets

$$R_\ell = \{\mathbf{s} \in \mathbb{C}^N : \mathbf{Q}[\mathbf{s}] = \mathbf{Q}_\ell\} \qquad \ell = 1, \ldots, L \tag{5.227}$$

The sets $\{R_\ell\}$, $\ell = 1, \ldots, L$, are called *Voronoi regions*.

It can be easily demonstrated that the sets $\{R_\ell\}$ are non-overlapping and cover the entire space $\mathbb{C}^N$:

$$\bigcup_{\ell=1}^{L} R_\ell = \mathbb{C}^N \quad R_i \cap R_j = \emptyset \qquad \forall i \neq j \tag{5.228}$$

In other words, as indicated by (5.227) every subset R_ℓ contains all input vectors associated by the quantization rule with the code vector $\mathbf{Q}_\ell$. An example of partition for $N = 2$ and $L = 4$ is illustrated in Figure 5.81.

Parameters determining VQ performance

We define the following parameters.

- Quantizer rate

$$R_q = \log_2 L \text{ (bit/vector) or (bit/symbol)} \tag{5.229}$$

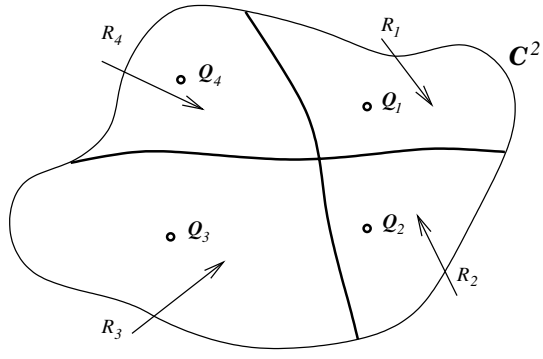


Figure 5.81. Partition of the source space $\mathbb{C}^2$ in four subsets or Voronoi regions.

- Rate per dimension

$$R_I = \frac{R_q}{N} = \frac{\log_2 L}{N} \text{ (bit/sample)} \tag{5.230}$$

- Rate in bit/s

$$R_b = \frac{R_I}{T_c} = \frac{\log_2 L}{NT_c} \text{ (bit/s)} \tag{5.231}$$

where T_c denotes the time interval between two consecutive samples of a vector. In other words, in (5.231) NT_c is the sampling period of the vector sequence $\{\mathbf{s}(k)\}$.

- Distortion

$$d(\mathbf{s}, \mathbf{Q}_i) \tag{5.232}$$

The distortion is a non-negative scalar function of a vector variable,

$$d : \mathbb{C}^N \times \mathcal{A} \to \Re_+ \tag{5.233}$$

If the input process $\mathbf{s}(k)$ is stationary and the probability density function $p_{\mathbf{s}}(\mathbf{a})$ is known, we can compute the mean distortion as

$$D(\mathcal{R}, \mathcal{A}) = E[d(\mathbf{s}, \mathbf{Q}[\mathbf{s}])] \tag{5.234}$$

$$= \sum_{\ell=1}^{L} E[d(\mathbf{s}, \mathbf{Q}_i) \mid \mathbf{s} \in R_\ell] P[\mathbf{s} \in R_\ell] \tag{5.235}$$

$$= \sum_{\ell=1}^{L} \bar{d}_\ell P[\mathbf{s} \in R_\ell] \tag{5.236}$$

where

$$\bar{d}_\ell = \frac{\displaystyle\int_{R_\ell} d(\mathbf{a}, \mathbf{Q}_\ell)\, p_{\mathbf{s}}(\mathbf{a})\, d\mathbf{a}}{\displaystyle\int_{R_\ell} p_{\mathbf{s}}(\mathbf{a})\, d\mathbf{a}} \tag{5.237}$$

If the source is also ergodic we obtain

$$D(\mathcal{R}, \mathcal{A}) = \lim_{K\to\infty} \frac{1}{K} \sum_{k=1}^{K} d(\mathbf{s}(k), \mathbf{Q}[\mathbf{s}(k)]) \tag{5.238}$$

In practice we always assume that the process $\{\mathbf{s}\}$ is stationary and ergodic, and we use the average distortion (5.238) as an estimate of the expectation (5.234).

Defining

$$\mathbf{Q}_i = [Q_{i,1}, Q_{i,2}, \ldots, Q_{i,N}]^T \tag{5.239}$$

we give below two measures of distortion of particular interest.

1. Distortion as the ℓ_ν norm to the μ-th power:

$$d(\mathbf{s}, \mathbf{Q}_i) = \|\mathbf{s} - \mathbf{Q}_i\|_\nu^\mu = \left[\sum_{n=1}^{N} |s_n - Q_{i,n}|^\nu\right]^{\mu/\nu} \tag{5.240}$$

   The most common version is the *squared distortion*:[11]

$$d(\mathbf{s}, \mathbf{Q}_i) = \|\mathbf{s} - \mathbf{Q}_i\|_2^2 = \sum_{n=1}^{N} |s_n - Q_{i,n}|^2 \tag{5.241}$$

2. Itakura–Saito distortion:

$$d(\mathbf{s}, \mathbf{Q}_i) = (\mathbf{s} - \mathbf{Q}_i)^H \mathbf{R}_{\mathbf{s}} (\mathbf{s} - \mathbf{Q}_i) = \sum_{n=1}^{N}\sum_{m=1}^{N} (s_n - Q_{i,n})^* [\mathbf{R}_{\mathbf{s}}]_{n,m} (s_m - Q_{i,m}) \tag{5.242}$$

   where $\mathbf{R}_{\mathbf{s}}$ is the autocorrelation matrix of the vector $\mathbf{s}^*(k)$, defined in (1.346), with elements $[\mathbf{R}_{\mathbf{s}}]_{n,m}$, $n, m = 1, 2, \ldots, N$.

Comparison between VQ and scalar quantization

Defining the mean distortion per dimension as $\widetilde{D} = D/N$, for a given rate R_I we find (see [9] and references therein)

$$\frac{\widetilde{D}_{SQ}}{\widetilde{D}_{VQ}} = F(N)\, S(N)\, M(N) \tag{5.243}$$

[11] Although the same symbol is used, the metric defined by (5.241) is the square of the Euclidean distance (1.38).

where

- $F(N)$ is the space filling gain. In the scalar case the partition regions must necessarily be intervals. In an N dimensional space, R_i can be "shaped" very closely to a sphere. The asymptotic value for $N \to \infty$ equals $F(\infty) = 2\pi e/12 = 1.4233 = 1.53\,\text{dB}$.
- $S(N)$ is the gain related to the shape of $p_{\mathbf{s}}(\mathbf{a})$, defined as[12]

$$S(N) = \frac{\|\tilde{p}_{\mathbf{s}}(\mathbf{a})\|_{1/3}}{\|\tilde{p}_{\mathbf{s}}(\mathbf{a})\|_{N/(N+2)}} \tag{5.245}$$

  where $\tilde{p}_{\mathbf{s}}(\mathbf{a})$ is the probability density function of the input $\mathbf{s}$ considered with uncorrelated components. $S(N)$ does not depend on the variance of the random variables of $\mathbf{s}$, but only on the norm order $N/(N+2)$ and shape $\tilde{p}_{\mathbf{s}}(\mathbf{a})$. For $N \to \infty$, we obtain

$$S(\infty) = \frac{\|\tilde{p}_{\mathbf{s}}(\mathbf{a})\|_{1/3}}{\|\tilde{p}_{\mathbf{s}}(\mathbf{a})\|_1} \tag{5.246}$$

- $M(N)$ is the memory gain, defined as

$$M(N) = \frac{\|\tilde{p}_{\mathbf{s}}(\mathbf{a})\|_{N/(N+2)}}{\|p_{\mathbf{s}}(\mathbf{a})\|_{N/(N+2)}} \tag{5.247}$$

  where $p_{\mathbf{s}}(\mathbf{a})$ is the probability density function of the input $\mathbf{s}$. The expression of $M(N)$ depends on the two functions $\tilde{p}_{\mathbf{s}}(\mathbf{a})$ and $p_{\mathbf{s}}(\mathbf{a})$, which differ for the correlation among the various vector components; obviously if the components of $\mathbf{s}$ are statistically independent, we have $M(N) = 1$; otherwise $M(N)$ increases as the correlation increases.

5.8.2 Optimum quantization

Our objective is to design a vector quantizer, choosing the code vectors of the codebook $\mathcal{A}$ and the partitioning $\mathcal{R}$ so that the mean distortion given by (5.234) is minimized.

Two necessary conditions arise.

Rule A (Optimum partition). Assuming the codebook $\mathcal{A} = \{\mathbf{Q}_1, \ldots, \mathbf{Q}_L\}$ fixed, we want to find the optimum partition $\mathcal{R}$ that minimizes $D(\mathcal{R}, \mathcal{A})$. Observing (5.234) the solution is given by

$$R_i = \{\mathbf{s} : d(\mathbf{s}, \mathbf{Q}_i) = \min_{\mathbf{Q}_\ell \in \mathcal{A}} d(\mathbf{s}, \mathbf{Q}_\ell)\} \qquad i = 1, \ldots, L \tag{5.248}$$

As illustrated in Figure 5.82, R_i contains all the points $\mathbf{s}$ "nearest" to $\mathbf{Q}_i$.

[12] Extending (5.240) to the continuous case we obtain

$$\|\tilde{p}_{\mathbf{s}}(\mathbf{a})\|^{\mu/\nu} = \left[\int \cdots \int \tilde{p}_{\mathbf{s}}^{\nu}(a_1, \ldots, a_N)\, da_1 \ldots da_N\right]^{\mu/\nu} \tag{5.244}$$

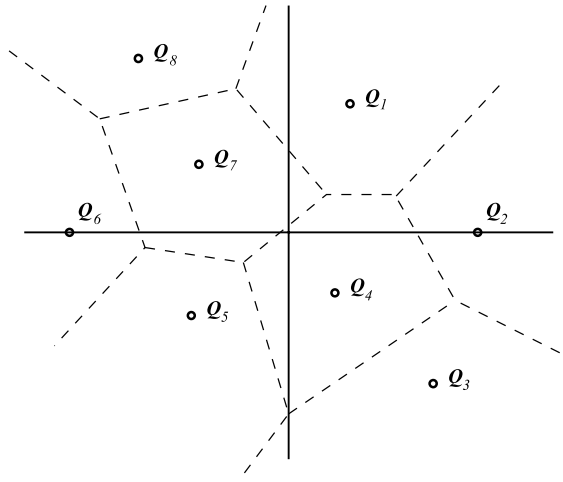


Figure 5.82. Example of partition for $K = 2$ and $N = 8$.

Rule B (Optimum codebook). Assuming the partition is $\mathcal{R}$ given, we want to find the optimum codebook $\mathcal{A}$. By minimizing (5.236) we obtain the solution

$$\mathbf{Q}_i : E[d(\mathbf{s}, \mathbf{Q}_i) \mid \mathbf{s} \in R_i] = \min_{\mathbf{Q}_j \in R_i} E[d(\mathbf{s}, \mathbf{Q}_j) \mid \mathbf{s} \in R_i] \tag{5.249}$$

In other words $\mathbf{Q}_i$ coincides with the centroid of the region R_i.

As a particular case, choosing the squared distortion (5.241), (5.237) becomes

$$\bar{d}_i \triangleq \frac{\displaystyle\int_{R_i} \|\mathbf{s} - \mathbf{Q}_i\|_2^2 \, p_{\mathbf{s}}(\mathbf{a}) \, d\mathbf{a}}{\displaystyle\int_{R_i} p_{\mathbf{s}}(\mathbf{a}) \, d\mathbf{a}} \tag{5.250}$$

and (5.249) yields

$$\mathbf{Q}_i = \frac{\displaystyle\int_{R_i} \mathbf{a} \, p_{\mathbf{s}}(\mathbf{a}) \, d\mathbf{a}}{\displaystyle\int_{R_i} p_{\mathbf{s}}(\mathbf{a}) \, d\mathbf{a}} \tag{5.251}$$

Generalized Lloyd algorithm

The generalized Lloyd algorithm, given in Figure 5.83, generates a sequence of suboptimum quantizers specified by $\{R_i\}$ and $\{\mathbf{Q}_i\}$ using the previous two rules.

1. Initialization. We choose an initial codebook $\mathcal{A}_0$ and a termination criterion based on a relative error ϵ between two successive iterations. The iteration index is denoted by j.

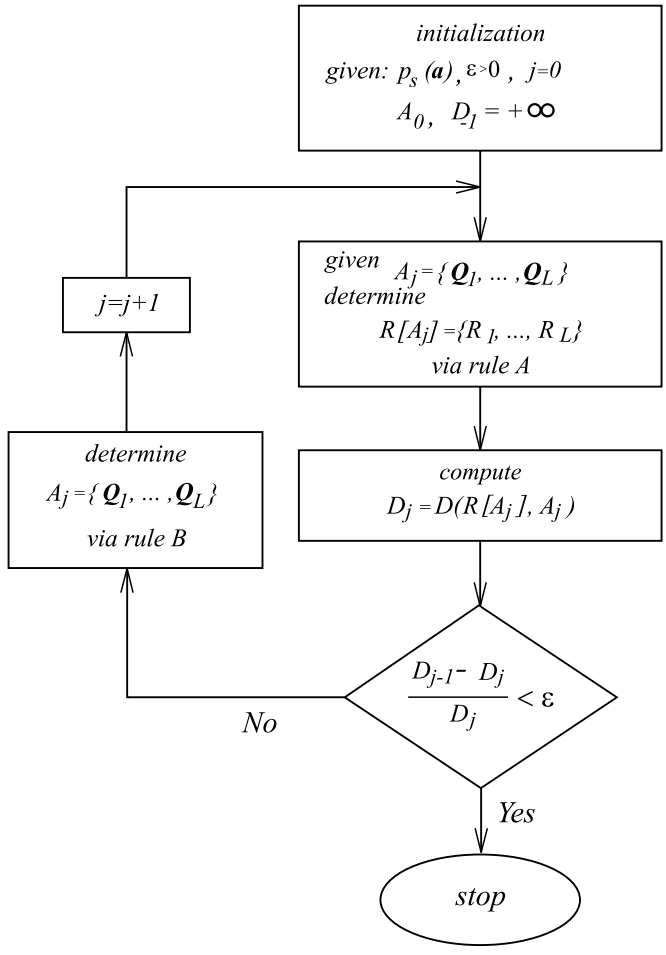


Figure 5.83. Generalized Lloyd algorithm for designing a vector quantizer.

2. Using the *rule A* we determine the optimum partition $\mathcal{R}[\mathcal{A}_j]$ using the codebook $\mathcal{A}_j$.
3. We evaluate the distortion associated with the choice of $\mathcal{A}_j$ and $\mathcal{R}[\mathcal{A}_j]$ using (5.236).
4. If

$$\frac{D_{j-1} - D_j}{D_j} < \epsilon \tag{5.252}$$

   we stop the procedure, otherwise we update the value of j.
5. Using *rule B* we evaluate the optimum codebook associated to the partition $\mathcal{R}[\mathcal{A}_{j-1}]$.
6. We go back to step 2.

The solution found is at least locally optimum; nevertheless, given that the number of locally optimum codes can be rather large, and some of the locally optimum codes may give rather poor performance, it is often advantageous to provide a good codebook to the algorithm to start with, as well as trying different initial codebooks.

The algorithm is clearly a generalization of the Lloyd algorithm given in Section 5.3.2: the only difference is that the vector version begins with a codebook (alphabet) rather than with an initial partition of the input space. However, the implementation of this algorithm is difficult for the following reasons.

- The algorithm assumes that $p_{\mathbf{s}}(\mathbf{a})$ is known. In the scalar quantization it is possible, in many applications, to develop an appropriate model of $p_s(a)$, but this becomes a more difficult problem with the increase of the number of dimensions N: in fact, the identification of the distribution type, for example, Gaussian or Laplacian, is no longer sufficient, as we also need to characterize the statistical dependence among the elements of the source vector.
- The computation of the input space partition is much harder for the VQ. In fact, whereas in the scalar quantization the partition of the real axis is completely specified by a set of $(L-1)$ points, in the two-dimensional case the partition is specified by a set of straight lines, and for the multi-dimensional case to find the optimum solution becomes very hard. For VQ with a large number of dimensions, the partition becomes also harder to describe geometrically.
- Also in the particular case (5.251), the calculation of the centroid is difficult for the VQ, because it requires evaluating a multiple integral on the region R_i.

5.8.3 LBG algorithm

An alternative approach led Linde, Buzo, and Gray [10] to consider some very long realizations of the input signal and to substitute (5.234) with (5.238) for K sufficiently large. The sequence used to design the VQ is called *training sequence* (TS) and is composed of K vectors

$$\{\mathbf{s}(m)\} \qquad m = 1, \ldots, K \tag{5.253}$$

The average distortion is now given by

$$D = \frac{1}{K} \sum_{k=1}^{K} d(\mathbf{s}(k), \mathbf{Q}[\mathbf{s}(k)]) \tag{5.254}$$

and the two rules to minimize D become:

Rule A

$$R_i = \{\mathbf{s}(k) : d(\mathbf{s}(k), \mathbf{Q}_i) = \min_{\mathbf{Q}_\ell \in \mathcal{A}} d(\mathbf{s}(k), \mathbf{Q}_\ell)\} \qquad i = 1, \ldots, L \tag{5.255}$$

that is R_i is given by all the elements $\{\mathbf{s}(k)\}$ of the TS nearest to $\mathbf{Q}_i$.

Rule B

$$\mathbf{Q}_i = \arg \min_{\mathbf{Q}_j \in \mathbb{C}^N} \frac{1}{m_i} \sum_{\mathbf{s}(k) \in R_i} d(\mathbf{s}(k), \mathbf{Q}_j) \tag{5.256}$$

where m_i is the number of elements of the TS that are inside R_i.

Using the structure of the Lloyd algorithm with the new cost function (5.254) and the two new rules (5.255) and (5.256), we arrive at the LBG algorithm. Before discussing the details, it is worthwhile pointing out some aspects of this new algorithm.

- It converges to a minimum, which is not guaranteed to be a global minimum, and generally depends on the choice of the TS.
- It does not require any stationarity assumption.
- The partition is determined without requiring the computation of expectations over $\mathbb{C}^N$.
- The computation of $\mathbf{Q}_i$ in (5.256) is still burdensome.

However, for the squared distortion (5.241) we have

$$d(\mathbf{s}(k), \mathbf{Q}_i) = \sum_{n=1}^{N} |s_n(k) - Q_{i,n}|^2 \tag{5.257}$$

and (5.256) simply becomes

Rule B

$$\mathbf{Q}_i = \frac{1}{m_i} \sum_{\mathbf{s}(k) \in R_i} \mathbf{s}(k) \tag{5.258}$$

that is $\mathbf{Q}_i$ coincides with the arithmetic mean of the TS vectors that are inside R_i.

Choice of the initial codebook

With respect to the choice of the initial codebook, the first L vectors of the TS can be used; however, if the data are highly correlated, it is necessary to use L vectors that are sufficiently spaced in time from each other.

A more effective alternative is that of taking as initial value the centroid of the TS and start with a codebook with a number of elements $L = 1$. Slightly changing the components of this code vector (*splitting procedure*), we derive two code vectors and an initial alphabet with $L = 2$; at this point, using the LBG algorithm, we determine the optimum VQ for $L = 2$. At convergence, each optimum code vector is changed to obtain two code vectors and the LBG algorithm is used for $L = 4$. Iteratively the splitting procedure and optimization is repeated until the desired number of elements for the codebook is obtained.

Let $\mathcal{A}_j = \{\mathbf{Q}_1, \ldots, \mathbf{Q}_L\}$ be the codebook at iteration j-th. The splitting procedure generates $2L$ N-dimensional vectors yielding the new codebook

$$\mathcal{A}_{j+1} = \{\mathcal{A}_j^-\} \cup \{\mathcal{A}_j^+\} \tag{5.259}$$

where

$$\mathcal{A}_j^- = \{\mathbf{Q}_i - \boldsymbol{\varepsilon}_-\} \qquad i = 1, \ldots, L\ , \tag{5.260}$$

$$\mathcal{A}_j^+ = \{\mathbf{Q}_i - \boldsymbol{\varepsilon}_+\} \qquad i = 1, \ldots, L \tag{5.261}$$

Typically $\boldsymbol{\varepsilon}_-$ is the zero vector,

$$\boldsymbol{\varepsilon}_- = \mathbf{0} \tag{5.262}$$

and

$$\boldsymbol{\varepsilon}_+ = \frac{1}{10}\sqrt{\frac{\mathtt{M}_s}{N}} \cdot \mathbf{1} \tag{5.263}$$

so that

$$\|\boldsymbol{\varepsilon}_+\|_2^2 \leq 0.01\ \mathtt{M}_s \tag{5.264}$$

where $\mathtt{M}_s$ is the power of the TS.

Description of the LBG algorithm with splitting procedure

Choosing $\epsilon > 0$ (typically $\epsilon = 10^{-3}$) and an initial alphabet given by the splitting procedure applied to the average of the TS, we obtain the LBG algorithm, whose block diagram is shown in Figure 5.84, whereas its operations are depicted in Figure 5.85.

Selection of the training sequence

A rather important problem associated with the use of a TS is that of empty cells. It is in fact possible that some regions R_i contain few or no elements of the TS: in this case the code vectors associated with these regions contribute little or nothing at all to the reduction of the total distortion.

Possible causes of this phenomenon are:

- TS too short: the training sequence must be sufficiently long, so that every region R_i contains at least 30-40 vectors;
- poor choice of the initial alphabet: in this case, in addition to the obvious solution of modifying this choice, we can limit the problem through the following splitting procedure.

Let R_i be a region that contains $m_i < m_{min}$ elements. We eliminate the code vector $\mathbf{Q}_i$ from the codebook and apply the splitting procedure limited only to the region that gives

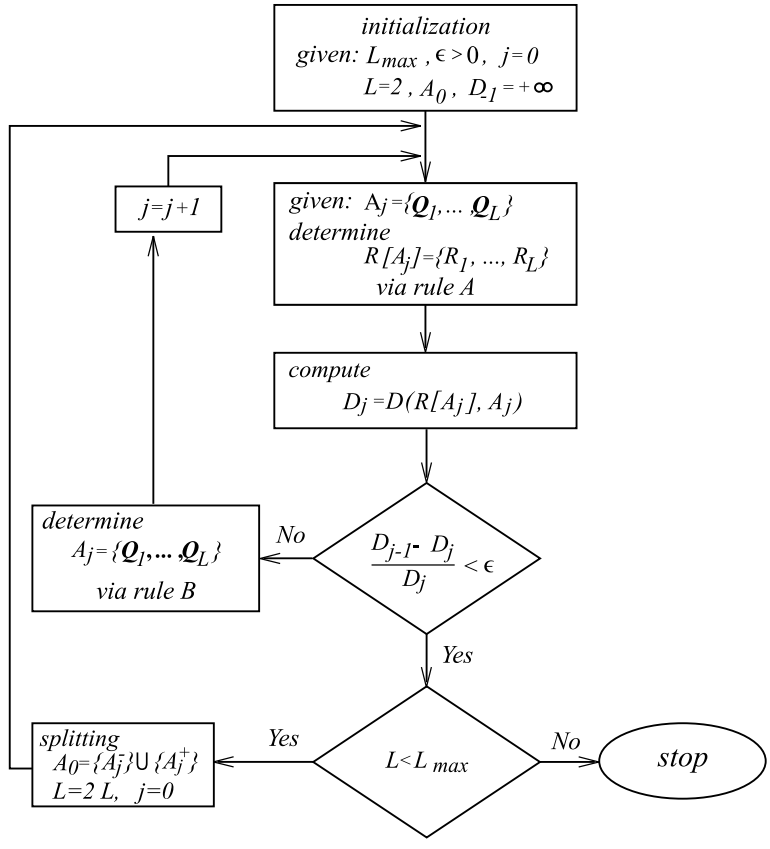


Figure 5.84. LBG algorithm with splitting procedure.

the largest contribution to the distortion; then we compute the new partition and proceed in the usual way.

We give some practical rules, taken from [3] for LPC applications,[13] that can be useful in the design of a vector quantizer.

- If $K/L \leq 30$, there is a possibility of empty regions, where we recall K is the number of vectors of the TS, and L is the number of code vectors.
- If $K/L \leq 600$, an appreciable difference between the distortion calculated with the TS and that calculated with a new sequence may exist.

In the latter situation, it may in fact happen that, for a very short TS, the distortion computed for vectors of the TS is very small; the extreme case is obtained by setting $K = L$, hence $D = 0$. In this situation, for a sequence different from the TS (*outside* TS) the distortion is

[13] These rules were derived in the VQ of LPC vectors. They can be considered valid in the case of strongly correlated vectors.

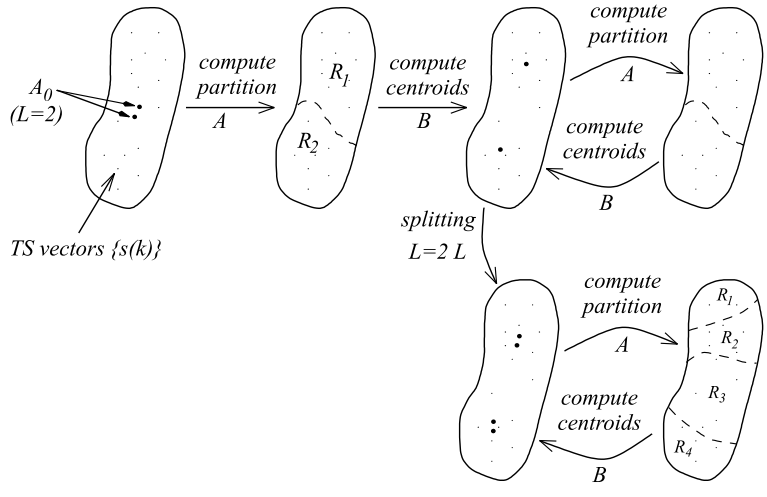


Figure 5.85. Operations of the LBG algorithm with splitting procedure.

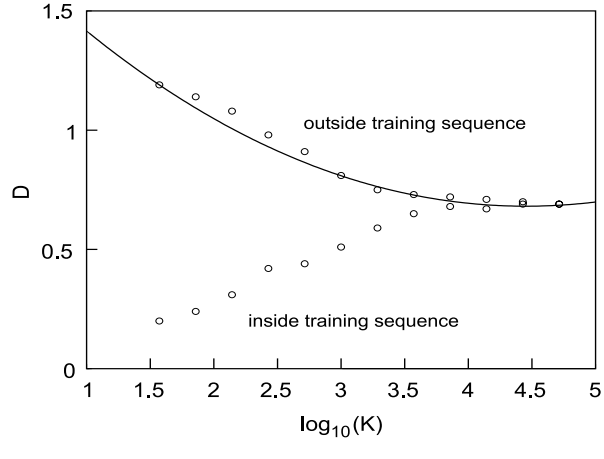


Figure 5.86. Values of the distortion as a function of the number of vectors K in the *inside* and *outside* training sequences.

in general very high. As illustrated in Figure 5.86, only if K is large enough, does the TS adequately represent the input process and no substantial difference appears between the distortion measured with vectors inside or outside TS [10].[14]

Finally we find that the LBG algorithm, even though very simple, requires numerous computations. We consider, for example, as vector source the LPC coefficients with $N = 10$, computed over windows of duration equal to 20 ms of a speech signal sampled at 8 kHz. Taking $L = 256$ we have a rate $R_b = 8$ bit/20 ms equal to 400 bit/s. As a matter of fact, the

[14] This situation is similar to that obtained by the LS method (see Section 3.2).

LBG algorithm requires a minimum $K = 600 \cdot 256 \approxeq 155000$ vectors for the TS, which roughly corresponds to three minutes of speech.

5.8.4 Variants of VQ

Tree search VQ

A random VQ, determined according to the LBG algorithm, requires:

- a large memory to store the codebook;
- a large computational complexity to evaluate the L distances for encoding.

A variant of VQ that requires a lower computational complexity, at the expense of a larger memory, is the tree search VQ. As illustrated in Figure 5.87, whereas in the memoryless VQ case the comparison of the input vector $\mathbf{s}$ must occur with all the elements of the codebook, thus determining a full search, in the tree search VQ we proceed by levels: first, we compare $\mathbf{s}$ with $\mathbf{Q}_{A1}$ and $\mathbf{Q}_{A2}$, then we proceed along the branch whose node has a representative vector "closest" to $\mathbf{s}$.

To determine the code vectors at different nodes, for a binary tree the procedure consists of the following steps.

1. Calculate the optimum quantizer for the first level by the LBG algorithm; the codebook contains 2 code vectors.
2. Divide the training sequence into subsequences relative to every node of level n ($n = 2, 3, \ldots, N_{LEV}$, $N_{LEV} = \log_2 L$); in other words, collect all vectors that are associated with the same code vector.
3. Apply the LGB algorithm to every subsequence to calculate the codebook of level n.

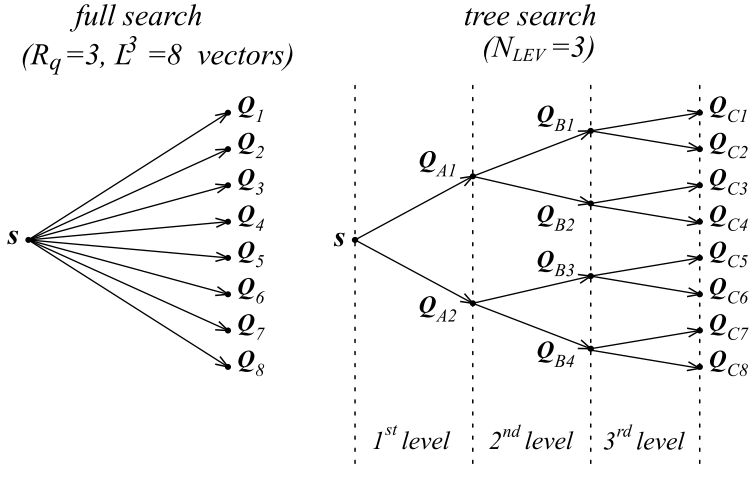


Figure 5.87. Comparison between full search and tree search.

Table 5.15 Comparison between full search and tree search.

	Computation of $d(\cdot,\cdot)$	Number of vectors to memorize
full search	2^{R_q}	2^{R_q}
tree search	$2R_q$	$\sum_{i=1}^{R_q} 2^i \simeq 2^{R_q+1}$
for $R_q = 10$ (bit/vector)	Computation of $d(\cdot,\cdot)$	Number of vectors to memorize
full search	1024	1024
tree search	20	2046

As an example, the memory requirements and the number of computations of $d(\cdot,\cdot)$ are shown in Table 5.15 for a given value of R_q (bit/vector) in the cases of full search and tree search. Although the performance is slightly lower, the computational complexity of the encoding scheme for a tree search is considerably reduced.

Multistage VQ

The multistage VQ technique presents the advantage of reducing both the encoder computational complexity and the memory required. The idea consists in dividing the encoding procedure into successive stages, where the first stage performs quantization with a codebook with a reduced number of elements. Successively, the second stage performs quantization of the error vector $\mathbf{e} = \mathbf{s} - \mathbf{Q}[\mathbf{s}]$: the quantized error gives a more accurate representation of the input vector. A third stage could be used to quantize the error of the second stage and so on.

We compare the complexity of a one-stage scheme with that of a two-stage scheme, illustrated in Figure 5.88. Let $R_q = \log_2 L$ be the rate in bit/vector for both systems and assume that all the code vectors have the same dimension $N = N_1 = N_2$.

- Two-stage:
  $R_q = \log_2 L_1 + \log_2 L_2$, hence $L_1 L_2 = L$.
  Computations of $d(\cdot,\cdot)$ for encoding: $L_1 + L_2$.
  Memory: $L_1 + L_2$ locations.

- One-stage:
  $R_q = \log_2 L$.
  Computations of $d(\cdot,\cdot)$ for encoding: L.
  Memory: L locations.

The advantage of a multistage approach in terms of cost of implementation is evident, however, it has lower performance than a one-stage VQ.

Product code VQ

The input vector is split into subvectors that are quantized independently, as illustrated in Figure 5.89.

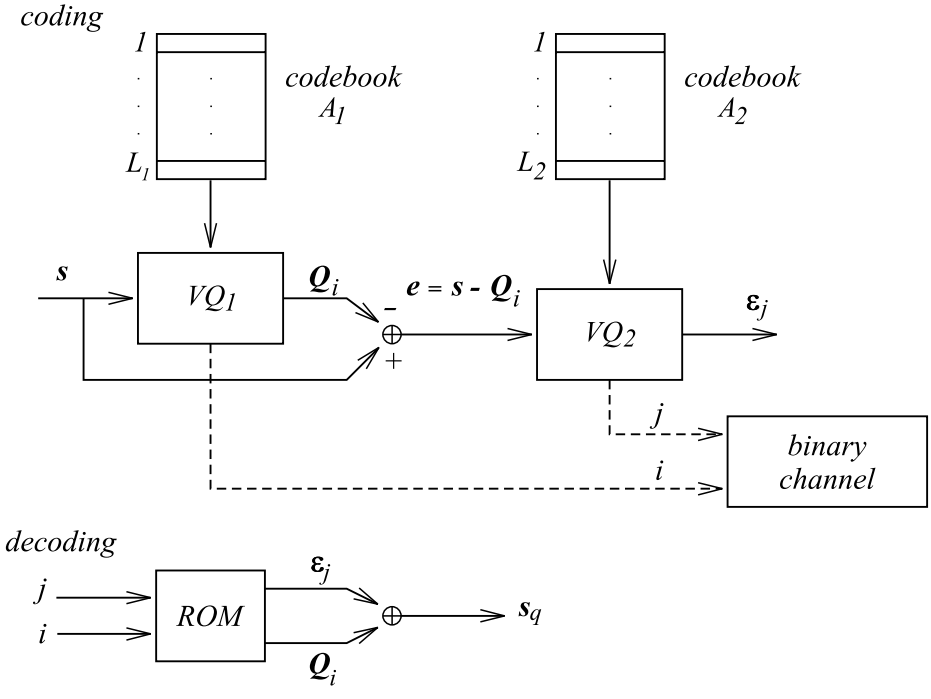


Figure 5.88. Multistage (two-stage) VQ.

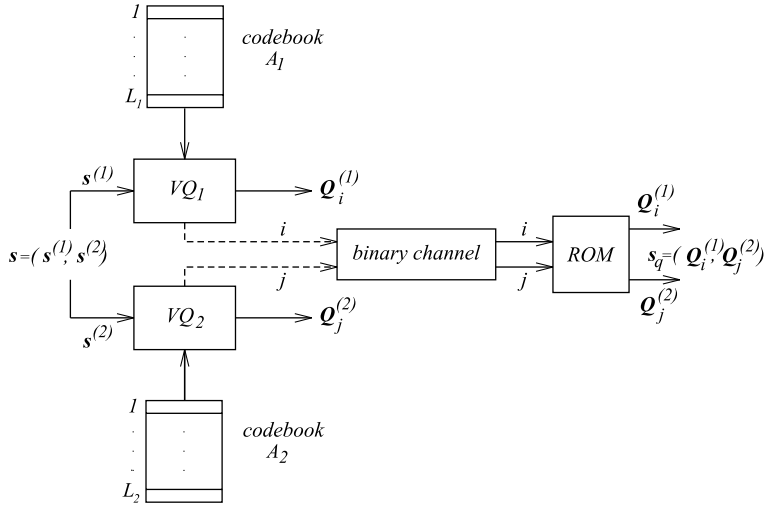


Figure 5.89. Product code VQ.

This technique is useful if a) there are input vector components that can be encoded separately because of their different effects, e.g., prediction gain and LPC coefficients, or b) the input vector has too large a dimension to be encoded directly.

It presents the disadvantage that it does not consider the correlation that may exist between the various subvectors, that could bring about a greater coding efficiency.

A more general approach is the *sequential search product code*, [7], where the quantization of the subvector n depends also on the quantization of previous subvectors.

With reference to Figure 5.89, assuming $L = L_1 L_2$ and $N = N_1 + N_2$, we note that the rate per dimension for the VQ is given by

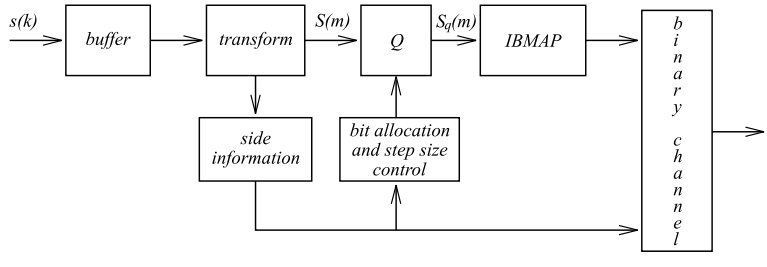
$$R_q = \frac{\log_2 L}{N} = \frac{\log_2 L_1}{N} + \frac{\log_2 L_2}{N} \tag{5.265}$$

whereas for the product code VQ it is given by

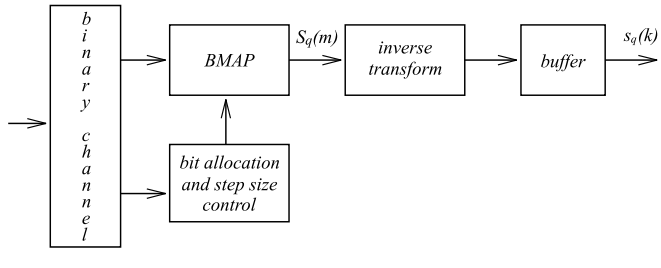
$$R_q = \frac{\log_2 L_1}{N_1} + \frac{\log_2 N_2}{N_2} \tag{5.266}$$

5.9 Other coding techniques

We briefly discuss two other coding techniques along with the perceptive aspects related to the hearing apparatus. For further details we refer the reader to [3, 6].



(a) Encoder



(b) Decoder

Figure 5.90. Block diagram of the ATC.

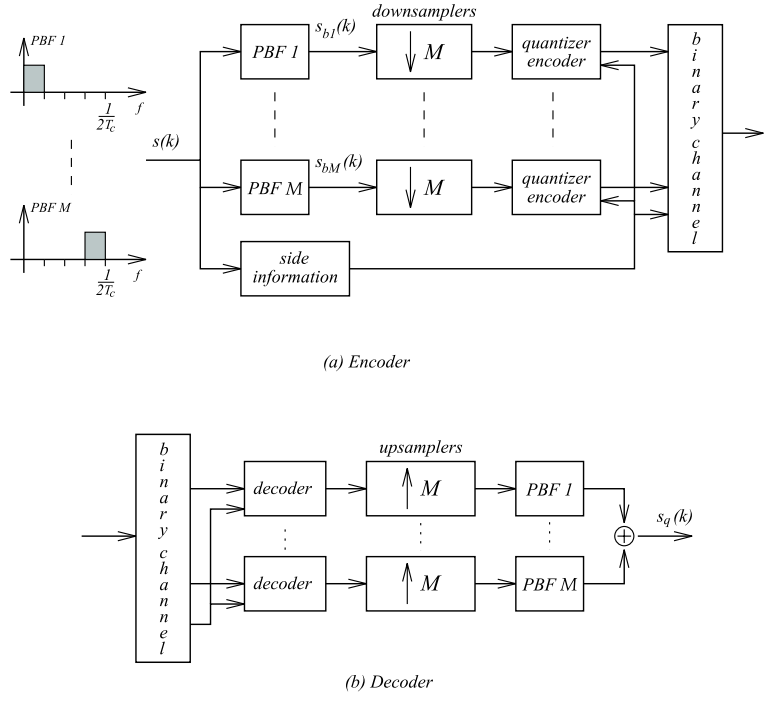


Figure 5.91. Block diagram of the SBC.

Adaptive transform coding (ATC)

The ATC takes advantage of the non-uniform energy distribution of a signal in some transformed domain, using for example the DFT or the DCT (see Sections 3.5.3 and 3.5.4). The basic scheme is illustrated in Figure 5.90 and includes a quantizer that adapts to the different inputs $\{S(m)\}$.

Sub-band coding (SBC)

The SBC exploits the same principle as the ATC, but it operates in the time domain by using a filter bank (see Figure 5.91).

5.10 Source coding

We briefly mention an important topic, namely *source coding*, which is used to "compress" digital information messages. In fact, a discrete-time, discrete-valued source signal can be encoded with a lower average bit rate by means of entropy coding [4], which assigns code words of variable lengths to possible input patterns, i.e. to highly probable input patterns are assigned shorter code words and vice versa. We cite the Lempel–Ziv algorithm as one of the most common source coding algorithms.

5.11 Speech and audio standards

We conclude this chapter by giving in Table 5.16 a partial list of the various standards to code audio and speech signals [11]. The first nine are for narrowband speech applications (see Table 5.1). It is interesting to observe the various standards that adopt CELP coding, listed in Table 5.17: we notice that most of them are for cellular radio applications. It is also interesting to compare the various standards to code video signals given in Table 5.18 with those of Table 5.16 for speech and audio.

Table 5.16 Main standards for audio and speech coding.

	Standard	Description
1	G.711	PCM at 64 kbit/s
2	G.721	ADPCM at 32 kbit/s
3	G.723	ADPCM at 24 and 40 kbit/s
4	G.726	G.723+G.721
5	G.727	embedded ADPCM at 40, 32, 24 and 16 kbit/s ("embedded" means that a code also includes those of lower rate)
6	G.728	LD-CELP at 16 kbit/s (LD stands for *low delay*)
7	G.729	CS-ACELP at 8 kbit/s
8	G.729 Annex A	CS-ACELP at 8 kbit/s with reduced complexity
9	G.723.1	MPC-MLQ at 5.3 and 6.4 kbit/s
10	G.722	SBC+ADPCM for *wide band speech* at 64, 56 and 48 kbit/s . A SBC scheme having two bands, $0 \div 4$ kHz and $4 \div 8$ kHz is used; in each band there is a G.721 encoder. Bit allocation in the two bands is dynamic, for example, 5+3 or 6+2
11	IS-54 (TIA)	VSELP at 7.95 kbit/s (VSELP stands for *vector sum excited linear prediction*)
12	FS-1015 (LPC-10E)	LPC at 2.4 kbit/s
13	FS-1016	CELP at 4.8 kbit/s
14	GSM-FR	RPE-LTP at 13 kbit/s (LTP stands for *long-term prediction*); there is also a 5.6 kbit/s version
15	MPEG1, Layer I	SBC at 192 kbit/s per audio channel (stereo) [generally $32 \div 448$ kbit/s total]
16	MPEG1, Layer II	SBC at 128 kbit/s per audio channel [generally $32 \div 384$ kbit/s total]
17	MPEG1, Layer III	SBC+MDCT+Huffman coding at 96 kbit/s per audio channel [generally $32 \div 320$ kbit/s total]
18	MPEG2, AAC	SBC+MDCT coding at 64 kbit/s per audio channel

Table 5.17 Main standards based on CELP.

Body	Abbreviations	Bit rate
ITU	G.723	5.27 and 6.3 kbit/s; coding of audio in multimedia systems
ITU	G.728	16 kbit/s
ITU	G.729	8 kbit/s; encoding of speech and data
TIA	IS-54	7.95 kbit/s; full rate for North America cellular systems based on D-AMPS
TIA	IS-95	1.2, 2.4, 4.8 and 9.6 kbit/s; coding for North America cellular systems based on CDMA
TIA/ETSI	US-1	12.2 kbit/s; enhanced full rate for GSM
ETSI	GSM-HR	5.6 kbit/s; half rate for GSM
U.S. (DoD)	FS 1016	4.8 kbit/s
U.S. (DoD)	MELP	2.4 kbit/s

Table 5.18 Bit rates for video standards.

Application	Target bit rate
ISDN Video Telephone	64÷128 kbit/s
ISDN Video Conferencing	128 kbit/s
MPEG1 CD-Rom Video	1.5 Mbit/s
MPEG2 TV (Broadcast Quality)	6 Mbit/s
HDTV (Broadcast Quality)	24 Mbit/s
TV (Studio Quality, Compressed)	34 Mbit/s
HDTV (Studio Quality, Compressed)	140 Mbit/s
TV (Studio Quality, Uncompressed)	216 Mbit/s
HDTV (Studio Quality, Uncompressed)	1 Gbit/s

Bibliography

[1] L. R. Rabiner and R. W. Schafer, *Digital processing of speech signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.

[2] *IEEE Signal Processing Magazine*, Sept. 1997. vol. 14.

[3] D. Sereno and P. Valocchi, *Codifica numerica del segnale audio*. L'Aquila: Scuola Superiore G. Reiss Romoli, 1996.

[4] N. S. Jayant and P. Noll, *Digital coding of waveforms*. Englewood Cliffs, NJ: Prentice-Hall, 1984.

[5] B. S. Atal and J. R. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates", in *Proc. ICASSP*, pp. 614–617, 1982.

[6] B. S. Atal, V. Cuperman, and A. Gersho, eds, *Advances in speech coding*. Boston, MA: Kluwer Academic Publishers, 1991.

[7] A. Gersho and R. M. Gray, *Vector quantization and signal compression*. Boston, MA: Kluwer Academic Publishers, 1992.

[8] R. M. Gray, "Vector quantization", *IEEE ASSP Magazine*, vol. 1, pp. 4–29, Apr. 1984.

[9] T. D. Lookabaugh and R. M. Gray, "High–resolution quantization theory and the vector quantizer advantage", *IEEE Trans. on Information Theory*, vol. 35, pp. 1020–1033, Sept. 1989.

[10] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design", *IEEE Trans. on Communications*, vol. 28, pp. 84–95, Jan. 1980.

[11] *IEEE Communication Magazine*, Sept. 1997. vol. 35.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 6

Modulation theory

The term *modulation* indicates the process of translating the information generated by a source into a signal that is suitable for transmission over a physical channel. In the case of digital transmission, the information is represented by a sequence of binary data (bits) generated by the source, or by a digital encoder of analog signals (see Chapter 5). The mapping of a digital sequence to a signal is called *digital modulation*, and the device that performs the mapping is called *digital modulator*. A modulator may employ a set of $M = 2$ waveforms to generate a signal (*binary modulation*), or, in general, $M \geq 2$ waveforms (*M-ary modulation*). The transmission medium determines the channel characteristics, as discussed in Chapter 4; it may consist for example of a twisted-pair cable, a coaxial cable, an optical fiber, an infrared link, a radio link, or a combination of them; in any case the channel modifies the transmitted waveform by introducing for example distortion, interference, and noise. In this chapter we assume that the channel introduces only additive white Gaussian noise (AWGN). We postpone the study of other effects to the next chapters; only in Section 6.12 we will give some simple results for a channel affected by *flat fading*.

The task of the receiver is to detect which signal was transmitted, based on the received signal. Using the vector representation of signals discussed in Section 1.2, in this chapter we will introduce the optimum receiver, referring to the *detection theory*, and present a survey of the main modulation techniques, e.g., PAM, PSK, QAM, orthogonal, and biorthogonal. The performance of each modulation-demodulation method is evaluated with reference to the *bit error probability*, and a comparison of the various methods is given in terms of spectral efficiency and required transmission bandwidth. The transmission rates achievable by the various modulation methods over a specific channel for a given target error probability are then compared with the Shannon bound, which indicates the maximum rate that can be achieved for reliable transmission.

6.1 Theory of optimum detection

We consider first the transmission of an isolated pulse.

With reference to the system illustrated in Figure 6.1, the transmitter generates randomly one of M real-valued waveforms $s_n(t)$, $n = 1, \ldots, M$, and sends it over the channel; the waveform is corrupted by real-valued additive white Gaussian noise w having zero mean and spectral density $N_0/2$. The variable a_0 in Figure 6.1 is modeled as a discrete r.v.

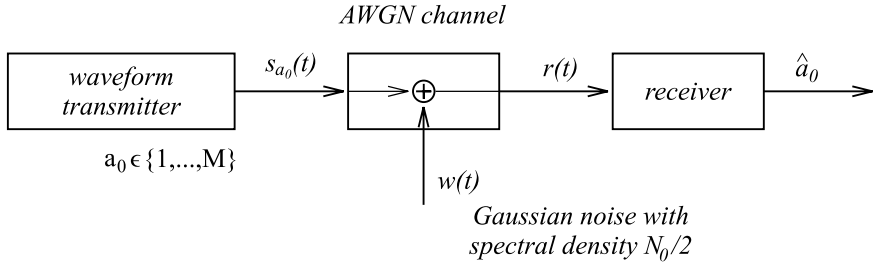


Figure 6.1. Model of the transmission system.

with values in $\{1, \ldots, M\}$, and represents the index, or symbol, of the transmitted signal. Assuming that the waveform with index m is transmitted, that is $a_0 = m$, the received, or observed, signal is given by

$$r(t) = s_m(t) + w(t) \tag{6.1}$$

The receiver, based on $r(t)$, must decide which among the M hypotheses

$$H_n : r(t) = s_n(t) + w(t) \qquad n = 1, 2, \ldots, M \tag{6.2}$$

is the most probable, and correspondingly must select the detected value $\hat{a}_0$ [1]. The theory exposed in this section can be immediately extended to the case of complex-valued signals.

We represent time-continuous signals using the vector notation introduced in Section 1.2. Let $\{\phi_i(t)\}$, $i = 1, \ldots, I$, be a complete basis for the M signals $\{s_m(t)\}$, $m = 1, \ldots, M$; let $\mathbf{s}_m$ be the vector representation of $s_m(t)$, $t \in \Re$,

$$s_m(t) \longleftrightarrow \mathbf{s}_m = [s_{m1}, \ldots, s_{mI}]^T \tag{6.3}$$

where

$$s_{mi} = \langle s_m, \phi_i \rangle = \int_{-\infty}^{+\infty} s_m(t)\phi_i^*(t)\, dt \qquad m = 1, \ldots, M \quad i = 1, \ldots, I \tag{6.4}$$

Recall that the set $\{\mathbf{s}_m\}$, $m = 1, \ldots, M$, is also called the system constellation.

The basis of I functions may be *incomplete* for the representation of the noise signal w. In any case, we express the noise as

$$w(t) = w_\phi(t) + w_\perp(t) \tag{6.5}$$

where

$$w_\phi(t) = \sum_{i=1}^{I} w_i \phi_i(t) \tag{6.6}$$

and where

$$w_i = \langle w, \phi_i \rangle = \int_{-\infty}^{+\infty} w(t)\phi_i^*(t)\, dt \tag{6.7}$$

In other words, w_ϕ is the component of w that lies in the space spanned by the basis $\{\phi_i(t)\}$, $t \in \Re$, $i = 1, \ldots, I$, whereas $w_\perp$ is the error due to this representation. Since

$$w_\perp(t) = w(t) - w_\phi(t) \tag{6.8}$$

is orthogonal to $\{\phi_i(t)\}$, $i = 1, \ldots, I$, and hence also to w_ϕ, we can say that $w_\perp$ lies outside of the desired signal space and, as we will state later using the *theorem of irrelevance*, it can be ignored because it is irrelevant for the detection. The vector representation of the component of the noise signal that lies in the span of $\{\phi_i(t)\}$ is given by

$$\mathbf{w} = [w_1, \ldots, w_I]^T = \mathbf{w}_\phi \tag{6.9}$$

Statistics of the random variables $\{w_i\}$

1. $\{w_i\}$, $i = 1, \ldots, I$, are jointly Gaussian random variables, as they are linear transformations of a Gaussian process (see (6.7)).
2. *Mean*:

$$E[w_i] = \int E[w(t)]\phi_i^*(t)\,dt = 0 \qquad i = 1, \ldots, I \tag{6.10}$$

as w has zero mean.

3. *Correlation*: as $w(t)$ is white noise, we have

$$E[w(t_1)w^*(t_2)] = \frac{N_0}{2}\delta(t_1 - t_2) \tag{6.11}$$

and

$$\begin{aligned} E[w_i w_j^*] &= \iint E[w(t_1)w^*(t_2)]\phi_i^*(t_1)\phi_j(t_2)\,dt_1\,dt_2 \\ &= \frac{N_0}{2}\iint \delta(t_1 - t_2)\phi_i^*(t_1)\phi_j(t_2)\,dt_1\,dt_2 \\ &= \frac{N_0}{2}\int \phi_i^*(t)\phi_j(t)\,dt \\ &= \frac{N_0}{2}\delta_{i-j} \qquad i, j = 1, \ldots, I \end{aligned} \tag{6.12}$$

because of the orthogonality of the basis $\{\phi_i(t)\}$.

Hence the components $\{w_i\}$ are uncorrelated. As $\{w_i\}$, $i = 1, \ldots, I$, are jointly Gaussian uncorrelated random variables with zero mean, then they are statistically independent with equal variance given by

$$\sigma_I^2 = \frac{N_0}{2} \tag{6.13}$$

Sufficient statistics

Defining

$$\mathbf{r} = [r_1, \ldots, r_I]^T \qquad \text{with } r_i = \langle r, \phi_i \rangle \tag{6.14}$$

the components of the vector $\mathbf{r}$ are called *sufficient statistics*[1] to decide among the M hypotheses. Therefore we get the formulation equivalent to (6.2),

$$H_n : \mathbf{r} = \mathbf{s}_n + \mathbf{w} \qquad n = 1, \ldots, M \tag{6.15}$$

From the above results, the probability density function of $\mathbf{r}$, under the hypothesis that waveform n is transmitted, is given by:[2]

$$p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n) = \frac{1}{\left(\sqrt{2\pi \dfrac{N_0}{2}}\right)^I} \exp\left(-\frac{1}{N_0}\|\boldsymbol{\rho} - \mathbf{s}_n\|^2\right) \qquad \boldsymbol{\rho} \in \Re^I \tag{6.16}$$

Decision criterion

We subdivide the space $\Re^I$ of the received signal $\mathbf{r}$ into M non-overlapping regions R_n ($\bigcup_{n=1}^{M} R_n = \Re^I$ and $R_n \cap R_m = \emptyset$ for $n \neq m$). Then we adopt the following decision rule:

$$\text{choose } H_n \text{ (and } \hat{a}_0 = n) \qquad \text{if } \mathbf{r} \in R_n \tag{6.17}$$

The choice of M regions is made so that the probability of a correct decision is maximum. Let $p_n = P[a_0 = n]$ be the transmission probability of the waveform n, or *a priori probability*. Recalling the total probability theorem, the probability of correct decision is given by

$$\begin{aligned} P[\mathbb{C}] &= P[\hat{a}_0 = a_0] \\ &= \sum_{n=1}^{M} P[\hat{a}_0 = n \mid a_0 = n] P[a_0 = n] \\ &= \sum_{n=1}^{M} p_n P[\mathbf{r} \in R_n \mid a_0 = n] \\ &= \sum_{n=1}^{M} \int_{R_n} p_n p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n)\, d\boldsymbol{\rho} \end{aligned} \tag{6.18}$$

---

[1] Given a desired signal corrupted by noise, in general the notion of *sufficient statistics* applies to any signal, or sequence of samples, that allows the optimum detection of the desired signal. In other words, no information is lost in considering a set of sufficient statistics instead of the received signal.

A particular case is represented by transformations that allow reconstruction of a signal using the basis identified by the desired signal. For example, considering the basis $\{e^{\pm j2\pi ft}, t \in \Re, f \in \mathcal{B}\}$ to represent a real-valued signal with passband $\mathcal{B}$ in the presence of additive noise, that is the Fourier transform of the noisy signal filtered by an ideal filter with passband $\mathcal{B}$, we are able to reconstruct the noisy signal within the passband of the desired signal; therefore the noisy signal filtered by a filter with passband $\mathcal{B}$ is a sufficient statistic.

[2] Here we use the formulation (1.377) for real-valued signals; we would get the same results using the formulation (1.380) for complex-valued signals.

We define the indicator function of the set R_n as

$$\mathcal{I}_n = \begin{cases} 1 & \boldsymbol{\rho} \in R_n \\ 0 & \text{elsewhere} \end{cases} \tag{6.19}$$

Then (6.18) becomes

$$P[\mathbb{C}] = \int_{\Re^I} \sum_{n=1}^{M} \mathcal{I}_n \, p_n \, p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n) \, d\boldsymbol{\rho} \tag{6.20}$$

The integrand function consists of M terms but, being the M regions non-overlapping, for each value of $\boldsymbol{\rho}$ only one of the terms is different from zero. Therefore the maximum value of the integrand function for each value of $\boldsymbol{\rho}$, and hence of the integral, is achieved if for each value of $\boldsymbol{\rho}$ we select among M terms the term that yields the maximum value of $p_n \, p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n)$. Thus we have the following decision criterion.[3]

Maximum a posteriori probability (MAP) criterion:

$$\boldsymbol{\rho} \in R_m \qquad \hat{a}_0 = m \qquad \text{if } m = \arg\max_n \; p_n \, p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n) \tag{6.22}$$

Using the Bayes' rule

$$p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n) = p_{\mathbf{r}}(\boldsymbol{\rho}) \, \frac{P[a_0 = n \mid \mathbf{r} = \boldsymbol{\rho}]}{p_n} \tag{6.23}$$

the decision criterion becomes

$$\hat{a}_0 = \arg\max_n \; P[a_0 = n \mid \mathbf{r} = \boldsymbol{\rho}] \tag{6.24}$$

In other words, given that we observe $\boldsymbol{\rho}$, the signal detected by (6.24) has the largest probability of having been transmitted. The probabilities $P[a_0 = n \mid \mathbf{r} = \boldsymbol{\rho}\,], \; n = 1, \ldots, M$, are the *a posteriori probabilities*.

We give a simple example of application of the MAP criterion for $I = 1$ and $M = 3$. Let the function $p_n \, p_{r|a_0}(\rho \mid n)$, $n = 1, 2, 3$, be given as shown in Figure 6.2. If we indicate with τ_1, τ_2, and τ_3 the intersection points of the various functions as illustrated in Figure 6.2, it is easy to verify that

$$R_1 = (-\infty, \tau_1] \qquad R_2 = (\tau_1, \tau_2] \cup (\tau_3, +\infty) \qquad R_3 = (\tau_2, \tau_3] \tag{6.25}$$

---

[3] arg means *argument*; for a function $f(x, n)$,

$$m = \arg\max_n f(x, n) \tag{6.21}$$

denotes the value of m that coincides with the value of n for which the function $f(x, n)$ is maximum for a given x. If two or more values of n that maximize $f(x, n)$ exist, a random choice is made to determine m. For a complex number c, $\arg(c)$ denotes the phase of c.

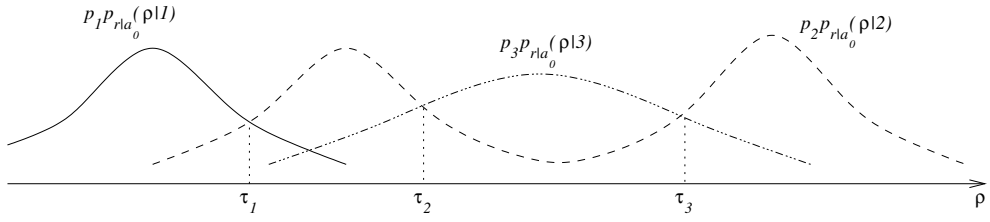


Figure 6.2. Illustration of the MAP criterion.

Maximum likelihood (ML) criterion. If the signals are equally likely *a priori*, i.e. $p_n = 1/M$, $\forall n$, the criterion (6.22) becomes

$$\boldsymbol{\rho} \in R_m \qquad \hat{a}_0 = m \qquad \text{if} \quad m = \arg\max_n \; p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n) \tag{6.26}$$

The ML criterion leads to choosing that value of n for which the conditional probability that $\mathbf{r} = \boldsymbol{\rho}$ is observed given $a_0 = n$ is maximum.

In some texts the ML criterion is formulated via the definition of the *likelihood ratios*:

$$\mathtt{L}_n(\boldsymbol{\rho}) = \frac{p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n)}{p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid 1)} \qquad n = 1, 2, \ldots, M \tag{6.27}$$

In this case the ML criterion becomes

$$\hat{a}_0 = m \qquad \text{if} \quad m = \arg\max_n \; \mathtt{L}_n(\boldsymbol{\rho}) \tag{6.28}$$

From (6.26), observing (6.16) we get

$$\hat{a}_0 = \arg\max_n \exp\left(-\frac{1}{N_0}\|\boldsymbol{\rho} - \mathbf{s}_n\|^2\right) \tag{6.29}$$

Taking the logarithm, which is a monotonic function, we obtain

$$\hat{a}_0 = \arg\min_n \|\boldsymbol{\rho} - \mathbf{s}_n\|^2 \tag{6.30}$$

Hence the ML criterion coincides with the *minimum distance criterion*: "decide for the signal vector $\mathbf{s}_m$, which is closest to the received signal vector $\boldsymbol{\rho}$". Moreover, the decision regions $\{R_n\}$, $n = 1, \ldots, M$, are easily determined.

An example is given in Figure 6.3 for the three signals of Example 1.2.2 on page 10. Considering a pair of vectors $\mathbf{s}_i, \mathbf{s}_j$, we draw the straight line of points that are equidistant from $\mathbf{s}_i$ and $\mathbf{s}_j$: this straight line defines the boundary between R_i and R_j. The procedure then is repeated for every pair of vectors. The decision region associated with each vector $\mathbf{s}_n$ is given by the intersection of two half-planes as illustrated in Figure 6.3.

Theorem of irrelevance

With regard to the decision process, we introduce a theorem that formalizes the distinction previously mentioned between relevant and irrelevant components of the received signal.

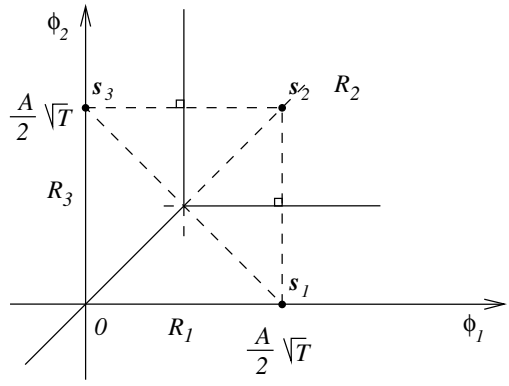


Figure 6.3. Construction of decision regions for the constellation of the Example 1.2.2.

Let us assume that the signal vector $\mathbf{r}$ can be split into two parts, $\mathbf{r} = [\mathbf{r}_1, \mathbf{r}_2]$. Then, under the hypothesis $a_0 = n$,

$$p_{\mathbf{r}|a_0}(\boldsymbol{\rho} \mid n) = p_{\mathbf{r}_1,\mathbf{r}_2|a_0}(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2 \mid n) \tag{6.31}$$

which, recalling the definition of conditional probability, can be rewritten as

$$p_{\mathbf{r}_1,\mathbf{r}_2|a_0}(\boldsymbol{\rho}_1, \boldsymbol{\rho}_2 \mid n) = p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n) p_{\mathbf{r}_1|a_0}(\boldsymbol{\rho}_1 \mid n) \tag{6.32}$$

Substitution of (6.32) into (6.22) leads to the following result.

Theorem 6.1
If $p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n)$ does *not depend* on the particular value n assumed by a_0, that is if

$$p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n) = p_{\mathbf{r}_2|\mathbf{r}_1}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1) \tag{6.33}$$

then *the optimum receiver can disregard the component* $\mathbf{r}_2$ *and base its decision only on the component* $\mathbf{r}_1$.

Corollary 6.1
A *sufficient* condition to disregard $\mathbf{r}_2$ is that

$$p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n) = p_{\mathbf{r}_2}(\boldsymbol{\rho}_2) \tag{6.34}$$

We illustrate the theorem by the following examples.

Example 6.1.1
The system (6.2) is represented using a larger basis, as illustrated in Figure 6.4, where the noise (6.5) has two components

$$\mathbf{w}_1 = \mathbf{w}_\phi \qquad \mathbf{w}_2 = \mathbf{w}_\perp \tag{6.35}$$

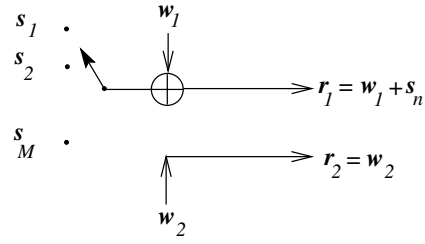


Figure 6.4. Example 6.1.1: the vector $\mathbf{r}_2$ is irrelevant.

We note that the received signal vector $\mathbf{r}_2$ coincides with the noise vector $\mathbf{w}_2$ that is statistically independent of $\mathbf{w}_1$ and $\mathbf{s}_n$. Therefore we have

$$p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n) = p_{\mathbf{r}_2}(\boldsymbol{\rho}_2) \tag{6.36}$$

hence, by Corollary 6.1, the component $\mathbf{r}_2 = \mathbf{w}_\perp$ can be disregarded by the optimum receiver.

Example 6.1.2
In the system shown in Figure 6.5, the noise vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ are statistically independent. As $\mathbf{r}_2 = \mathbf{r}_1 + \mathbf{w}_2$, if $\mathbf{r}_1$ is known, then $\mathbf{r}_2$ depends only on the noise $\mathbf{w}_2$, that is independent of the particular $\mathbf{s}_n$ transmitted. Then

$$p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n) = p_{\mathbf{w}_2}(\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1) \tag{6.37}$$

does not depend on n: therefore (6.33) is verified and $\mathbf{r}_2$ is irrelevant.

Example 6.1.3
As in the previous example, the noise vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ in Figure 6.6 are statistically independent. Under this condition, however, $\mathbf{r}_2$ *cannot* be disregarded by the optimum receiver: in fact, from $\mathbf{r}_2 = \mathbf{w}_2 + \mathbf{w}_1$ and $\mathbf{w}_1 = \mathbf{r}_1 - \mathbf{s}_n$, we get

$$p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n) = p_{\mathbf{w}_2}(\boldsymbol{\rho}_2 - \mathbf{w}_1) = p_{\mathbf{w}_2}(\boldsymbol{\rho}_2 - \boldsymbol{\rho}_1 + \mathbf{s}_n) \tag{6.38}$$

that depends explicitly on n.

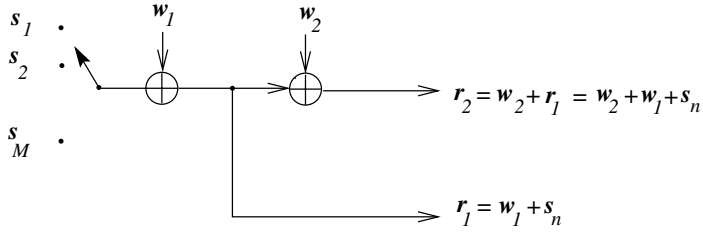


Figure 6.5. Example 6.1.2: the vector $\mathbf{r}_2$ is irrelevant.

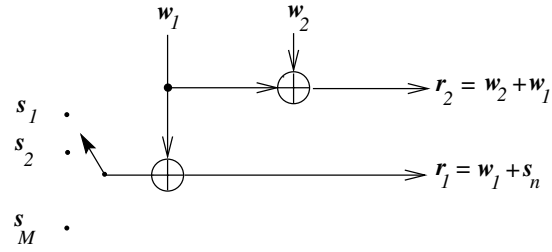


Figure 6.6. Example 6.1.3: the vector $\mathbf{r}_2$ is relevant

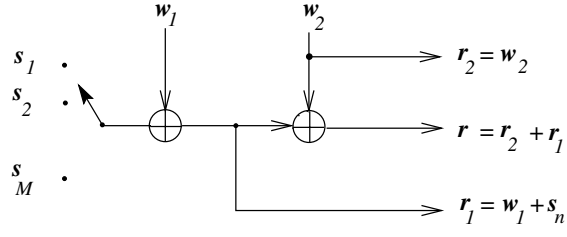


Figure 6.7. Example 6.1.4: the vector $\mathbf{r}_2$ is irrelevant.

Example 6.1.4
In Figure 6.7, *if* the noise vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ are statistically independent, observing (6.34) the signal $\mathbf{r}_2$ can be neglected by the optimum receiver. In fact, it is:

$$p_{\mathbf{r}_2|\mathbf{r}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1, n) = p_{\mathbf{w}_2|\mathbf{w}_1,a_0}(\boldsymbol{\rho}_2 \mid \boldsymbol{\rho}_1 - \mathbf{s}_n, n) = p_{\mathbf{w}_2}(\boldsymbol{\rho}_2) \tag{6.39}$$

that does not depend on n.

We note that Example 6.1.1 is a particular case of Example 6.1.4.

Implementations of the maximum likelihood criterion

We give now two implementations of the ML criterion, assuming that $\{s_m(t)\}$ and $\{\phi_i(t)\}$ have finite duration in the interval $(0, t_0)$.

Implementation type 1. As illustrated in Figure 6.8, there are two fundamental blocks: the first determines the I components of the vector $\mathbf{r}$, and the second computes the M distances

$$D_n = \|\mathbf{r} - \mathbf{s}_n\|^2 \qquad n = 1, \ldots, M \tag{6.40}$$

We note that the filter on branch i has impulse response given by $\text{rect}((t - t_0/2)/t_0)$, and yields the output

$$y_i(t) = \int_{t-t_0}^{t} r(\tau)\phi_i^*(\tau)\, d\tau \tag{6.41}$$

which sampled at $t = t_0$, from (6.14), yields $y_i(t_0) = r_i$.

An equivalent implementation is based on the equivalence illustrated in Figure 6.9, where a correlation demodulator is substituted by a matched filter with impulse response $\phi_i^*(t_0 - t)$.

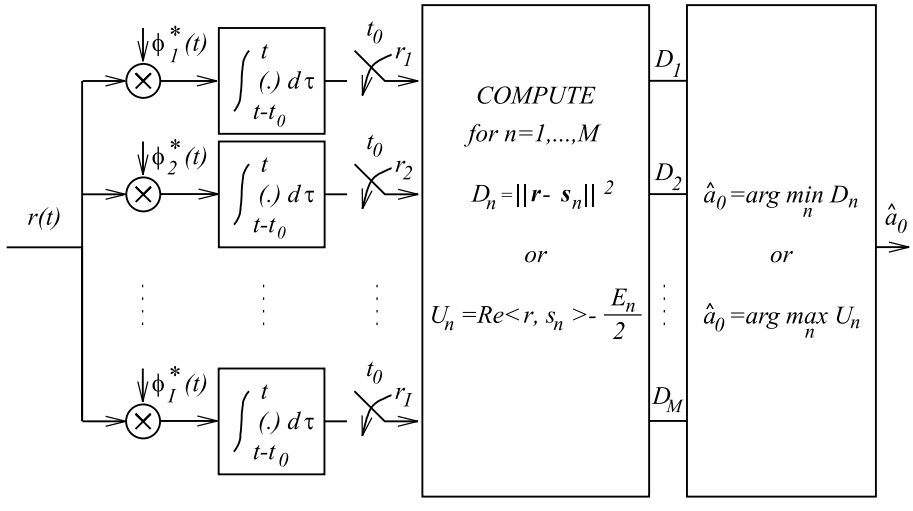


Figure 6.8. Implementation type 1 of the ML criterion.

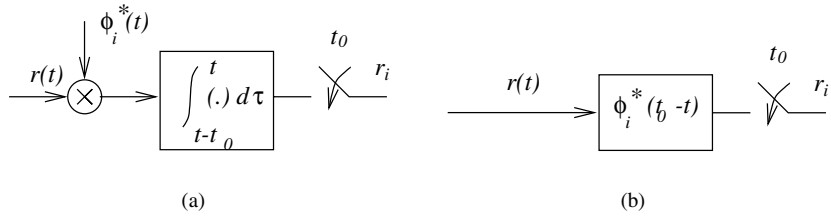


Figure 6.9. (a) Correlation demodulator and equivalent (b) matched filter (MF) demodulator.

Implementation type 2. Using (1.39), from

$$\|\boldsymbol{\rho} - \mathbf{s}_n\|^2 = \|\boldsymbol{\rho}\|^2 + \|\mathbf{s}_n\|^2 - 2Re[\langle \boldsymbol{\rho}, \mathbf{s}_n \rangle] \tag{6.42}$$

the ML criterion becomes

$$\hat{a}_0 = \arg\max_n \left[Re\langle \boldsymbol{\rho}, \mathbf{s}_n \rangle - \frac{\|\mathbf{s}_n\|^2}{2} \right] \tag{6.43}$$

$$= \arg\max_n \left[Re\left[\int_{-\infty}^{+\infty} \rho(t) s_n^*(t)\, dt \right] - \frac{E_n}{2} \right] \tag{6.44}$$

where E_n is the energy of s_n,

$$E_n = \int_{-\infty}^{+\infty} |s_n(t)|^2\, dt \tag{6.45}$$

The implementation of (6.44) is illustrated in Figure 6.10, whereas the equivalent criterion (6.43) is also given in Figure 6.8.

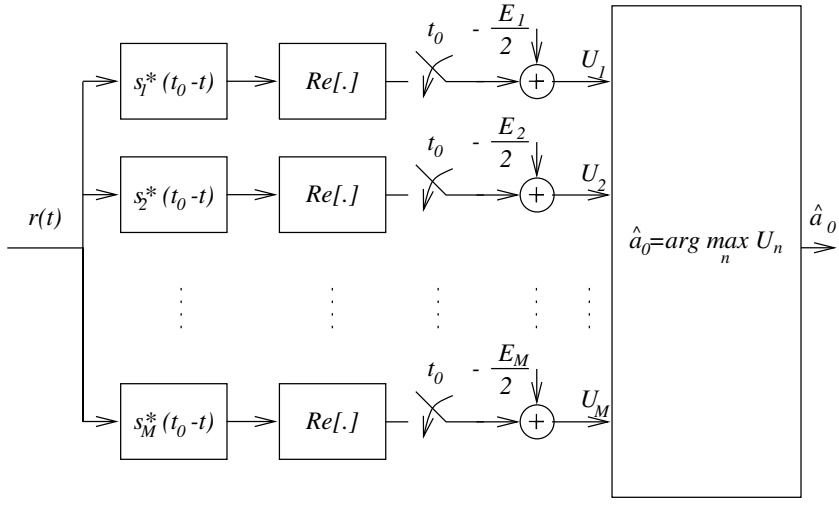


Figure 6.10. Implementation type 2 of the ML criterion.

Typically, in the applications we have $I < M$, hence *implementation type 1* is more convenient.

Error probability

In general, the error probability of the system is defined as

$$P_e = P[\mathrm{E}] = P[\hat{a}_0 \neq a_0] = 1 - P[\mathrm{C}] \tag{6.46}$$

where $P[\mathrm{C}]$ is given by (6.18). Using the total probability theorem, we express the error probability as

$$P_e = \sum_{n=1}^{M} p_n \, P[\mathbf{r} \notin R_n \mid a_0 = n] \tag{6.47}$$

We examine the case of two signals, whose vector representation is illustrated in Figure 6.11. For convenience we choose ϕ_2 parallel to the line joining $\mathbf{s}_1$ and $\mathbf{s}_2$.

Independently of the basis system, assuming ρ real, from (1.41) the squared distance between the two signals is given by

$$d^2 = E_1 + E_2 - 2\rho\sqrt{E_1 E_2} \tag{6.48}$$

where

$$E_i = \int_{-\infty}^{+\infty} |s_i(t)|^2 \, dt \qquad i = 1, 2 \tag{6.49}$$

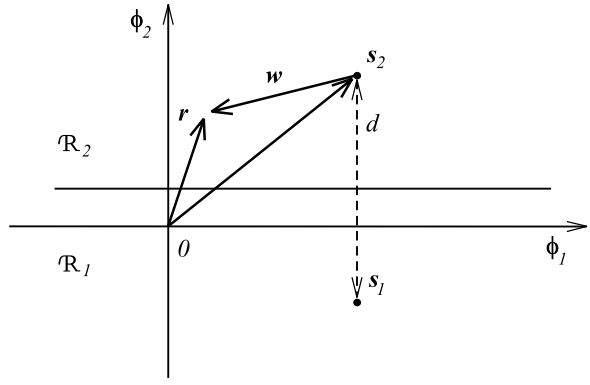


Figure 6.11. Binary constellation and corresponding decision regions.

and

$$\rho = \frac{\int_{-\infty}^{+\infty} s_1(t) s_2^*(t)\, dt}{\sqrt{E_1 E_2}} \tag{6.50}$$

In the case of two equally likely signals, equation (6.47) becomes

$$P_e = \tfrac{1}{2}\{P[\mathbf{r} \notin R_1 \mid a_0 = 1] + P[\mathbf{r} \notin R_2 \mid a_0 = 2]\} \tag{6.51}$$

We assume that $\mathbf{s}_2$ is transmitted, which means $a_0 = 2$. Given the received signal vector $\mathbf{r}$ as in Figure 6.11, we get a decision error if the noise $\mathbf{w} = \mathbf{r} - \mathbf{s}_2$ has a projection on the line joining $\mathbf{s}_1$ and $\mathbf{s}_2$ that is smaller than $-d/2$. As for all projections on an orthonormal basis, the noise component w_2 is Gaussian with zero mean and variance

$$\sigma_I^2 = \frac{N_0}{2} \tag{6.52}$$

Then the conditional error probability is given by

$$P[\mathbf{r} \notin R_2 \mid a_0 = 2] = P\left[w_2 < -\frac{d}{2}\right] = Q\left(\frac{d}{2\sigma_I}\right) \tag{6.53}$$

where

$$Q(a) = \int_a^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{b^2}{2}}\, db \tag{6.54}$$

is the Gaussian complementary distribution function whose values are reported in Appendix 6.A. Likewise, it is

$$P[\mathbf{r} \notin R_1 \mid a_0 = 1] = P\left[w_2 > \frac{d}{2}\right] = Q\left(\frac{d}{2\sigma_I}\right) \tag{6.55}$$

From (6.51), we obtain

$$P_e = Q\left(\frac{d}{2\sigma_I}\right) \tag{6.56}$$

Observation 6.1
The error probability depends only on the ratio between the distance of the signals of the constellation at the decision point and the standard deviation per dimension of the noise. For this reason it is useful to define the following signal-to-noise ratio at the decision point

$$\gamma = \left(\frac{d}{2\sigma_I}\right)^2 \tag{6.57}$$

We will now derive an alternative expression for P_e as a function of the modulator parameters. Substitution of (6.48) and (6.52) in (6.56) yields

$$P_e = Q\left(\sqrt{\frac{E_1 + E_2 - 2\rho\sqrt{E_1 E_2}}{2N_0}}\right) \tag{6.58}$$

If $E_1 = E_2 = E_s$, we get

$$P_e = Q\left(\sqrt{\frac{E_s(1-\rho)}{N_0}}\right) \tag{6.59}$$

6.1.1 Examples of binary signalling

We let $M = 2$ and $E_1 = E_2 = E_s$.

Antipodal signals ($\rho = -1$)

The signal set is composed of two antipodal signals:

$$s_1(t) = s(t) \qquad s_2(t) = -s(t) \qquad s(t) \text{ defined in } (0, T) \tag{6.60}$$

For $E_s = \int_0^T |s(t)|^2\, dt$, we have

$$\rho = \frac{\int_{-\infty}^{+\infty} s_1(t) s_2^*(t)\, dt}{E_s} = -1 \tag{6.61}$$

The basis has only one element, $I = 1$, with

$$\phi(t) = \frac{s(t)}{\sqrt{E_s}} \tag{6.62}$$

The vector representation of Figure 6.12 follows, where

$$\mathbf{s}_1 = \left(\sqrt{E_s}\right) \qquad \mathbf{s}_2 = \left(-\sqrt{E_s}\right) \tag{6.63}$$

The *implementation type 1* of the optimum ML receiver is depicted in Figure 6.13.

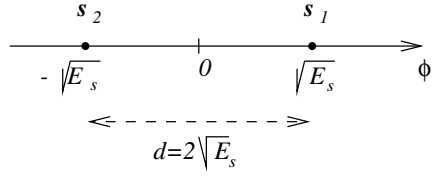


Figure 6.12. Vector representation of antipodal signals.

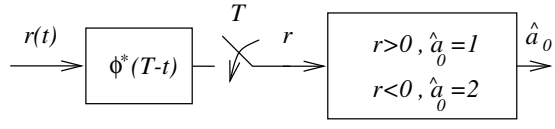


Figure 6.13. ML receiver for binary antipodal signalling.

As $\rho = -1$, (6.59) becomes

$$P_e = Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \tag{6.64}$$

A modulation technique with antipodal signals is *binary phase shift keying* (2-PSK or BPSK), where $s(t)$, defined by (6.60), is shown in Figure 6.14. In this case

$$s_1(t) = A\cos(2\pi f_0 t + \varphi_0) \qquad 0 < t < T \tag{6.65}$$

$$s_2(t) = A\cos(2\pi f_0 t + \varphi_0 + \pi) = -s_1(t) \qquad 0 < t < T \tag{6.66}$$

Orthogonal signals ($\rho = 0$)

We consider the two signals

$$s_1(t) = A\cos(2\pi f_0 t) \qquad s_2(t) = A\sin(2\pi f_0 t) \qquad 0 < t < T \tag{6.67}$$

Observing (1.71), if $f_0 = k/2T$, k integer, or else $f_0 \gg 1/T$, then

$$E_s = E_1 = E_2 = \frac{A^2 T}{2} \quad \text{and} \quad \rho \simeq 0 \tag{6.68}$$

A basis is composed of the signals themselves

$$\phi_1(t) = \frac{s_1(t)}{\sqrt{E_s}} = \sqrt{\frac{2}{T}}\cos(2\pi f_0 t) \qquad 0 < t < T \tag{6.69}$$

and

$$\phi_2(t) = \frac{s_2(t)}{\sqrt{E_s}} = \sqrt{\frac{2}{T}}\sin(2\pi f_0 t) \qquad 0 < t < T \tag{6.70}$$

We note that $\phi_1(t)$ and $\phi_2(t)$ are "windowed" versions of sinusoidal signals.

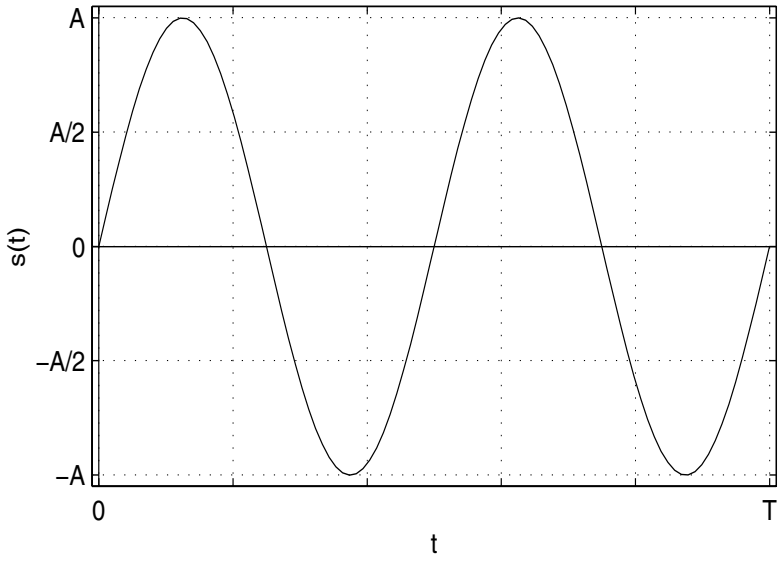


Figure 6.14. Plot of $s(t) = A\cos(2\pi f_0 t + \varphi_0)$, $0 < t < T$, for $f_0 = 2/T$ and $\varphi_0 = -\pi/2$.

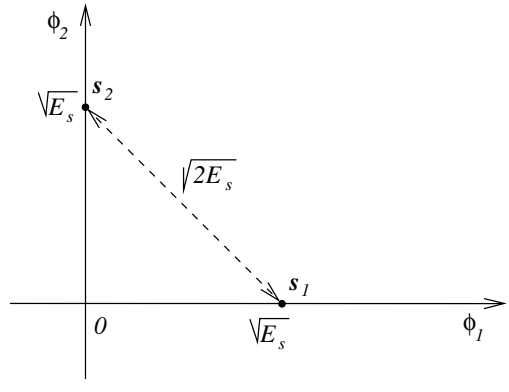


Figure 6.15. Vector representation of orthogonal signals.

The vector representation is given in Figure 6.15. As the two vectors $\mathbf{s}_1$ and $\mathbf{s}_2$ are orthogonal, their distance is $\sqrt{2E_s}$. This distance is reduced by a factor of $\sqrt{2}$ as compared to the case of antipodal signals with the same value of E_s. The optimum ML receiver is depicted in Figure 6.16.

As $\rho = 0$, (6.59) becomes

$$P_e = Q\left(\sqrt{\frac{E_s}{N_0}}\right) \tag{6.71}$$

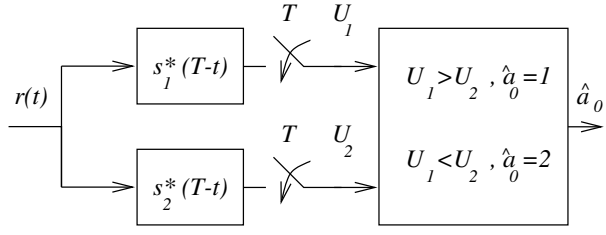


Figure 6.16. ML receiver for binary orthogonal signalling.

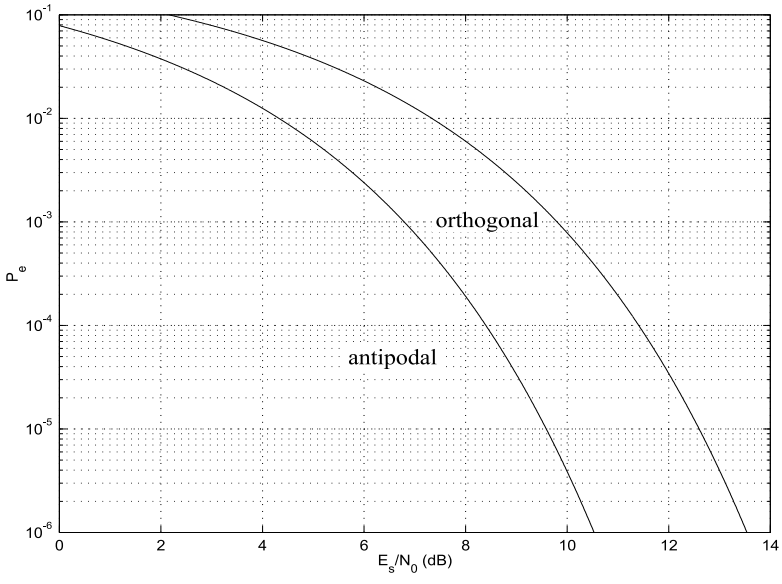


Figure 6.17. Error probability as a function of E_s/N_0 for binary antipodal and orthogonal signalling.

From the curves of probability of error versus E_s/N_0 plotted in Figure 6.17, we note that for a given P_e we have a loss of 3 dB in E_s/N_0 for the orthogonal signalling scheme as compared to the antipodal scheme.

We examine in detail another binary orthogonal signalling scheme.

Binary FSK

We consider the two signals of Figure 6.18, given by

$$s_1(t) = A\cos(2\pi(f_0 - f_d)t + \varphi_0) \qquad s_2(t) = A\cos(2\pi(f_0 + f_d)t + \varphi_0) \qquad 0 < t < T \tag{6.72}$$

where φ_0 is an arbitrary phase.

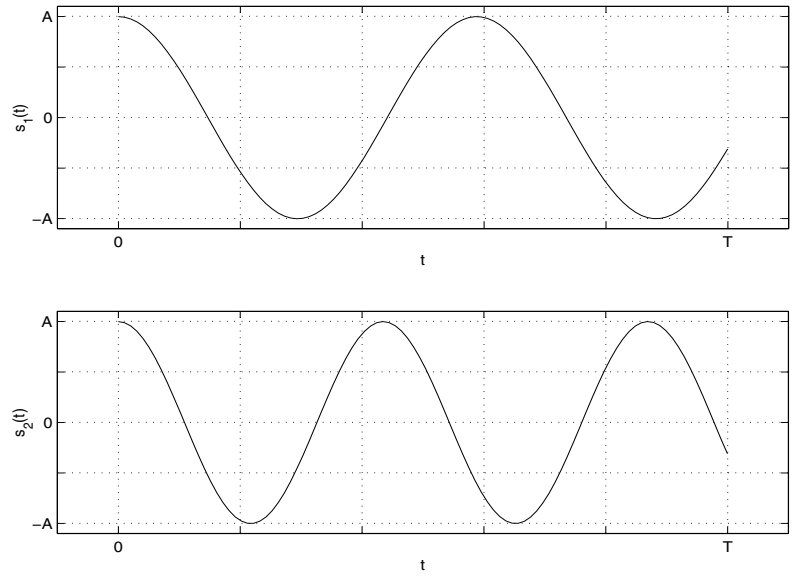


Figure 6.18. Binary FSK signals with $f_0 = 2/T$, $f_d = 0.3/T$ and $\varphi_0 = 0$.

We have two "windowed" sinusoidal functions, one with frequency $f_0 - f_d$ and the other with frequency $f_0 + f_d$; f_0 is called the *carrier frequency*, and f_d is the *frequency deviation*. Also in this case, if $f_0 \pm f_d \gg 1/T$, it holds

$$E_s = E_1 = E_2 \simeq \frac{A^2 T}{2} \tag{6.73}$$

and

$$\rho = \frac{\displaystyle\int_{-\infty}^{+\infty} s_1(t) s_2(t)\, dt}{\dfrac{A^2 T}{2}} = \operatorname{sinc}(4 f_d T) \tag{6.74}$$

As FSK is a binary modulation, we have

$$P_e = Q\left(\sqrt{\frac{E_s(1-\rho)}{N_0}}\right) \tag{6.75}$$

Introducing the *modulation index* h as the ratio between the frequency deviation f_d and the Nyquist frequency of the transmission system equal to $1/(2T)$, we have

$$h = \frac{f_d}{1/(2T)} = 2 f_d T \tag{6.76}$$

Therefore $\rho = \operatorname{sinc}(2h)$.

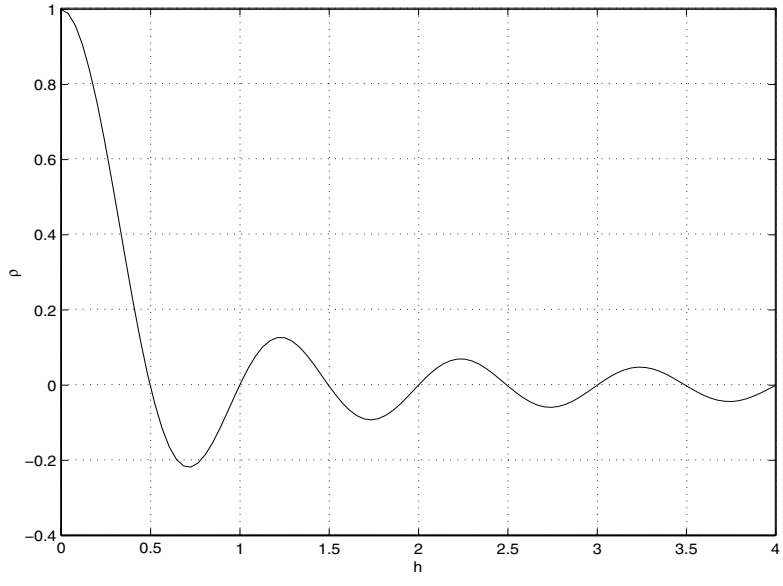


Figure 6.19. Correlation coefficient ρ as a function of the modulation index h.

From the plot of ρ as a function of h illustrated in Figure 6.19, we get the minimum value of ρ, $\rho_{min} = -0.22$, for $h = 0.715$ and $P_e = Q(\sqrt{1.22E_s/N_0})$, with a gain of 0.7 dB in E_s/N_0 as compared to the case $\rho = 0$, and a loss of 2.3 dB with respect to antipodal signalling.

From (6.74) we have $\rho = 0$ for $h = 1/2$, that is for $f_d = 1/(4T)$: in this case we speak of *minimum shift keying* (MSK).[4] There are other values of h (1, 1.5, 2, ...) that yield $\rho = 0$, however, they imply larger f_d, with the consequent requirement of larger channel bandwidth.

6.1.2 Bounds on the probability of error

Let $\{s_n(t)\}$, $n = 1, \ldots, M$, be M equally likely signals, with squared distances

$$d^2(s_n, s_m) = d^2_{nm} = \int_{-\infty}^{+\infty} |s_n(t) - s_m(t)|^2 \, dt \tag{6.77}$$

and

$$d^2_{min} = \min_{n,m} d^2_{nm} \tag{6.78}$$

Upper bound

We assume s_m is transmitted. An error event that leads to choosing s_n is expressed as

$$\mathrm{E}_{nm} = \{r : d(r, s_n) < d(r, s_m)\} \tag{6.79}$$

[4] In fact, MSK also requires that the phase of the modulated signal be continuous (see Section 18.5).

Using (6.53), the probability of the event E_{nm} is given by

$$P[\mathrm{E}_{nm}] = Q\left(\frac{d_{nm}}{2\sigma_I}\right) \tag{6.80}$$

In general, from (6.47), the error probability in the case of equally likely signals is

$$P_e = \sum_{m=1}^{M} P[\mathrm{E} \mid s_m]\frac{1}{M} \tag{6.81}$$

On the other hand, as s_m is transmitted, an error occurs if s_n is chosen, $n = 1, \ldots, M$, $n \neq m$. Therefore, applying the *union bound*, we have

$$P[\mathrm{E} \mid s_m] = P\left[\bigcup_{n=1, n\neq m}^{M} \mathrm{E}_{nm}\right] \leq \sum_{n=1, n\neq m}^{M} P[\mathrm{E}_{nm}] \tag{6.82}$$

Then an upper bound on P_e is given by

$$P_e \leq \frac{1}{M}\sum_{m=1}^{M}\sum_{n=1, n\neq m}^{M} Q\left(\frac{d_{nm}}{2\sigma_I}\right) \tag{6.83}$$

Since $d_{min} \leq d_{nm}$, then $Q(d_{nm}/(2\sigma_I)) \leq Q(d_{min}/(2\sigma_I))$, and a looser bound than (6.83) is given by

$$P_e \leq (M-1)Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{6.84}$$

Lower bound

Given s_n, let $d_{min,n}$ be the distance of s_n from the nearest signal. Limiting the evaluation of the error probability to error events that are associated to the nearest signals, we obtain the following lower bound

$$P_e \geq \frac{1}{M}\sum_{n=1}^{M} Q\left(\frac{d_{min,n}}{2\sigma_I}\right) \tag{6.85}$$

We get a looser bound by introducing N_{min}, the number of signals $\{s_n(t)\}$ whose distance is d_{min} from the nearest signal: $2 \leq N_{min} \leq M$. Limiting the equation in (6.85) to such signals, we have

$$P_e \geq \frac{N_{min}}{M} Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{6.86}$$

In other words, given s_m, an error event E is reduced to considering only a signal at minimum distance, if there is one. In the particular case of $d_{min,n} = d_{min}$, we have $N_{min} = M$, and $P_e \geq Q(d_{min}/(2\sigma_I))$.

For example, for the constellation of Figure 6.3 with $M = 3$ we have

$$d_{12} = \frac{A}{2}\sqrt{T} \qquad d_{13} = A\sqrt{\frac{T}{2}} \qquad d_{23} = \frac{A}{2}\sqrt{T} \tag{6.87}$$

Then $d_{min} = A\sqrt{T}/2$ and $N_{min} = 3$.

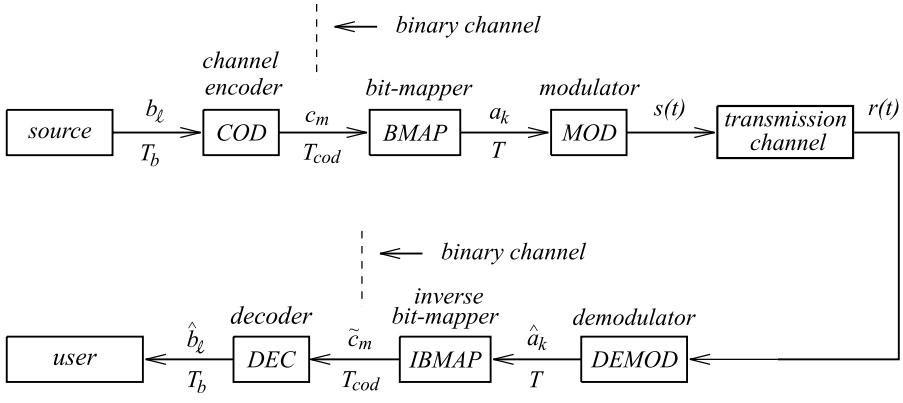


Figure 6.20. Simplified model of a transmission system.

6.2 Simplified model of a transmission system and definition of binary channel

With reference to Figure 6.20, we discuss now some aspects of a communication system, where the information message consists of a sequence of information bits $\{b_\ell\}$ generated at instants ℓT_b. Some bits may be inserted in the message $\{b_\ell\}$ to generate an *encoded message* $\{c_m\}$, according to rules that will be investigated in Chapters 11 and 12. The bits of $\{c_m\}$ are then mapped to the symbols $\{a_k\}$, which assume values in an M-ary alphabet and are generated at instants kT.[5]

The value of the generic symbol a_k is then modulated, that is, it is associated with a waveform, according to the scheme of Figure 6.1:

$$a_k \longrightarrow s_{a_k}(t - kT) \tag{6.88}$$

Therefore the transmitter generates the signal

$$s(t) = \sum_{k=-\infty}^{+\infty} s_{a_k}(t - kT) \tag{6.89}$$

which is sent over the transmission channel. Let $s_{Ch}(t)$ be the signal at the output of the transmission channel, which is assumed to introduce additive white Gaussian noise $w(t)$ with PSD $N_0/2$.

Note that the system of Figure 6.1 has been investigated assuming that an isolated waveform is transmitted. In the model of Figure 6.20, the transmission of a waveform is repeated every *symbol period* T. However, with reference to Figure 6.8, assuming that the transmitted waveforms do not give rise to intersymbol interference (ISI) at the demodulator

[5] Note that there are systems in which encoder and modulator are jointly considered, see, for example, Chapter 12. In that case the notion of binary channel cannot be referred to the transmission of the sequence $\{c_m\}$.

output[6] we can still study the system assuming that an isolated symbol is transmitted, for example, the symbol a_0 transmitted at instant $t = 0$.

At the receiver, the bits $\{\tilde{c}_\ell\}$ are obtained by inverse bit mapping from the detected message $\{\hat{a}_k\}$. The information bits $\{\hat{b}_\ell\}$ are then recovered by a decoding process.

Definition 6.1
The transformation that maps $\tilde{c}_m$ into c_m is called a *binary channel*. It is characterized by the bit rate $1/T_{cod}$, which is the transmission rate of the bits of the sequence $\{c_m\}$, and by the bit error probability

$$P_{bit} = P_{BC} = P[\tilde{c}_m \neq c_m] \qquad \tilde{c}_m, c_m \in \{0, 1\} \tag{6.90}$$

In the case of a *binary symmetric channel* (BSC), it is assumed that $P[\tilde{c}_m \neq c_m \mid c_m = 0] = P[\tilde{c}_m \neq c_m \mid c_m = 1]$. We say that the BSC is *memoryless* if, for every choice of N distinct instants $m_1, m_2, \ldots, m_N$, the following relation holds:

$$\begin{aligned} &P[\tilde{c}_{m_1} \neq c_{m_1}, \tilde{c}_{m_2} \neq c_{m_2}, \ldots, \tilde{c}_{m_N} \neq c_{m_N}] \\ &\quad = P[\tilde{c}_{m_1} \neq c_{m_1}]\, P[\tilde{c}_{m_2} \neq c_{m_2}] \ldots P[\tilde{c}_{m_N} \neq c_{m_N}] \end{aligned} \tag{6.91}$$

In a *memoryless binary symmetric channel* the probability distribution of $\{\tilde{c}_m\}$ is obtained from that of $\{c_m\}$ and P_{BC} according to the statistical model shown in Figure 6.21.

We note that the aim of the channel encoder is to introduce redundancy in the sequence $\{c_m\}$, which is exploited by the decoder to detect and/or correct errors introduced by the binary channel.

The overall objective of the transmission system is to reproduce the sequence of information bits $\{b_\ell\}$ with a high degree of reliability, measured by the bit error probability

$$P_{bit}^{(dec)} = P[\hat{b}_\ell \neq b_\ell] \tag{6.92}$$

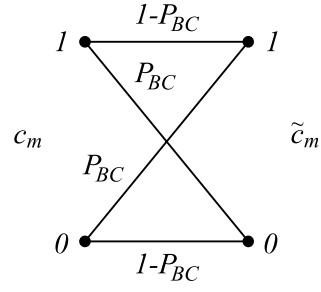


Figure 6.21. Memoryless binary symmetric channel.

[6] Absence of ISI in this context means that the *optimum reception* of the waveform transmitted at instant kT, $s_{a_k}(t - kT)$, is not influenced by the presence of the waveforms associated with symbols transmitted at other instants. For example, all signalling schemes that employ pulses with finite duration in the interval $(0, T)$ do not give rise to ISI. However, this is a particular case of the Nyquist criterion for the absence of ISI that will be discussed in Section 7.3.3.

Typically, it is required $P_{bit}^{(dec)} \simeq 10^{-2}$–$10^{-3}$ for PCM or ADPCM coded speech (see Chapter 5) and $P_{bit}^{(dec)} \simeq 10^{-7}$–$10^{-11}$ for data messages.

Parameters of a transmission system

We give several general definitions widely used to describe the various modulation systems that will be treated in the following sections.

As in practical systems the transmitted signal s is distorted by the transmission channel, we consider the desired signal at the receiver input, s_{Ch}; in particular, for an ideal AWGN channel $s_{Ch}(t) = s(t)$.

- T_b: bit period (s). It is equal to the time interval between two consecutive bits of the information message. We assume the message $\{b_\ell\}$ is composed of binary i.i.d. symbols.
- $R_b = 1/T_b$: bit rate of the system (bit/s).
- T: modulation interval or symbol period (s).
- $1/T$: modulation rate or symbol rate (Baud).
- L_b: number of information message bits per modulation interval.
- M: cardinality of the alphabet of symbols at the transmitter.
- I: number of dimensions of the signal space or of the signal constellation.
- R_I: rate of the encoder-modulator (bit/dim).
- $\mathtt{M}_{s_{Ch}}$: statistical power of the desired signal at the receiver input (V^2).
- $E_{s_{Ch}}$: average energy of an isolated pulse (V^2s).
- E_I: average energy per dimension (V^2s/dim).
- E_b: average energy per information bit (V^2s/bit).
- $N_0/2$: spectral density of additive white noise introduced by the channel (V^2/Hz).
- B_{min}: *conventional* minimum bandwidth of the modulated signal (Hz).
- ν: spectral efficiency of the system (bit/s/Hz).
- Γ: *conventional* signal-to-noise ratio at the receiver input.
- Γ_I: signal-to-noise ratio per dimension.
- $\mathtt{P}_{s_{Ch}}$: available power of the desired signal at the receiver input (W).
- $\mathtt{T}_{w_i}$: effective receiver noise temperature (K).
- $\mathcal{S}$: *sensitivity* (W). It expresses the minimum value of $\mathtt{P}_{s_{Ch}}$ such that the system achieves a given performance in terms of bit error probability.

Relations among parameters

1. *Rate of the encoder-modulator*:

$$R_I = \frac{L_b}{I} \tag{6.93}$$

2. *Number of information bits per modulation interval*: via the channel encoder (COD) and the bit-mapper (BMAP), L_b information bits of the message $\{b_\ell\}$ are mapped in an M-ary symbol, a_k. In general we have

$$L_b \leq \log_2 M \tag{6.94}$$

where the equality holds for a system without coding, or, with abuse of language, for an uncoded system. In this case we also have

$$R_I = \frac{\log_2 M}{I} \tag{6.95}$$

3. *Symbol period*:

$$T = T_b L_b \tag{6.96}$$

4. *Statistical power of the desired signal at the receiver input*:

$$\mathtt{M}_{s_{Ch}} = \frac{E_{s_{Ch}}}{T} \tag{6.97}$$

We note that, for *continuous* transmission (see Chapter 7), $\mathtt{M}_{s_{Ch}}$ is finite and consequently we define $E_{s_{Ch}} = \mathtt{M}_{s_{Ch}} T$; on the other hand, for transmission of an *isolated pulse* $E_{s_{Ch}}$ is finite and we define $\mathtt{M}_{s_{Ch}}$ through (6.97).

5. *Average energy per dimension*:

$$E_I = \frac{E_{s_{Ch}}}{I} \tag{6.98}$$

6. *Average energy per information bit*:

$$E_b = \frac{E_I}{R_I} = \frac{E_{s_{Ch}}}{L_b} \tag{6.99}$$

For an uncoded system, (6.99) becomes:

$$E_b = \frac{E_{s_{Ch}}}{\log_2 M} \tag{6.100}$$

7. *Conventional minimum bandwidth of the modulated signal*:

$$B_{min} = \frac{1}{2T} \qquad \text{for baseband signals} \tag{6.101}$$

$$B_{min} = \frac{1}{T} \qquad \text{for passband signals} \tag{6.102}$$

For the orthogonal and biorthogonal signals of Section 6.7, the definition of B_{min} will be different and will include the factor $1/M$.

8. *Spectral efficiency*:

$$\nu = \frac{1/T_b}{B_{min}} = \frac{L_b}{B_{min}T} \tag{6.103}$$

In practice ν measures how many bits per unit of time are sent over a channel with the conventional bandwidth B_{min}. In terms of R_I, from (6.93), we have

$$\nu = \frac{R_I I}{B_{min}T} \tag{6.104}$$

Later we will see that, for most uncoded systems, $R_I = \nu/2$.

9. *Conventional signal-to-noise ratio at the receiver input*:

$$\Gamma = \frac{\mathtt{M}_{s_{Ch}}}{(N_0/2)2B_{min}} = \frac{E_{s_{Ch}}}{N_0 B_{min}T} \tag{6.105}$$

In general Γ expresses the ratio between the statistical power of the desired signal at the receiver input and the statistical power of the noise measured with respect to the conventional bandwidth B_{min}. We note that, for the same value of $N_0/2$, if B_{min} doubles, the statistical power must also double to maintain a given ratio Γ.

10. *Signal-to-noise ratio per dimension*:

$$\Gamma_I = \frac{E_I}{N_0/2} = \frac{2E_{s_{Ch}}}{N_0 I} \tag{6.106}$$

is the ratio between the energy per dimension of an isolated pulse E_I and the noise variance per dimension σ_I^2 given by (6.13). Using (6.99), the general relation becomes

$$\Gamma_I = 2R_I \frac{E_b}{N_0} \tag{6.107}$$

It is interesting to observe that in most modulation systems it turns out $\Gamma_I = \Gamma$.

11. *Link budget*: if the receiver is matched to the transmission medium for the maximum transfer of power, from (4.92) we obtain an alternative expression of (6.105) given by

$$\Gamma = \frac{\mathtt{P}_{s_{Ch}}}{\mathtt{k}\mathtt{T}_{w_i} B_{min}} \tag{6.108}$$

We observe that (6.105) is useful to analyze the system, and (6.108) is usually employed to evaluate the *link budget*.

In the next sections some examples of modulation systems *without channel coding* are illustrated.

6.3 Pulse amplitude modulation (PAM)

Pulse amplitude modulation, also called *amplitude shift keying* (ASK), is the first example of multilevel baseband signalling, i.e. M may take values larger than 2. Let h_{Tx} be a real-valued finite-energy pulse with support $(0, t_0)$; a transmitted isolated pulse is expressed as

$$s_n(t) = \alpha_n h_{Tx}(t) \qquad t \in \Re \qquad n = 1, 2, \ldots, M \tag{6.109}$$

where

$$\alpha_n = 2n - 1 - M \tag{6.110}$$

In other words, PAM signals are obtained by modulating in amplitude the pulse shape h_{Tx}.

Energy of s_n:

$$E_n = \alpha_n^2 E_h \qquad E_h = \int_{-\infty}^{+\infty} |h_{Tx}(t)|^2 \, dt \tag{6.111}$$

Average energy of the system:[7]

$$E_s = \frac{1}{M} \sum_{n=1}^{M} E_n = \frac{M^2 - 1}{3} E_h \tag{6.112}$$

Basis function:

$$\phi(t) = \frac{h_{Tx}(t)}{\sqrt{E_h}} \tag{6.113}$$

Vector representation:

$$s_n = \alpha_n \sqrt{E_h} \qquad n = 1, \ldots, M \tag{6.114}$$

as illustrated in Figure 6.22 for $M = 8$. The minimum distance is equal to

$$d_{min} = 2\sqrt{E_h} = d \tag{6.115}$$

The transmitter is shown in Figure 6.23. The bit mapper is composed of a serial-to-parallel (S/P) converter followed by a map that translates a sequence of $\log_2 M$ bits into the corresponding value of a_0. The map is a Gray encoder (see Appendix 6.B). An example for $M = 8$ is illustrated in Table 6.1. The symbol α_n is input to an interpolator filter with impulse response h_{Tx}. The filter output yields the transmitted signal s_{a_0}.

---

[7] A few useful formulae are:

$$\sum_{i=1}^{M} i = \frac{M(M+1)}{2} \qquad \sum_{i=1}^{M} i^2 = \frac{M(M+1)(2M+1)}{6} \qquad \sum_{i=1}^{M} i^3 = \left(M \frac{M+1}{2} \right)^2$$

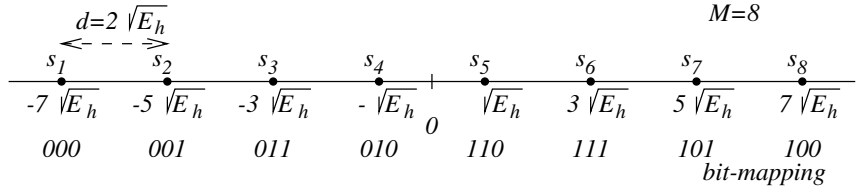


Figure 6.22. Vector representation, or signal constellation, of an 8-PAM system.

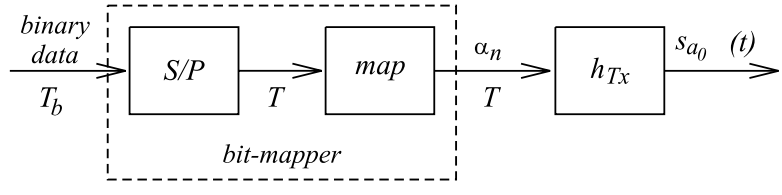


Figure 6.23. Transmitter of a PAM system for an isolated pulse.

Table 6.1 Bit map for a 8-PAM.

Gray coding of symbols ($M = 8$)		
Three-bit sequence	α_n	a_0
000	-7	1
001	-5	2
011	-3	3
010	-1	4
110	1	5
111	3	6
101	5	7
100	7	8

The type 1 implementation of the ML receiver is shown in Figure 6.24 and consists of a matched filter to h_{Tx} followed by a sampler. In this case, from (6.15) r is given by

$$r = s_n + w \tag{6.116}$$

where $s_n = \alpha_n (d/2)$, $n = 1, \ldots, M$, and w is a real-valued Gaussian r.v. with zero mean and variance $N_0/2$.

From the observation of r, a threshold detector yields the detected symbol $\hat{a}_0$. The transmitted bits are then recovered by an inverse bit mapper.

Minimum bandwidth of the modulated signal, equal to the Nyquist frequency, see Definition 7.1 on page 559*:*

$$B_{min} = \frac{1}{2T} \tag{6.117}$$

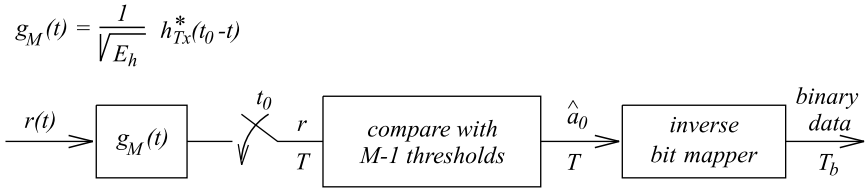


Figure 6.24. ML receiver, implementation type 1, of a PAM system for an isolated pulse. The thresholds are set at $(2n - M)\sqrt{E_h} = (n - (M/2))\, d$, $n = 1, 2, \ldots, M - 1$.

Spectral efficiency:

$$\nu = \frac{(1/T)\log_2 M}{1/(2T)} = 2\log_2 M \ \text{(bit/s/Hz)} \tag{6.118}$$

Signal-to-noise ratio: from (6.105) it follows

$$\Gamma = \frac{E_s}{N_0/2} \tag{6.119}$$

Note that (6.119) expresses Γ as the ratio between the signal energy and the variance of the noise component: therefore, as $I = 1$, it follows that $\Gamma_I = \Gamma$.

Symbol error probability: from the total probability theorem, letting $d = 2\sqrt{E_h}$, and considering the outer constellation symbols separately from the others, we have

$$\begin{aligned}
P[\mathrm{E} \mid s_M] &= P[\mathrm{E} \mid s_1] \\
&= P[\hat{a}_0 \neq 1 \mid a_0 = 1] \\
&= P\left[r > \left(1 - \frac{M}{2}\right) d \mid a_0 = 1\right] \\
&= P\left[\alpha_1 \frac{d}{2} + w > \left(1 - \frac{M}{2}\right) d \mid a_0 = 1\right] \\
&= P\left[(1 - M)\frac{d}{2} + w > \left(1 - \frac{M}{2}\right) d\right] \\
&= P\left[w > \frac{d}{2}\right] \\
&= Q\left(\frac{d}{2\sigma_I}\right)
\end{aligned} \tag{6.120}$$

and

$$\begin{aligned} P[\mathrm{E} \mid s_n] &= P[\hat{a}_0 \neq n \mid a_0 = n] \qquad n = 2, \ldots, M-1 \\ &= P\left[\left\{r < (\alpha_n - 1)\frac{d}{2}\right\} \cup \left\{r > (\alpha_n + 1)\frac{d}{2}\right\} \mid a_0 = n\right] \\ &= P\left[\alpha_n \frac{d}{2} + w < (\alpha_n - 1)\frac{d}{2}\right] + P\left[\alpha_n \frac{d}{2} + w > (\alpha_n + 1)\frac{d}{2}\right] \\ &= P\left[w < -\frac{d}{2}\right] + P\left[w > \frac{d}{2}\right] \\ &= 2Q\left(\frac{d}{2\sigma_I}\right) \end{aligned} \tag{6.121}$$

where $\sigma_I^2 = N_0/2$. Then, for equally likely symbols we have

$$\begin{aligned} P_e &= \frac{1}{M}\left[2Q\left(\frac{d}{2\sigma_I}\right) + (M-2)2Q\left(\frac{d}{2\sigma_I}\right)\right] \\ &= 2\left(1 - \frac{1}{M}\right) Q\left(\frac{d}{2\sigma_I}\right) \end{aligned} \tag{6.122}$$

In terms of E_s we get[8]

$$P_e = 2\left(1 - \frac{1}{M}\right) Q\left(\sqrt{\frac{6}{M^2 - 1}\frac{E_s}{N_0}}\right) \tag{6.123}$$

In terms of Γ, substitution of (6.119) into (6.123) yields

$$P_e = 2\left(1 - \frac{1}{M}\right) Q\left(\sqrt{\frac{3}{M^2 - 1}\Gamma}\right) \tag{6.124}$$

Assuming that Gray coding is adopted at the transmitter, the bit error probability is given by

$$P_{bit} \simeq \frac{P_e}{\log_2 M} \qquad \text{valid for} \quad \Gamma \gg 1 \tag{6.125}$$

Equation (6.125) expresses the fact that, for Γ sufficiently large, if an error event occurs, it is very likely that one of the symbols at the minimum distance from the transmitted symbol is detected. Thus with high probability only one bit of the $\log_2 M$ bits associated with the transmitted symbol is incorrectly recovered.

Curves of P_{bit} as a function of Γ are shown in Figure 6.25 for different values of M.

In the Appendix 6.C, two other baseband modulation schemes are described: *pulse position modulation* (PPM) and *pulse duration modulation* (PDM).

[8] These results are valid also for continuous transmission with modulation period T, assuming absence of ISI at the decision point; in other words, the autocorrelation function of $h_{Tx}(t)$ must be a Nyquist pulse (see Section 7.3.3).

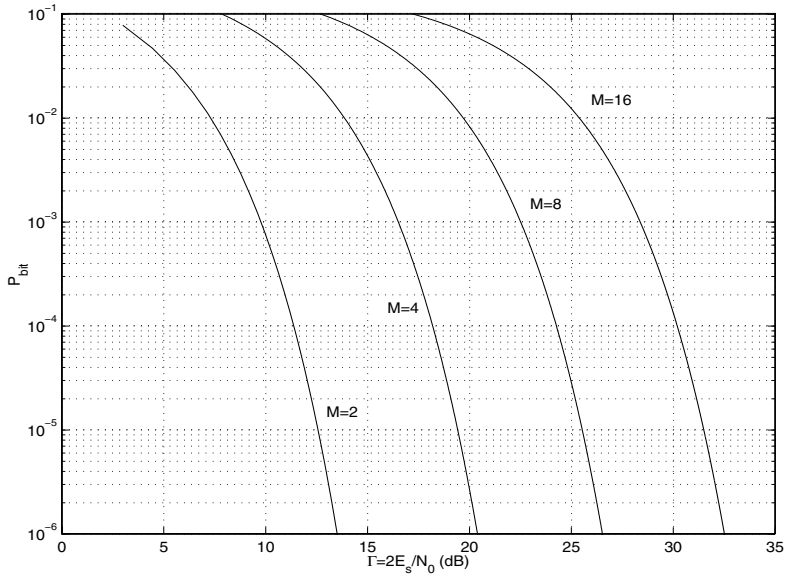


Figure 6.25. Bit error probability as a function of Γ for M-PAM transmission.

6.4 Phase-shift keying (PSK)

Phase-shift keying is an example of passband modulation. Let h_{Tx} be a real-valued finite-energy baseband pulse with support $(0, t_0)$. Let[9]

$$\varphi_n = \frac{\pi}{M}(2n-1) \qquad n = 1, \ldots, M \tag{6.126}$$

then the generic transmitted pulse is given by

$$s_n(t) = h_{Tx}(t)\cos(2\pi f_0 t + \varphi_n) \qquad t \in \Re \qquad n = 1, \ldots, M \tag{6.127}$$

that is, signals are obtained by choosing one of the M possible values of the phase of a sinusoidal function with frequency f_0, modulated by h_{Tx}.[10]

In the following sections we will denote by θ_k the r.v. that determines the transmitted signal phase at instant kT. Consequently, the values of θ_k are given by φ_n, $n = 1, \ldots, M$. In this section we consider the case of an isolated pulse transmitted at instant $k = 0$.

An alternative expression of (6.127) is given by

$$s_n(t) = Re[e^{j\varphi_n} h_{Tx}(t) e^{j2\pi f_0 t}] \tag{6.128}$$

---

[9] A more general definition of φ_n is given by $\varphi_n = \frac{\pi}{M}(2n-1) + \varphi_0$, where φ_0 is a constant phase.

[10] We obtain (6.66) by assuming $h_{Tx}(t) = w_T(t)$, where w_T is the rectangular window of duration T defined in (1.473).

Moreover, setting

$$\alpha_n = e^{j\varphi_n} = e^{j\frac{\pi}{M}(2n-1)} \tag{6.129}$$

we have

$$s_n(t) = \alpha_{n,I} h_{Tx}(t)\cos(2\pi f_0 t) - \alpha_{n,Q} h_{Tx}(t)\sin(2\pi f_0 t) \tag{6.130}$$

where

$$\alpha_{n,I} = Re[\alpha_n] = \cos\frac{\pi}{M}(2n-1) \tag{6.131}$$

$$\alpha_{n,Q} = Im[\alpha_n] = \sin\frac{\pi}{M}(2n-1) \tag{6.132}$$

Energy of s_n: if f_0 is greater than the bandwidth of h_{Tx}, using Parseval theorem we get

$$E_n = \frac{E_h}{2} \tag{6.133}$$

Average energy of the system:

$$E_s = \frac{1}{M}\sum_{n=1}^{M} E_n = \frac{E_h}{2} \tag{6.134}$$

Basis functions:

$$\phi_1(t) = \sqrt{\frac{2}{E_h}} h_{Tx}(t)\cos(2\pi f_0 t) \tag{6.135}$$

$$\phi_2(t) = -\sqrt{\frac{2}{E_h}} h_{Tx}(t)\sin(2\pi f_0 t) \tag{6.136}$$

Vector representation:

$$\mathbf{s}_n = \sqrt{\frac{E_h}{2}}\left[\cos\left(\frac{\pi}{M}(2n-1)\right), \sin\left(\frac{\pi}{M}(2n-1)\right)\right]^T \qquad n = 1, 2, \ldots, M \tag{6.137}$$

as illustrated in Figure 6.26 for $M = 8$.

We note that the desired signal at the decision point, $\boldsymbol{s}_n$, aside from the factor $\sqrt{\frac{E_h}{2}}$, coincides with α_n. Note that the signal constellation lies on a circle and the various vectors differ in the phase φ_n.

The minimum distance is equal to

$$d_{min} = 2\sqrt{E_s}\sin\frac{\pi}{M} = \sqrt{2E_h}\sin\frac{\pi}{M} \tag{6.138}$$

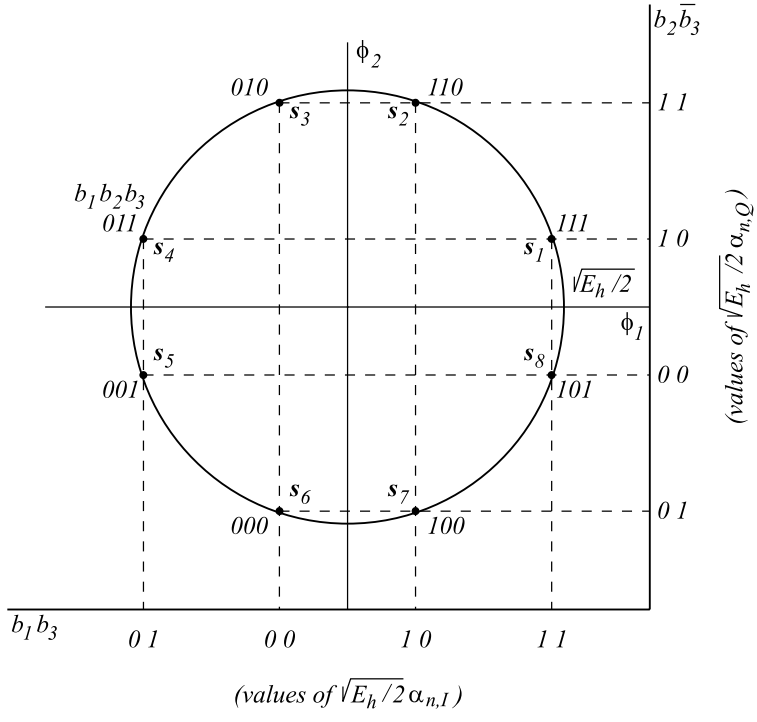


Figure 6.26. Signal constellation of an 8-PSK system.

In Figure 6.26, the projections of $\mathbf{s}_n$ on the axis ϕ_1 (in phase) and on the axis ϕ_2 (quadrature) are also represented, together with the Gray coding of the various symbols represented by the bits b_1, b_2, b_3.

A PSK transmitter for $M = 8$ is shown in Figure 6.27. The bit mapper maps a sequence of $\log_2 M$ bits to a constellation point with value α_n. The quadrature components $\alpha_{n,I}$ and $\alpha_{n,Q}$ are input to interpolator filters h_{Tx}. The filter output signals are multiplied by the carrier signal, $\cos(2\pi f_0 t)$, and by the carrier signal phase-shifted by $\pi/2$, for example, by a Hilbert filter, $\sin(2\pi f_0 t)$, respectively. The transmitted signal (6.130) is obtained by adding the two components.

The type 1 implementation of the ML receiver is illustrated in Figure 6.28. From the general scheme of Figure 6.8, we note that the basis functions (6.136) are implemented partially by a correlator with a sinusoidal signal, and partially by a filter matched to h_{Tx}. From Figure 6.26 we note that the decision regions are angular sectors with phase $2\pi/M$. For $M = 2, 4, 8$, simple decision rules can be defined. For $M > 8$ detection can be made by observing the phase v_r of the received vector[11] $\mathbf{r} = [r_I, r_Q]^T$.

[11] For the sake of notation uniformity with the following chapters, the components r_1 and r_2 of $\mathbf{r}$ will be indicated by r_I and r_Q, respectively.

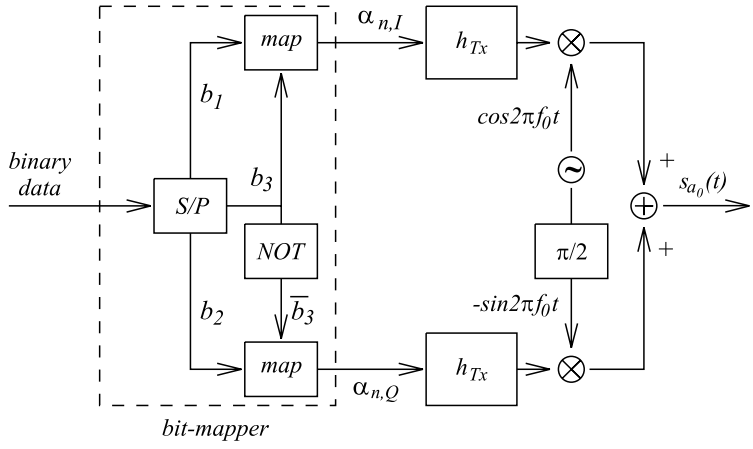


Figure 6.27. Transmitter of an 8-PSK system for an isolated pulse.

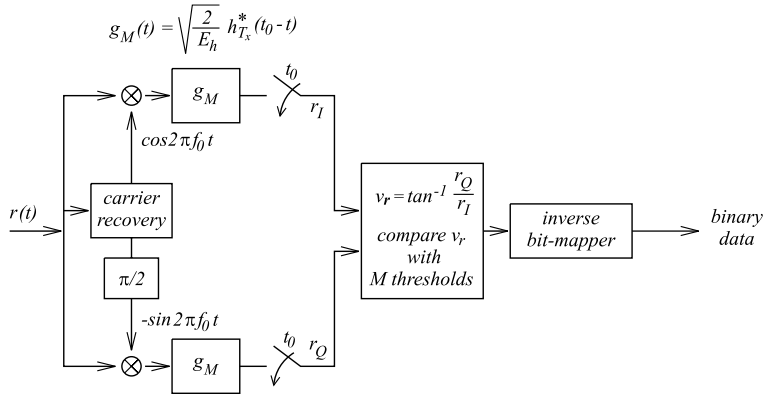


Figure 6.28. ML receiver, implementation type 1, of an M-PSK system for an isolated pulse. Thresholds are set at $(2\pi/M)n$, $n = 1, \ldots, M$.

Minimum bandwidth of the modulated signal (passband signal)*:*

$$B_{min} = \frac{1}{T} \tag{6.139}$$

Spectral efficiency:

$$\nu = \frac{(1/T)\log_2 M}{1/T} = \log_2 M \text{ (bit/s/Hz)} \tag{6.140}$$

We note that for $M = L^2$ we have the same spectral efficiency as PAM.

Signal-to-noise ratio: from (6.105) we have

$$\Gamma = \frac{E_s}{N_0} = \frac{E_s/2}{\sigma_I^2} \tag{6.141}$$

We note that Γ also expresses the ratio between the energy per dimension and the variance of the noise components; moreover $\Gamma_I = \Gamma$ if $M > 2$.

Symbol error probability: with equally likely signals, exploiting the symmetry of the signalling scheme we get

$$\begin{aligned} P_e &= P[\mathrm{E} \mid s_n] \\ &= 1 - P[\mathrm{C} \mid s_n] \\ &= 1 - P[\mathbf{r} \in R_n \mid a_0 = n] \\ &= 1 - \iint_{R_n} p_{\mathbf{r}}(\rho_I, \rho_Q \mid a_0 = n)\, d\rho_I\, d\rho_Q \end{aligned} \tag{6.142}$$

where the angular sector R_n is illustrated in Figure 6.29. For $a_0 = n$ we get $\mathbf{r} = \mathbf{w} + \mathbf{s}_n$, where $\mathbf{s}_n$ is given by (6.137); then, observing (6.16), (6.142) becomes

$$P_e = 1 - \iint_{R_n} \frac{1}{\pi N_0} \exp\left\{-\frac{1}{N_0}\left[\left(\rho_I - \sqrt{E_s}\cos\varphi_n\right)^2 + \left(\rho_Q - \sqrt{E_s}\sin\varphi_n\right)^2\right]\right\} d\rho_I\, d\rho_Q \tag{6.143}$$

Using polar coordinates, we get

$$P_e = 1 - \int_{-\frac{\pi}{M}}^{\frac{\pi}{M}} p_\theta(z)\, dz \tag{6.144}$$

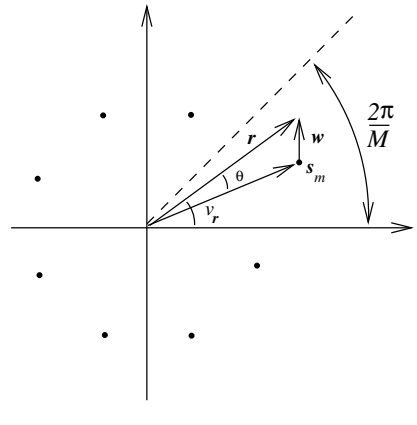


Figure 6.29. Decision region for an M-PSK signal.

where

$$p_\theta(z) = \frac{e^{-E_s/N_0}}{2\pi}\left\{1 + \sqrt{\frac{\pi E_s}{N_0}} e^{(E_s/N_0)\cos^2 z} 2\cos z \left[1 - Q\left(\sqrt{\frac{2E_s}{N_0}}\cos z\right)\right]\right\} \tag{6.145}$$

for $-\pi \le z \le \pi$.

The integral (6.144) cannot be solved in closed form. If $E_s/N_0 \gg 1$, for $M \ge 4$ we can use the approximation (6.363) in (6.145) to obtain

$$p_\theta(z) \simeq \sqrt{\frac{E_s}{\pi N_0}} \cos z e^{-\frac{E_s}{N_0}\sin^2 z} \tag{6.146}$$

In turn, substituting (6.146) in (6.144), and observing (6.141), we get

$$\begin{aligned} P_e &= 2Q\left(\sqrt{\frac{2E_s}{N_0}}\sin\frac{\pi}{M}\right) \\ &= 2Q\left(\sqrt{2\Gamma}\sin\frac{\pi}{M}\right) \end{aligned} \tag{6.147}$$

Assuming that Gray coding is adopted at the transmitter, the bit error probability is given by

$$P_{bit} = \frac{P_e}{\log_2 M} \qquad \text{valid for} \quad \frac{E_s}{N_0} \gg 1 \tag{6.148}$$

Curves of P_{bit} as a function of $\Gamma = E_s/N_0$ are shown in Figure 6.30. We consider in detail two particular cases.

Binary PSK (BPSK)

For $M = 2$ we get $\varphi_1 = \varphi_0$ and $\varphi_2 = \pi + \varphi_0$, where φ_0 is an arbitrary phase. Then $I = 1$, and a basis is given by the function

$$\phi_1(t) = \sqrt{\frac{2}{E_h}} h_{Tx}(t)\cos(2\pi f_0 t + \varphi_0) \tag{6.149}$$

The signal constellation, illustrated in Figure 6.31, comprises the vectors $s_1 = \sqrt{E_s}$ and $s_2 = \sqrt{E_s}$, hence $d_{min} = 2\sqrt{E_s}$. Therefore

$$\Gamma_I = 2\Gamma \tag{6.150}$$

Moreover, it is $\nu = 1$. This result is due to the fact that a BPSK system does not efficiently use the available bandwidth $1/T$: in fact only half of the band carries information. The information in the other half can be deduced by symmetry and is therefore redundant.

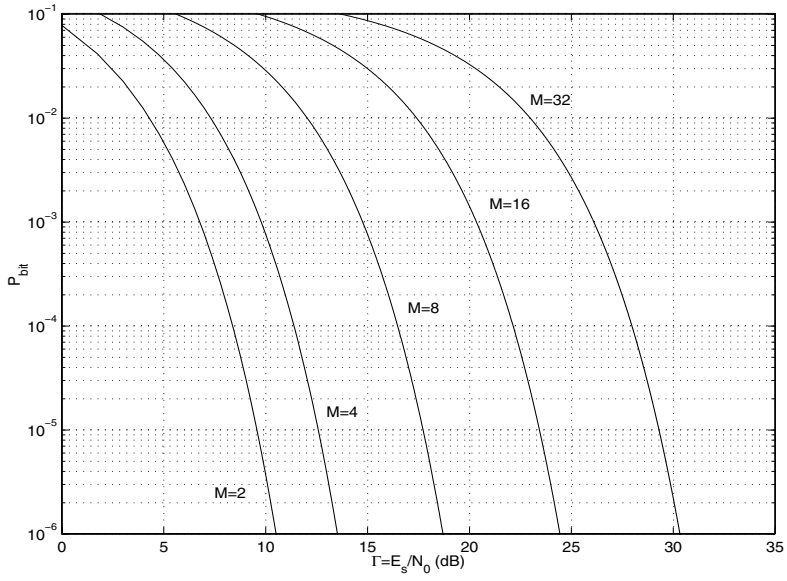


Figure 6.30. Bit error probability as a function of Γ for M-PSK transmission.

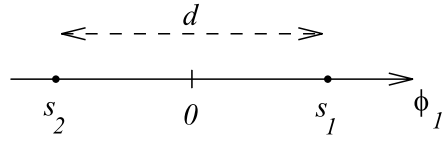
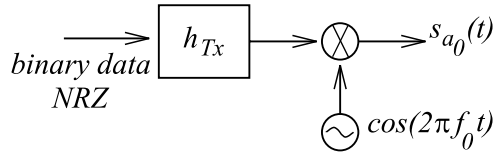


Figure 6.31. Signal constellation of a BPSK system.

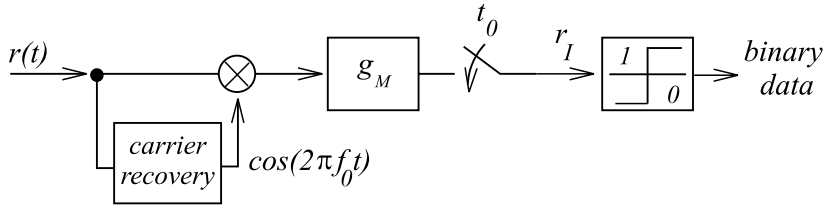
From (6.64), obtained for antipodal signals, and using (6.141), the evaluation of P_e yields

$$\begin{aligned} P_e &= P_{bit} \\ &= Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \\ &= Q(\sqrt{2\Gamma}) \end{aligned} \tag{6.151}$$

The transmitter and the receiver for a BPSK system are shown in Figure 6.32 and have a very simple implementation. The bit mapper of the transmitter maps '0' in '-1' and '1' in '$+1$' to generate NRZ binary data (see Appendix 7.A). At the receiver, the decision element implements the "sign" function to detect NRZ binary data. The inverse bit mapping to recover the bits of the information message is straightforward.



(a) Transmitter



(b) Receiver

Figure 6.32. Schemes of transmitter and receiver for a BPSK system with $\varphi_0 = 0$.

Quadrature PSK (QPSK)

PSK for $M = 4$ is usually called *quadrature* PSK (QPSK). With reference to the vector representation of Figure 6.33, as w_I and w_Q are statistically independent, the probability of correct decision is given by

$$P[\mathrm{C} \mid \mathbf{s}_1] = P\left[w_I > -\frac{\sqrt{E_h}}{2}, w_Q > -\frac{\sqrt{E_h}}{2}\right] = \left(1 - Q\left(\sqrt{\frac{E_h}{2N_0}}\right)\right)^2 \tag{6.152}$$

As from (6.134) $E_s = E_h/2$ and $\Gamma = E_s/N_0$, we get

$$P_e = 1 - P[\mathrm{C}] = 1 - P[\mathrm{C} \mid \mathbf{s}_1] = 2Q\left(\sqrt{\Gamma}\right)\left[1 - \tfrac{1}{2}Q\left(\sqrt{\Gamma}\right)\right] \tag{6.153}$$

For $\Gamma \gg 1$, the following approximations are valid:

$$P_e \simeq 2Q\left(\sqrt{\Gamma}\right) \tag{6.154}$$

and

$$P_{bit} \simeq Q\left(\sqrt{\Gamma}\right) \tag{6.155}$$

The QPSK transmitter is obtained by simplification of the general scheme (6.130), as illustrated in Figure 6.34. The binary bit maps are given in Table 6.2. The ML receiver for QPSK is illustrated in Figure 6.35. As the decision thresholds are set at $(0, \pi/2, \pi, 3\pi/2)$

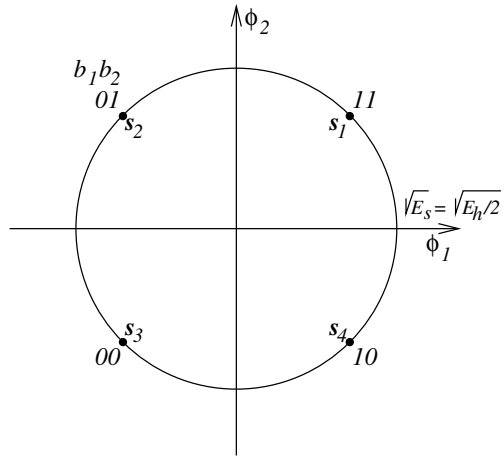


Figure 6.33. Signal constellation of a QPSK system.

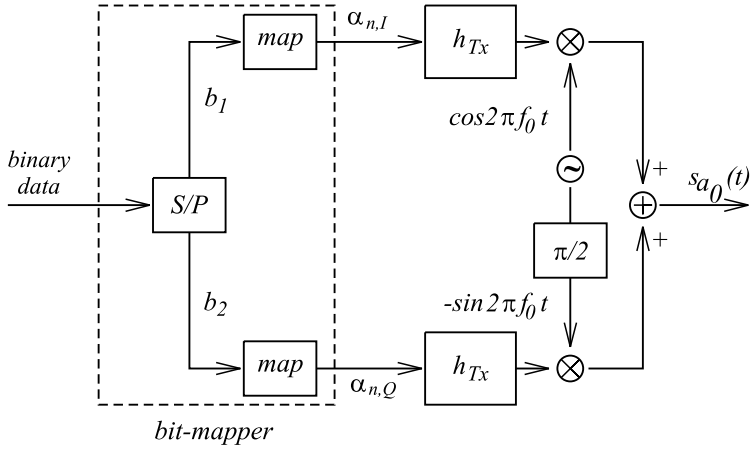


Figure 6.34. QPSK transmitter for an isolated pulse.

Table 6.2 Binary bit map for a QPSK system.

Binary bit map	
b_1 (b_2)	$\alpha_{n,I}$ ($\alpha_{n,Q}$)
0	$-1/\sqrt{2}$
1	$1/\sqrt{2}$

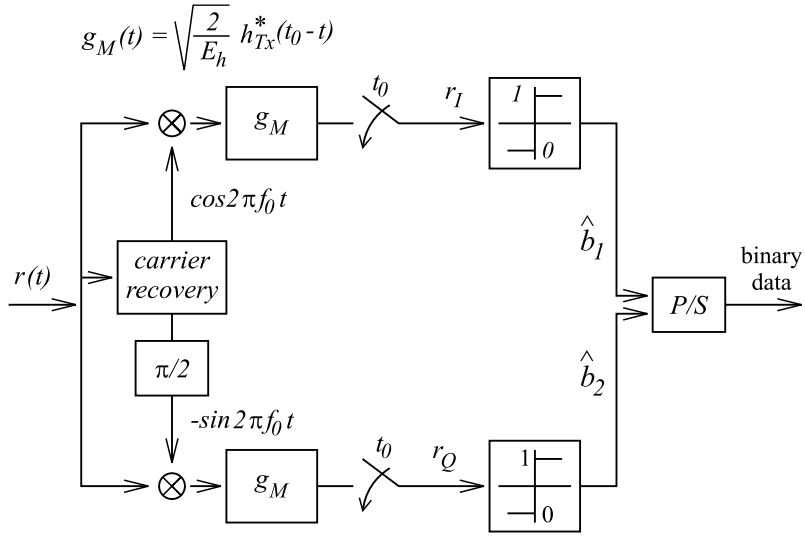


Figure 6.35. ML receiver for a QPSK system.

(see Figure 6.33), decisions can be made independently on r_I and r_Q, using a simple threshold detector with threshold set at zero.

We observe that, for $h_{Tx}(t) = K\mathrm{w}_T(t)$, the transmitter filter is a simple *holder*. At the receiver the matched filter plus sampler becomes an integrator that is cleared before each integration over a symbol period of duration T. In other words, it consists of an *integrate-and-dump*.

6.5 Differential PSK (DPSK)

We assume now that the receiver recovers the carrier signal, except for a phase offset φ_a. In particular, with reference to the scheme of Figure 6.28, the reconstructed carrier is $\cos(2\pi f_0 t - \varphi_a)$. In this case $\boldsymbol{s}_n$ coincides with $\sqrt{E_s}\, e^{j\varphi_a}\, \alpha_n$, where α_n is given by (6.129). Consequently, it is as if the constellation at the receiver were rotated by φ_a. To prevent this problem there are two strategies. By the *coherent method*, a receiver estimates φ_a from the received signal, and considers the original constellation for detection, using the signal $\boldsymbol{r}\, e^{-j\hat{\varphi}_a}$, where $\hat{\varphi}_a$ is the estimate of φ_a. By the *differential non-coherent method*, a receiver detects the data using the difference between the phases of signals at successive sampling instants. In other words

- for M-PSK, the phase of the transmitted signal at instant kT is given by (6.126), with

$$\theta_k \in \left\{ \frac{\pi}{M}, \frac{3\pi}{M}, \ldots, \frac{(2M-1)\pi}{M} \right\} \tag{6.156}$$

- for M-DPSK,[12] the transmitted phase at instant kT is given by

$$\psi'_k = \psi'_{k-1} + \theta_k \qquad \theta_k \in \left\{0, \frac{2\pi}{M}, \ldots, \frac{2\pi}{M}(M-1)\right\} \tag{6.157}$$

  that is, the phase associated with the transmitted signal at instant kT is equal to that transmitted at the previous instant $(k-1)T$ plus the increment θ_k, which can assume one of M values. We note that the decision thresholds for θ_k are now placed at $(\pi/M)(2n-1)$, $n = 1, \ldots, M$.

For a phase offset equal to φ_a introduced by the channel, the phase of the signal at the detection point becomes

$$\psi_k = \psi'_k + \varphi_a \tag{6.158}$$

In any case,

$$\psi_k - \psi_{k-1} = \theta_k \tag{6.159}$$

and the ambiguity of φ_a is removed. For phase-modulated signals, three differential non-coherent receivers that determine an estimate of (6.159) are discussed in Chapter 18.

6.5.1 Error probability for an M-DPSK system

For $E_s/N_0 \gg 1$, using the definition of the Marcum function $Q_1(\cdot,\cdot)$ (see Appendix 6.A) it can be shown that the error probability of an isolated symbol is approximated by the following bound [2, 3]

$$\begin{aligned} P_e \lesssim\ & 1 + Q_1\left(\sqrt{\frac{E_s}{N_0}\left(1 - \sin\frac{\pi}{M}\right)},\ \sqrt{\frac{E_s}{N_0}\left(1 + \sin\frac{\pi}{M}\right)}\right) \\ & - Q_1\left(\sqrt{\frac{E_s}{N_0}\left(1 + \sin\frac{\pi}{M}\right)},\ \sqrt{\frac{E_s}{N_0}\left(1 - \sin\frac{\pi}{M}\right)}\right) \end{aligned} \tag{6.160}$$

Moreover, if M is large, the approximation (6.369) can be used and we get

$$\begin{aligned} P_e &\simeq 2Q\left[\sqrt{\frac{E_s}{N_0}}\left(\sqrt{1 + \sin\frac{\pi}{M}} - \sqrt{1 - \sin\frac{\pi}{M}}\right)\right] \\ &\simeq 2Q\left(\sqrt{\frac{E_s}{N_0}}\sin\frac{\pi}{M}\right) \end{aligned} \tag{6.161}$$

[12] Note that we consider a differential non-coherent receiver with which is associated a differential symbol encoder at the transmitter (see (6.157)) or ((6.169)). However, as we will see in the next section, a differential encoder and a coherent receiver can be used.

For Gray coding of the values of θ_k in (6.156), the bit error probability is given by

$$P_{bit} = \frac{P_e}{\log_2 M} \tag{6.162}$$

where P_e is given by (6.161).

For $M = 2$, the exact formula of the error probability is [2, 3]

$$P_{bit} = P_e = \tfrac{1}{2} e^{-\frac{E_s}{N_0}} \tag{6.163}$$

For $M = 4$, the exact formula is [2, 3]

$$P_e = 2Q_1(a, b) - I_0(ab)\, e^{-0.5(a^2+b^2)} \tag{6.164}$$

where

$$a = \sqrt{\frac{E_s}{N_0}(1 - \sqrt{1/2})} \qquad b = \sqrt{\frac{E_s}{N_0}(1 + \sqrt{1/2})} \tag{6.165}$$

and where the function I_0 is defined in (4.216).

Using the previous results, a comparison in terms of P_{bit} between DPSK (6.161) and PSK (6.147) is given in Figure 6.36: we note that, for $P_{bit} = 10^{-3}$, DPSK presents a loss of only 1.2 dB in Γ for $M = 2$, that increases to 2.3 dB for $M = 4$, and to 3 dB for $M > 4$.

As a DPSK receiver is simpler as compared to a coherent PSK receiver, in that it does not require recovery of the carrier phase, for $M = 2$ DPSK is usually preferred to PSK.

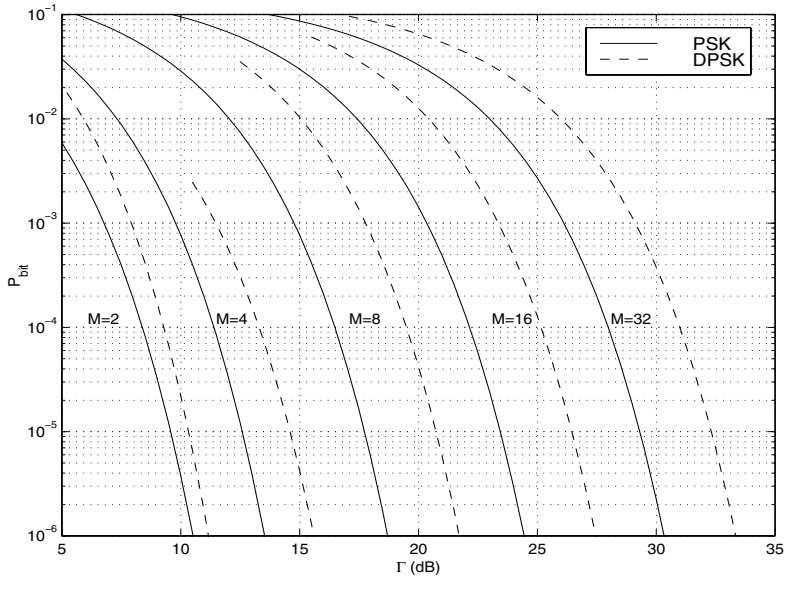


Figure 6.36. Comparison between PSK and DPSK.

Note that, if the previously received sample is used as a reference, DPSK gives lower performance with respect to PSK, especially for $M \geq 4$, because both the current sample and the reference sample are corrupted by noise. This drawback can be mitigated if the reference sample is constructed by using more than one previously received samples [4]. In this way we establish a gradual transition between differential phase demodulation and coherent demodulation. In particular, if the reference sample is constructed using the samples received in the two previous modulation intervals, DPSK and PSK yield similar performance [4].

6.5.2 Differential encoding and coherent demodulation

If φ_a is a multiple of $2\pi/M$, at the receiver the phase difference can be formed between the phases of two consecutive coherently detected symbols, instead of between the phases of two consecutive samples. In this case, symbols are differentially encoded before modulation.

Binary case ($M = 2$, differentially encoded BPSK)

Let b_k be the value of the information bit at instant kT, $b_k \in \{0, 1\}$.

BPSK system without differential encoding. The phase $\theta_k \in \{0, \pi\}$ is associated with b_k by the bit map of Table 6.3.

Differential encoder. For any $c_{-1} \in \{0, 1\}$, we encode the information bits as

$$c_k = c_{k-1} \oplus b_k \qquad b_k \in \{0, 1\} \qquad k \geq 0 \tag{6.166}$$

where $\oplus$ denotes the modulo 2 sum; therefore $c_k = c_{k-1}$ if $b_k = 0$, and[13] $c_k = \bar{c}_{k-1}$ if $b_k = 1$. For the bit map of Table 6.4 we have that $b_k = 1$ causes a phase transition, and $b_k = 0$ causes a phase repetition.

Decoder. If $\{\hat{c}_k\}$ are the detected coded bits at the receiver, the information bits are recovered by

$$\hat{b}_k = \hat{c}_k \oplus (-\hat{c}_{k-1}) = \hat{c}_k \oplus \hat{c}_{k-1} \tag{6.167}$$

Table 6.3 Bit map for a BPSK system.

b_k	Transmitted phase θ_k (rad)
0	0
1	π

[13] $\bar{c}$ denotes the one's complement of c: $\bar{1} = 0$ and $\bar{0} = 1$.

Table 6.4 Bit map for a differentially encoded BPSK system.

c_k	Transmitted phase ψ_k (rad)
0	0
1	π

We note that a phase ambiguity $\varphi_a = \pi$ does not alter the recovered sequence $\{\hat{b}_k\}$: in fact, in this case $\{\hat{c}_k\}$ becomes $\{\hat{c}'_k = \hat{c}_k \oplus 1\}$ and we have

$$(\hat{c}_k \oplus 1) \oplus (\hat{c}_{k-1} \oplus 1) = \hat{c}_k \oplus \hat{c}_{k-1} = \hat{b}_k \tag{6.168}$$

Multilevel case

Let $\{d_k\}$ be a multilevel information sequence, with $d_k \in \{0, 1, \ldots, M-1\}$. In this case we have

$$c_k = c_{k-1} \underset{M}{\oplus} d_k \tag{6.169}$$

where $\underset{M}{\oplus}$ denotes the modulo M sum. Because $c_k \in \{0, 1, \ldots, M-1\}$, the phase associated with the bit map is $\psi_k \in \{\pi/M, 3\pi/M, \ldots, (2M-1)\pi/M\}$. This encoding and bit-mapping scheme are equivalent to (6.157).

At the receiver the information sequence is recovered by

$$\hat{d}_k = \hat{c}_k \underset{M}{\oplus} (-\hat{c}_{k-1}) \tag{6.170}$$

It is easy to see that an offset equal to $j \in \{0, 1, \ldots, (M-1)\}$ in the sequence $\{\hat{c}_k\}$, corresponding to a phase offset equal to $\{0, 2\pi/M, \ldots, (M-1)2\pi/M\}$ in $\{\psi_k\}$, does not cause errors in $\{\hat{d}_k\}$. In fact,

$$\left(\hat{c}_k \underset{M}{\oplus} j\right) \underset{M}{\oplus} \left[-\left(\hat{c}_{k-1} \underset{M}{\oplus} j\right)\right] = \hat{c}_k \underset{M}{\oplus} (-\hat{c}_{k-1}) = \hat{d}_k \tag{6.171}$$

Performance of a PSK system with differential encoding and *coherent demodulation* by the scheme of Figure 6.28, is worse as compared to a system with absolute phase encoding. However, for small P_e, up to values of the order of 0.1, we observe that an error in $\{\hat{c}_k\}$ causes two errors in $\{\hat{d}_k\}$. Approximately, P_e increases by a factor 2,[14] which causes a negligible loss in terms of Γ.

To combine Gray encoding of values of c_k with the differential encoding (6.169), a two step procedure is adopted:

[14] If we indicate with $P_{e,Ch}$ the channel error probability, then the error probability after decoding is given by [2]

$$\textit{Binary case} \quad P_{bit} = 2P_{bit,Ch}\,[1 - P_{bit,Ch}] \tag{6.172}$$

$$\textit{Quaternary case} \quad P_e = 4P_{e,Ch} - 8P_{e,Ch}^2 + 8P_{e,Ch}^3 - 4P_{e,Ch}^4 \tag{6.173}$$

Table 6.5 Gray coding for $M = 8$.

Three information bits			values of d_k
0	0	0	0
0	0	1	1
0	1	1	2
0	1	0	3
1	1	0	4
1	1	1	5
1	0	1	6
1	0	0	7

1. represent the values of d_k with a Gray encoder using a combinatorial table, as illustrated for example in Table 6.5 for $M = 8$;
2. determine the differentially encoded symbols according to (6.169).

Example 6.5.1 (Differential encoding 2B1Q)
We consider a differential encoding scheme for a four-level system that makes the reception insensitive to a possible change of sign of the transmitted sequence. For $M = 4$ this implies insensitivity to a phase rotation equal to π in a 4-PSK signal or to a change of sign in a 4-PAM signal.

For $M = 4$ we give the law between the binary representation of $d_k = (d_k^{(1)}, d_k^{(0)})$, $d_k^{(i)} \in \{0, 1\}$, and the binary representation of $c_k = (c_k^{(1)}, c_k^{(0)})$, $c_k^{(i)} \in \{0, 1\}$:

$$\begin{aligned} c_k^{(1)} &= d_k^{(1)} \oplus c_{k-1}^{(1)} \\ c_k^{(0)} &= d_k^{(0)} \oplus c_k^{(1)} \end{aligned} \tag{6.174}$$

The bit map is given in Table 6.6.

The equations of the differential decoder are

$$\begin{aligned} \hat{d}_k^{(1)} &= \hat{c}_k^{(1)} \oplus \hat{c}_{k-1}^{(1)} \\ \hat{d}_k^{(0)} &= \hat{c}_k^{(0)} \oplus \hat{c}_k^{(1)} \end{aligned} \tag{6.175}$$

Table 6.6 Bit map for the differential encoder 2B1Q.

$c_k^{(1)}$	$c_k^{(0)}$	Transmitted symbol a_k
0	0	-3
0	1	-1
1	0	1
1	1	3

6.6 AM-PM or quadrature amplitude modulation (QAM)

Quadrature amplitude modulation is another example of passband modulation. Consider choosing a bit mapper that associates to a sequence of $\log_2 M$ bits a symbol from a constellation of cardinality M and elements given by the complex numbers

$$\alpha_n \qquad n = 1, 2, \ldots, M \tag{6.176}$$

If we modulate a symbol of the constellation by a real baseband pulse h_{Tx} with finite energy E_h and support $(0, t_0)$, we obtain the isolated generic transmitted pulse given by

$$s_n(t) = \alpha_{n,I} h_{Tx}(t) \cos(2\pi f_0 t) - \alpha_{n,Q} h_{Tx}(t) \sin(2\pi f_0 t) \qquad t \in \Re \qquad n = 1, \ldots, M \tag{6.177}$$

where $\alpha_{n,I}$ and $\alpha_{n,Q}$ denote the real and imaginary part of α_n, respectively. From (6.176) we also have

$$s_n(t) = Re[\alpha_n h_{Tx}(t) e^{j2\pi f_0 t}] \tag{6.178}$$

The expression (6.177) indicates that the transmitted signal is obtained by modulating in amplitude two carriers in quadrature. However, if the amplitudes $|\alpha_n|$, $n = 1, \ldots, M$, are not all equal, equation (6.178) suggests that the transmitted signals are obtained not only by varying the phase of the carrier but also the amplitude, hence the name *amplitude modulation-phase modulation* (AM-PM). In fact QAM may be regarded as an extension of PSK.

Energy of s_n: if f_0 is larger than the bandwidth of h_{Tx}, we have

$$E_n = |\alpha_n|^2 \frac{E_h}{2} \tag{6.179}$$

Average energy of the system:

$$E_s = \frac{1}{M} \sum_{n=1}^{M} E_n \tag{6.180}$$

For a rectangular constellation $M = L^2$, and

$$\alpha_{n_I,I}, \alpha_{n_Q,Q} \in [-(L-1), -(L-3), \ldots, -3, -1, 1, 3, \ldots, (L-1)] \tag{6.181}$$

Then

$$\begin{aligned} E_s &= \frac{2}{3}(L^2 - 1)\frac{E_h}{2} \\ &= \frac{M-1}{3} E_h \end{aligned} \tag{6.182}$$

hence

$$E_h = E_s \frac{3}{M-1} \tag{6.183}$$

Basis functions: basis functions for the signals defined in (6.177) are given by

$$\begin{aligned}\phi_1(t) &= \sqrt{\frac{2}{E_h}} h_{Tx}(t)\cos(2\pi f_0 t)\\ \phi_2(t) &= -\sqrt{\frac{2}{E_h}} h_{Tx}(t)\sin(2\pi f_0 t)\end{aligned} \tag{6.184}$$

Vector representation:

$$\mathbf{s}_n = \sqrt{\frac{E_h}{2}}[\alpha_{n,I}, \alpha_{n,Q}]^T \qquad n = 1, \ldots, M \tag{6.185}$$

as illustrated in Figure 6.37 for various values of M.

We note that, except for the factor $\sqrt{\frac{E_h}{2}}$, in a QAM system $\boldsymbol{s}_n$ coincides with α_n.

It is important to observe that for the signals in (6.185) the minimum distance between two symbols is equal to $\sqrt{2E_h}$, hence

$$d_{min} = \sqrt{2E_h} \tag{6.186}$$

Consequently, to maintain a given d_{min}, for every additional bit of information, that is doubling M, we need to increase the average energy of the system by about 3 dB, according to the law

$$d_{min}^2 = E_s \frac{6}{M-1} \tag{6.187}$$

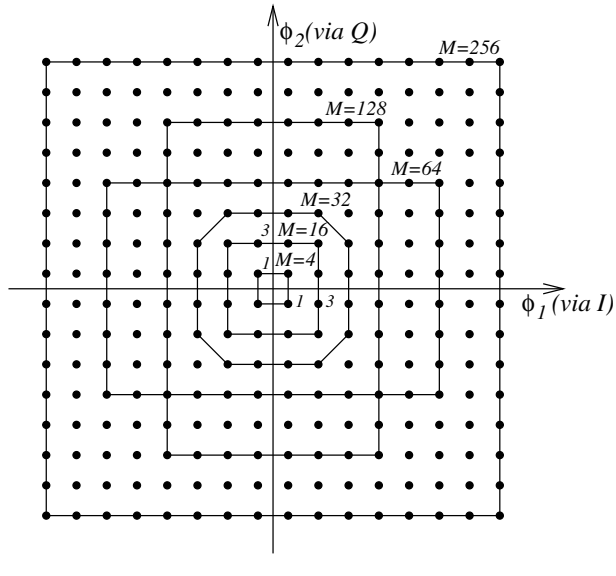


Figure 6.37. Signal constellations of M-QAM. The term $\sqrt{E_h/2}$ in (6.185) is normalized to one.

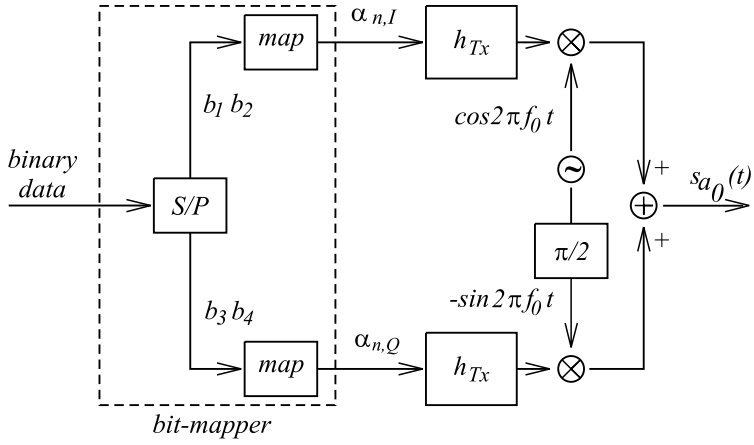


Figure 6.38. Transmitter of a 16-QAM system for an isolated pulse.

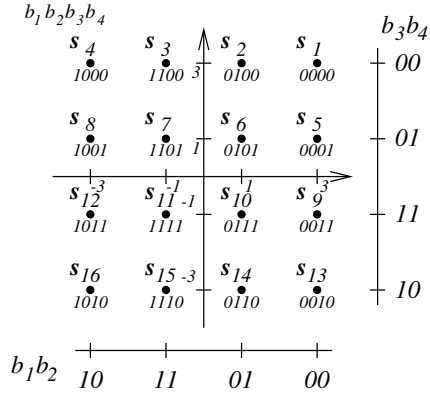


Figure 6.39. Signal constellation of a 16-QAM system.

The transmitter of an M-QAM system is illustrated in Figure 6.38 for $M = 16$. The bit map and the signal constellation of a 16-QAM system are shown in Figure 6.39. We note that the signals that are multiplied by the two carriers are PAM signals: in this example they are 4-PAM signals.

The ML receiver for a 16-QAM system is illustrated in Figure 6.40. We note that, as the 16-QAM constellation is rectangular, the decision regions are also rectangular and detection on the I and Q branches can be made independently by observing r_I and r_Q. In general, however, given $\mathbf{r} = [r_I, r_Q]^T$, we need to compute the M distances from the points $\mathbf{s}_n$, $n = 1, \ldots, M$, and choose the nearest to $\mathbf{r}$.

The following parameters of QAM systems are equal to those of PSK:

$$B_{min} = \frac{1}{T} \tag{6.188}$$

$$\nu = \log_2 M \tag{6.189}$$

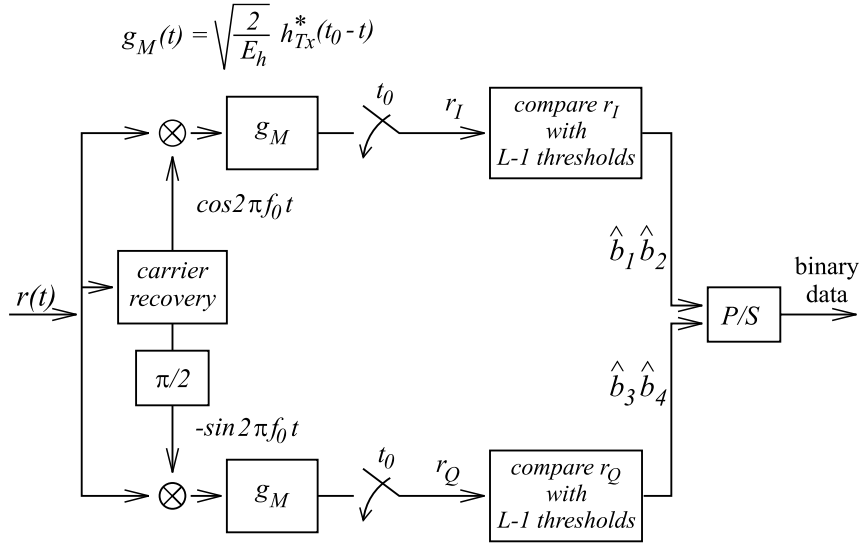


Figure 6.40. ML receiver for a 16-QAM system.

and

$$\Gamma = \frac{E_s}{N_0} \tag{6.190}$$

Moreover, we have

$$\Gamma_I = \Gamma \tag{6.191}$$

Symbol error probability for $M = L^2$, rectangular constellation. We first evaluate the probability of correct decision for a 16-QAM signal. We need to consider the following cases ($d = \sqrt{2E_h}$):

$$P[\mathrm{C} \mid \mathbf{s}_n] = \left[1 - Q\left(\frac{d}{2\sigma_I}\right)\right]^2 \qquad n = 1, 4, 13, 16 \tag{6.192}$$

$$P[\mathrm{C} \mid \mathbf{s}_n] = \left[1 - Q\left(\frac{d}{2\sigma_I}\right)\right]\left[1 - 2Q\left(\frac{d}{2\sigma_I}\right)\right] \qquad n = 2, 3, 5, 8, 9, 12, 14, 15 \tag{6.193}$$

$$P[\mathrm{C} \mid \mathbf{s}_n] = \left[1 - 2Q\left(\frac{d}{2\sigma_I}\right)\right]^2 \qquad n = 6, 7, 10, 11 \tag{6.194}$$

The probability of error is then given by

$$P_e = 1 - P[\mathrm{C}] = 3Q\left(\frac{d}{2\sigma_I}\right) - 2.25Q^2\left(\frac{d}{2\sigma_I}\right) \simeq 3Q\left(\frac{d}{2\sigma_I}\right) \tag{6.195}$$

where the last approximation is valid for large values of $d/(2\sigma_I)$.

In general, for a rectangular constellation with M elements, we get

$$P_e \simeq 4\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\frac{d}{2\sigma_I}\right) \tag{6.196}$$

Another expression can be found in terms of Γ using (6.186), (6.183), and (6.190),

$$P_e \simeq 4\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3}{(M-1)}\Gamma}\right) \tag{6.197}$$

The bit error probability is approximated as

$$P_{bit} \simeq \frac{P_e}{\log_2 M} \tag{6.198}$$

Curves of P_{bit} as a function of Γ are shown in Figure 6.41. We note that, to achieve a given P_{bit}, if M is increased by a factor 4, we need to increase Γ by 6 dB: in other words, if we increase by one the number of bits per symbol, on average we need an increase of the energy of the system of 3 dB. We arrived at the same result using the notion of d_{min} in (6.187).

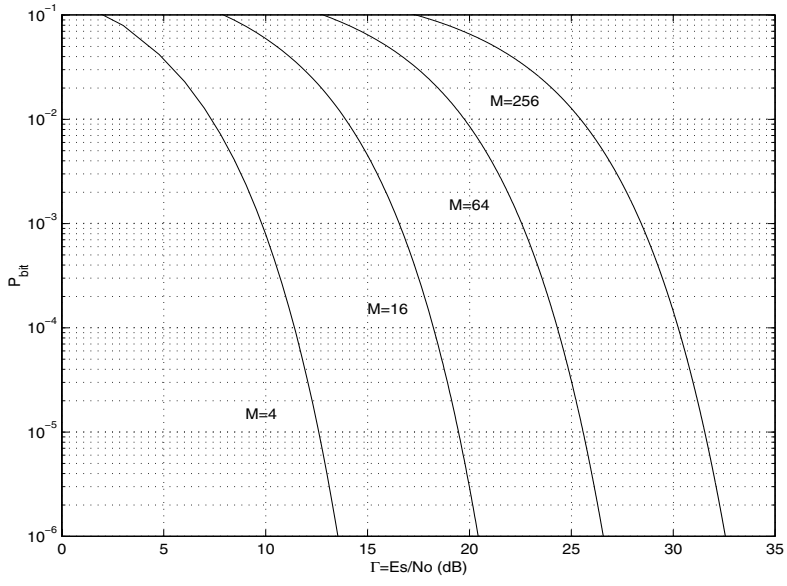


Figure 6.41. Bit error probability as a function of Γ for M-QAM transmission with rectangular constellation.

Comparison between PSK and QAM

A comparison between the performance of the two modulation systems is shown in Figure 6.42. For given P_{bit} and M, the gain of QAM with respect to PSK is given in terms of Γ in Table 6.7, where only the argument of the Q function in the expression of P_{bit} is considered. In general, for given $M \geq 4$ and Γ, QAM yields a lower P_{bit}, while having the same spectral efficiency as PSK.

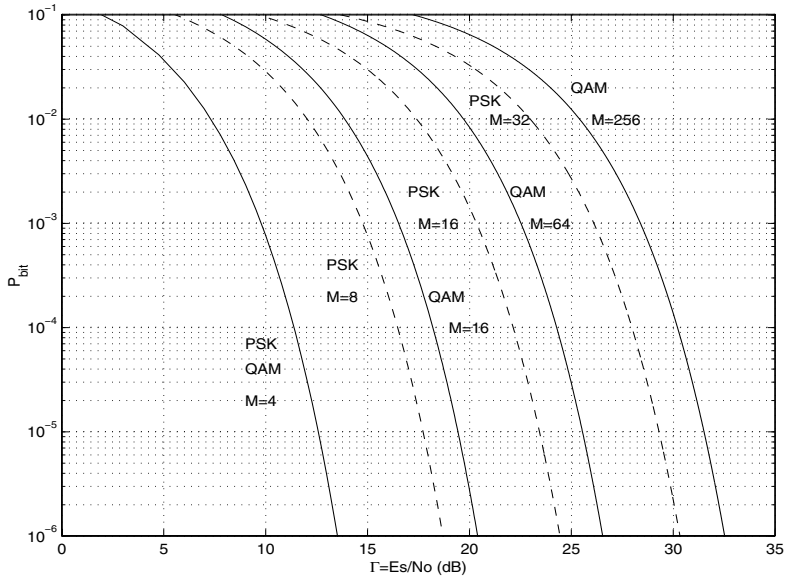


Figure 6.42. Comparison between PSK and QAM systems in terms of P_{bit} as a function of Γ.

Table 6.7 Gain of QAM with respect to PSK in terms of Γ, for given M.

M	$10\log_{10}\left(\frac{3/(M-1)}{2\sin^2(\pi/M)}\right)$ (dB)
4	0.00
8	1.65
16	4.20
32	7.02
64	9.95
128	12.92
256	15.92

6.7 Modulation methods using orthogonal and biorthogonal signals

6.7.1 Modulation with orthogonal signals

The isolated generic transmitted pulse, s_n, belongs to a set of M orthogonal signals with support $(0, t_0)$ and energy E_s, hence

$$\langle s_i, s_j \rangle = \int_0^{t_0} s_i(t) s_j^*(t)\, dt = E_s \delta_{i-j} \qquad i, j = 1, \ldots, M \tag{6.199}$$

A basis for these signals is simply given by the functions

$$\phi_n(t) = \frac{s_n(t)}{\sqrt{E_s}} \qquad n = 1, \ldots, M \tag{6.200}$$

The vector representations of sets of orthogonal signals for $M = 2$ and $M = 3$ are illustrated in Figure 6.43, where

$$\underset{1\ \ 2}{\boldsymbol{s}_n} = \sqrt{E_s}\ [\underset{n}{0}, 0, \ldots, 0, \underset{M}{1}, 0, \ldots, 0]^T \tag{6.201}$$

We note that the distance between any two signals is equal to $d_{min} = \sqrt{2E_s}$.

We will now consider a few examples of orthogonal signalling schemes.

Example 6.7.1 (Multilevel FSK)

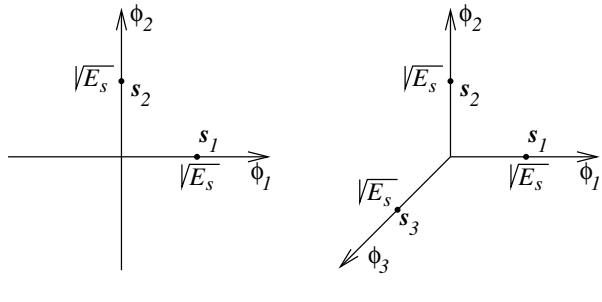
1. *Coherent*

$$s_n(t) = A \sin(2\pi f_n t + \varphi) \qquad 0 < t < T \qquad n = 1, \ldots, M \tag{6.202}$$

where the conditions

$$\begin{aligned} f_n - f_{n-1} &= \frac{1}{2T} \\ f_n + f_\ell &= k\,\frac{1}{T} \ \ (k \text{ integer}) \qquad \text{or else} \quad f_1 \gg \frac{1}{T} \end{aligned} \tag{6.203}$$

guarantee orthogonality among the signals.



(a) $M = 2$. (b) $M = 3$.

Figure 6.43. Vector representations of sets of orthogonal signals for (a) $M = 2$ and (b) $M = 3$.

2. *Non-coherent*

$$s_n(t) = A\sin(2\pi f_n t + \varphi_n) \qquad 0 < t < T \qquad n = 1, \ldots, M \tag{6.204}$$

where the conditions

$$\begin{aligned} f_n - f_{n-1} &= \frac{1}{T} \\ f_n + f_\ell &= k\,\frac{1}{T} \;\; (k \text{ integer}) \qquad \text{or else} \quad f_1 \gg \frac{1}{T} \end{aligned} \tag{6.205}$$

guarantee orthogonality among the signals. In (6.204) the uniform r.v. φ_n is introduced as each signal has an arbitrary phase.

We note that in both cases the bandwidth required by the passband modulation system is proportional to M. In the case of coherent demodulation, we use the definition

$$B_{min} = \frac{M}{2T} \tag{6.206}$$

Correspondingly, from (6.103) we have

$$\nu = \frac{2\log_2 M}{M} \tag{6.207}$$

and from (6.105)

$$\Gamma = \frac{2E_s}{N_0 M} \tag{6.208}$$

As $I = M$, we have

$$\Gamma_I = \Gamma \tag{6.209}$$

For non-coherent demodulation, we have $B_{min} = M/T$, $\nu = (\log_2 M)/M$, $\Gamma = E_s/(N_0 M)$, and $\Gamma_I = 2\Gamma$.

Example 6.7.2 (Binary modulation with orthogonal signals)
In case we use only two orthogonal signals out of L available signals to realize a binary modulation, for $B_{min} = L/(2T)$, the spectral efficiency is

$$\nu = \frac{2}{L} \tag{6.210}$$

The system is not efficient in exploiting the available bandwidth.

Example 6.7.3 (Code division modulation)
Let

$$\{p_{n,j}\} \qquad n = 1, \ldots, L \quad j = 0, \ldots, L-1 \tag{6.211}$$

be the Walsh code of length $L = 2^m$, determined as discussed in Appendix 6.D.

Choosing $M \leq L$, the signal s_n is given by

$$s_n(t) = \sum_{j=0}^{L-1} p_{n,j} \mathrm{w}_{T_c}(t - jT_c) \qquad n = 1, \ldots, M \quad 0 < t < LT_c = T \tag{6.212}$$

The plots of s_n, $n = 1, \ldots, L$, for $L = 8$ are shown in Figure 6.71.

Choosing $M = L$, we note that also in this case the required bandwidth is proportional to $1/T_c = M/T$, and we adopt the definition

$$B_{min} = \frac{M}{2T} \tag{6.213}$$

as the modulation is of the baseband type. Moreover, from (6.93)

$$R_I = \frac{\log_2 M}{M}$$

Correspondingly, $\nu = (2 \log_2 M)/M$, $\Gamma = (2E_s)/(N_0 M)$, and $\Gamma_I = \Gamma$.

Example 6.7.4 (Binary code division modulation)
We analyze in detail the previous example for $M = 2$ and $L > M$. In this case

$$\{s_n(t)\} \qquad n = 1, 2 \tag{6.214}$$

coincide with two Walsh signals of length L and duration $LT_c = T$. Setting

$$B_{min} = \frac{L}{2T} \tag{6.215}$$

from (6.93), (6.103), (6.105), and (6.106) we obtain, respectively,

$$R_I = \frac{1}{2} = 0.5 \tag{6.216}$$

$$\nu = \frac{1}{(L/2T)T} = \frac{2}{L} \tag{6.217}$$

$$\Gamma = \frac{2E_s}{N_0 L} \tag{6.218}$$

$$\Gamma_I = \frac{E_s}{N_0} \tag{6.219}$$

Example 6.7.5 (Binary orthogonal modulation with coding)
We have $M = 2$ as in the previous example. Now, however, the elements of the set

$$\{s_n(t)\} \qquad n = 1, 2 \tag{6.220}$$

are chosen as two Walsh signals of length L and duration $LT_c = 2T$. Redundancy is introduced because the behavior of $s_n(t)$ in the interval $(T, 2T)$ depends on the behavior in

the interval $(0, T)$. With reference to the interval $(0, 2T)$, the number of information bits is equal to $\log_2 M = 1$: consequently, as T is the modulation interval we have $L_b = 1/2$; note that we also have $I = 2$. Then, with respect to the binary code division modulation presented above, bandwidth and rate are halved,

$$B_{min} = \frac{L}{4T} \tag{6.221}$$

$$R_I = \frac{0.5}{2} = 0.25 \tag{6.222}$$

The other parameters are given by

$$\nu = \frac{0.25}{(L/4T)T} 2 = \frac{2}{L} \tag{6.223}$$

$$\Gamma = \frac{4E_s}{N_0 L} \tag{6.224}$$

and

$$\Gamma_I = \frac{E_s}{N_0} \tag{6.225}$$

We note that this case can be regarded as an example of a repetition code where the same symbol is repeated twice.

Probability of error

The ML receiver is given by the general scheme of Figure 6.8, where $I = M$ and ϕ_i is proportional to s_i according to (6.200). As the various signals have equal energy, the decision variables are given by

$$U_n = Re[\langle r, s_n \rangle] = Re\left[\int_0^{t_0} r(t) s_n^*(t)\, dt\right] \qquad n = 1, \ldots, M \tag{6.226}$$

Assuming the signal s_m is transmitted, we have

$$\begin{aligned} U_n &= E_s \delta_{n-m} + Re\left[\int_0^{t_0} w(t) s_n^*(t)\, dt\right] \\ &= E_s \delta_{n-m} + \sqrt{E_s} w_n \end{aligned} \tag{6.227}$$

where $w_n = Re[\langle w, \phi_n \rangle]$ is the n-th noise component. Then $\{U_n\}$, $n = 1, \ldots, M$, are Gaussian r.v.s with mean

$$\mathrm{m}_{U_n} = E[U_n] = E_s \delta_{n-m} \tag{6.228}$$

and cross-covariance

$$E[(U_n - \mathrm{m}_{U_n})(U_\ell - \mathrm{m}_{U_\ell})] = \frac{N_0}{2} E_s \delta_{\ell - n} \tag{6.229}$$

Hence, the r.v.s $\{U_n\}$ are statistically independent with variance $E_s N_0/2$.

The probability of correct decision, conditioned on s_m, is equal to

$$\begin{aligned} P[\complement \mid s_m] &= P[U_m > U_1, \ldots, U_m > U_{m-1}, U_m > U_{m+1}, \ldots, U_m > U_M] \\ &= \int_{-\infty}^{+\infty} p_{U_m}(a) \left[\int_{-\infty}^{a} \cdots \int_{-\infty}^{a} p_{U_1}(b_1) \ldots p_{U_M}(b_M)\, db_1 \ldots db_M \right] da \\ &= \frac{1}{\sqrt{2\pi (E_s N_0/2)}} \int_{-\infty}^{+\infty} e^{-\frac{1}{2}\frac{(a-E_s)^2}{E_s(N_0/2)}} \left[\frac{1}{\sqrt{2\pi (E_s N_0/2)}} \int_{-\infty}^{a} e^{-\frac{1}{2}\frac{b^2}{E_s(N_0/2)}} db \right]^{M-1} da \end{aligned} \tag{6.230}$$

With the change of variables

$$\alpha = \frac{a}{\sqrt{E_s N_0/2}} \qquad \beta = \frac{b}{\sqrt{E_s N_0/2}} \tag{6.231}$$

it follows

$$P[\complement \mid s_m] = \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\alpha - \sqrt{\frac{2E_s}{N_0}}\right)^2} [1 - Q(\alpha)]^{M-1}\, d\alpha \tag{6.232}$$

We note that (6.232) is independent of s_m: consequently $P[\complement \mid s_m]$ is the same for each s_m. Therefore for equally likely signals we get

$$P[\complement] = P[\complement \mid s_m] \tag{6.233}$$

The error probability is given by

$$\begin{aligned} P_e &= 1 - P[\complement] \\ &= 1 - \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\alpha - \sqrt{\frac{2E_s}{N_0}}\right)^2} [1 - Q(\alpha)]^{M-1}\, d\alpha \end{aligned} \tag{6.234}$$

Let M be a power of 2: with each signal s_m we associate a binary representation, also called *character*, with $\log_2 M$ bits. Then a signal error occurs if a character different from the transmitted character is detected. This error event happens with probability P_e. For each bit of the transmitted character, among the possible $(M-1)$ wrong characters only $M/2$ yield a wrong bit. Therefore we have

$$\begin{aligned} P_{bit} &= \frac{M/2}{M-1} P_e \\ &\simeq \frac{1}{2} P_e \end{aligned} \tag{6.235}$$

for M sufficiently large.

Curves of P_{bit} as a function of $\Gamma = 2E_s/(N_0 M)$ and E_b/N_0 are given, respectively, in Figure 6.44 and Figure 6.45.[15] We note that, in contrast with QAM modulation, for a given P_{bit} Γ decreases as M increases. The drawback is an increase of the required bandwidth with increasing M.

[15] The computation of the integral (6.234) was carried out using the Hermite polynomial series expansion, as indicated in [5, page 294].

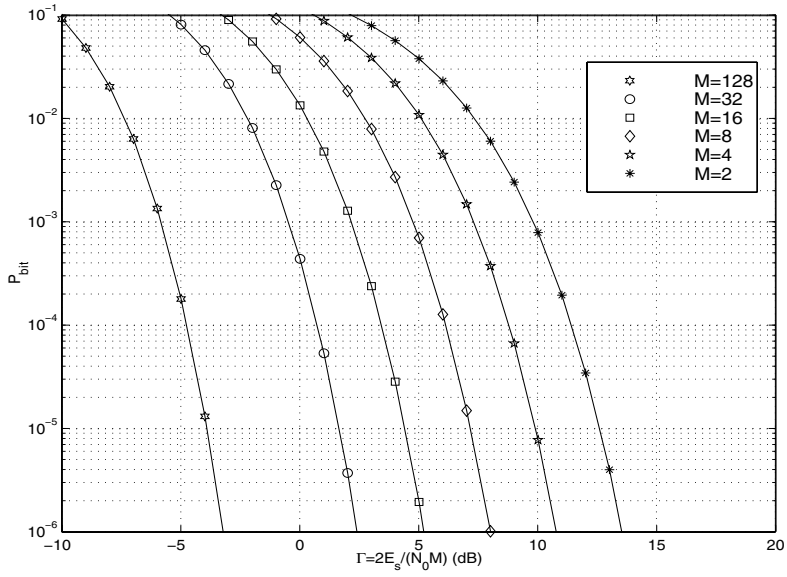


Figure 6.44. Bit error probability as a function of Γ for transmission with M orthogonal signals.

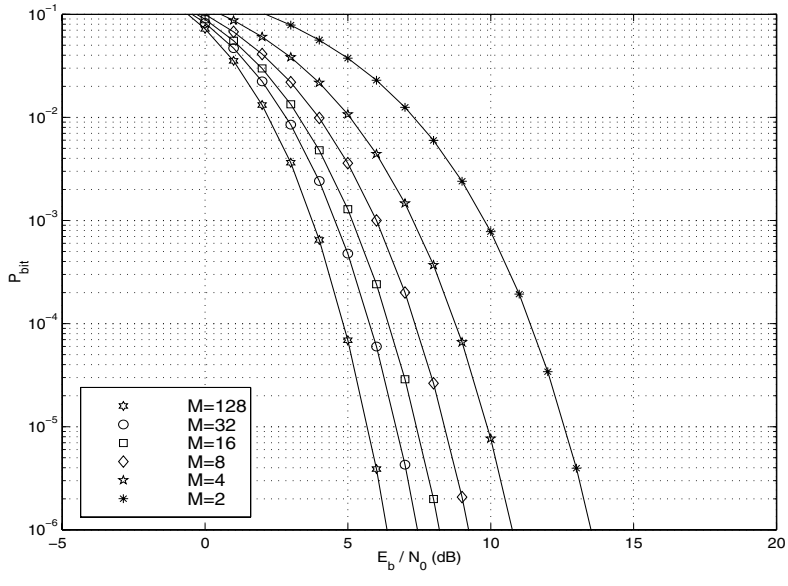


Figure 6.45. Bit error probability as a function of E_b/N_0 for transmission with M orthogonal signals.

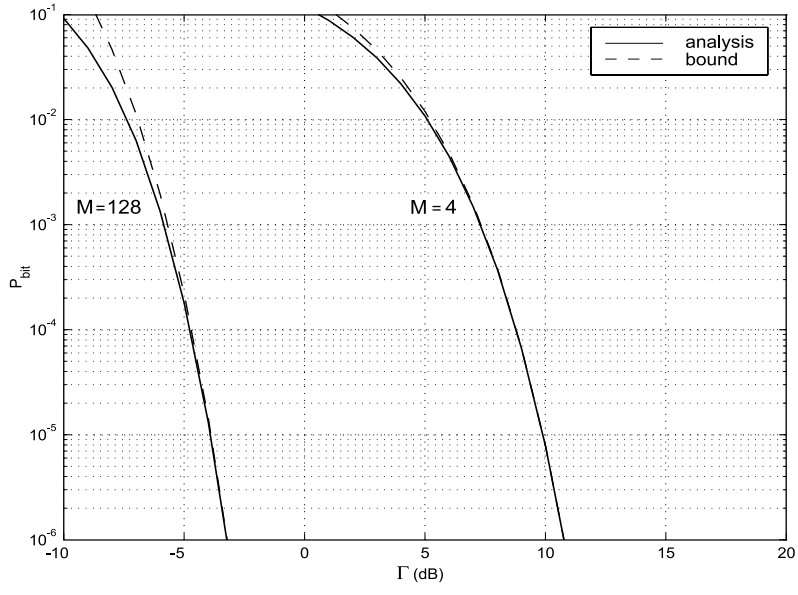


Figure 6.46. Comparison between the exact error probability and the limit (6.236) for transmission with M orthogonal signals.

Exploiting the bound (6.84), a useful approximation of P_{bit} is given by

$$P_{bit} \leq \frac{M}{2} Q\left(\sqrt{\frac{E_s}{N_0}}\right) \tag{6.236}$$

Figure 6.46 shows a comparison between the error probability obtained by exact computation and the bound (6.236) for two values of M.

Limit of the probability of error for M increasing to infinity

We give in Table 6.8 the values of E_b/N_0 needed to achieve $P_{bit} = 10^{-6}$, for various values of M. In fact we can show that

$$P_{bit} \xrightarrow[M\to\infty]{} 0 \tag{6.237}$$

only if the following condition is satisfied

$$\frac{E_b}{N_0} > -1.59\,\text{dB} \tag{6.238}$$

otherwise $P_{bit} \xrightarrow[M\to\infty]{} 1$. Therefore $-1.59\,\text{dB}$ is the minimum value of E_b/N_0 that is necessary to reach an error probability that can be made *arbitrarily small for* $M \to \infty$ (see Section 6.10).

Table 6.8 Values of E_b/N_0 required to obtain $P_{bit} = 10^{-6}$ for various values of M.

M	E_b/N_0 (dB)
2^3	9.4
2^4	8.3
2^5	7.5
2^6	7.0
2^{10}	5.4
2^{15}	4.5
2^{20}	3.9
$\vdots$	$\vdots$
∞	-1.59

6.7.2 Modulation with biorthogonal signals

The elements of a set of M biorthogonal signals are $M/2$ orthogonal signals and their antipodal signals: for example, 4-PSK is a biorthogonal signalling scheme. A further example of biorthogonal signalling with $2M$ signals is given by a signalling scheme using the M orthogonal signals in (6.212) and their antipodal signals. For biorthogonal signalling with M signals, the required bandwidth is proportional to $M/2$.

We give the parameters of the system in the two cases of non-coherent and coherent demodulation.

Passband signalling with non-coherent demodulation:

$$B_{min} = \frac{M}{2}\frac{1}{T} \tag{6.239}$$

$$\nu = 2\frac{\log_2 M}{M} \tag{6.240}$$

$$\Gamma = \frac{2E_s}{N_0 M} \tag{6.241}$$

and, as $I = M/2$,

$$\Gamma_I = 2\Gamma \tag{6.242}$$

Baseband signalling or passband signalling with coherent demodulation:

$$B_{min} = \frac{M}{2}\frac{1}{2T} \tag{6.243}$$

$$\nu = 4\frac{\log_2 M}{M} \tag{6.244}$$

$$\Gamma = \frac{4E_s}{N_0 M} \tag{6.245}$$

and

$$\Gamma_I = \Gamma \tag{6.246}$$

Probability of error

The receiver consists of $M/2$ correlators, or *matched filters*, which provide the decision variables

$$\{U_n\} \qquad n = 1, \ldots, \frac{M}{2} \tag{6.247}$$

The optimum receiver selects the output with the largest absolute value, $|U_i|$; subsequently it selects s_i or $-s_i$ depending on the sign of U_i.

To compute the probability of correct decision, we proceed as in the previous case.

Assuming that s_m is taken as one of the signals of the basis, then

$$\begin{aligned} P[\complement \mid s_m] &= P[U_m > 0, |U_m| > |U_1|, \ldots, |U_m| > |U_{m-1}|, |U_m| \\ &\quad > |U_{m+1}|, \ldots, |U_m| > |U_{M/2}|] \\ &= \int_0^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}\left(\alpha - \sqrt{\frac{2E_s}{N_0}}\right)^2} [1 - 2Q(\alpha)]^{M/2-1}\, d\alpha \end{aligned} \tag{6.248}$$

The symbol error probability is given by

$$P_e = 1 - P[\complement \mid s_m] \tag{6.249}$$

The bit error probability can be approximated as

$$P_{bit} \simeq \tfrac{1}{2} P_e \tag{6.250}$$

Curves of P_{bit} as a function of $\Gamma = 4E_s/(N_0 M)$ and E_b/N_0 are plotted, respectively, in Figure 6.47 and in Figure 6.48, for various values of M.

A bound to (6.249) for transmission with M biorthogonal signals is given by

$$P_e \leq (M-2) Q\left(\sqrt{\frac{E_s}{N_0}}\right) + Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \tag{6.251}$$

where the first term arises from the comparison with $(M-2)$ orthogonal signals, and the second arises from the comparison with an antipodal signal.

Figure 6.49 shows a comparison between the error probability obtained by exact computation and the bound (6.251) for two values of M.

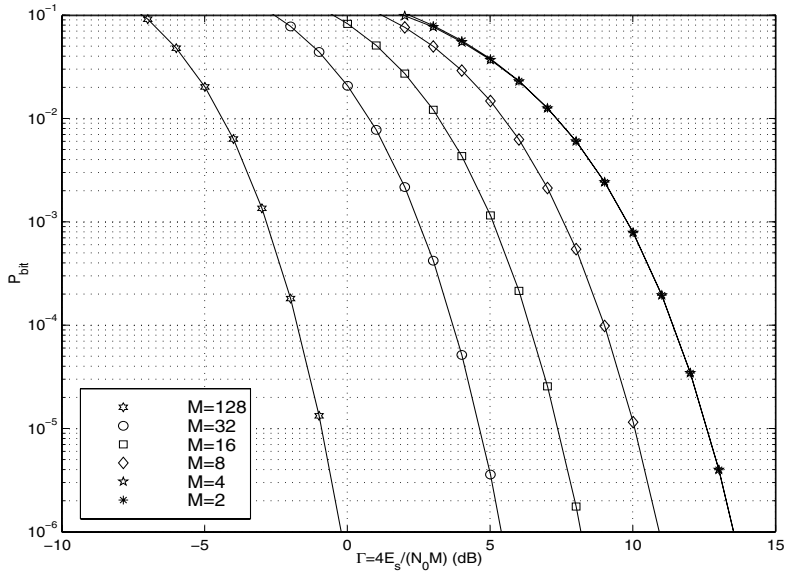


Figure 6.47. Bit error probability as a function of Γ for transmission with M biorthogonal signals.

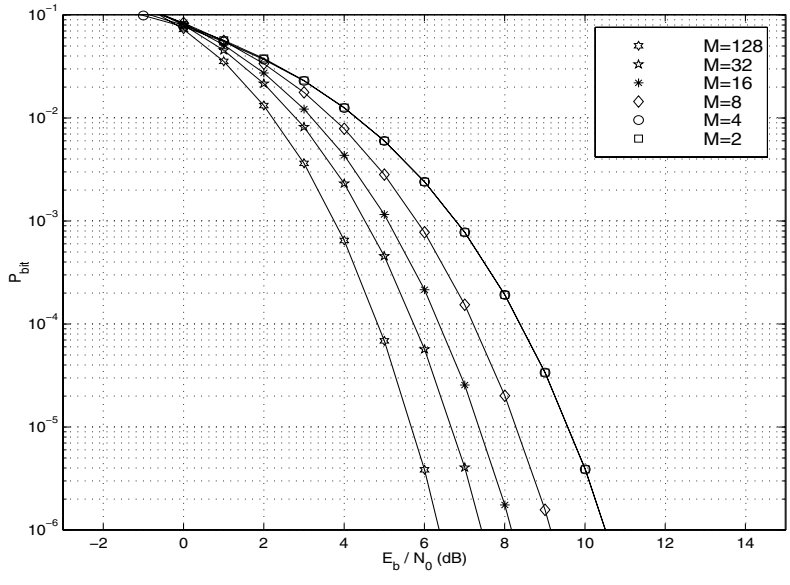


Figure 6.48. Bit error probability as a function of E_b/N_0 for transmission with M biorthogonal signals.

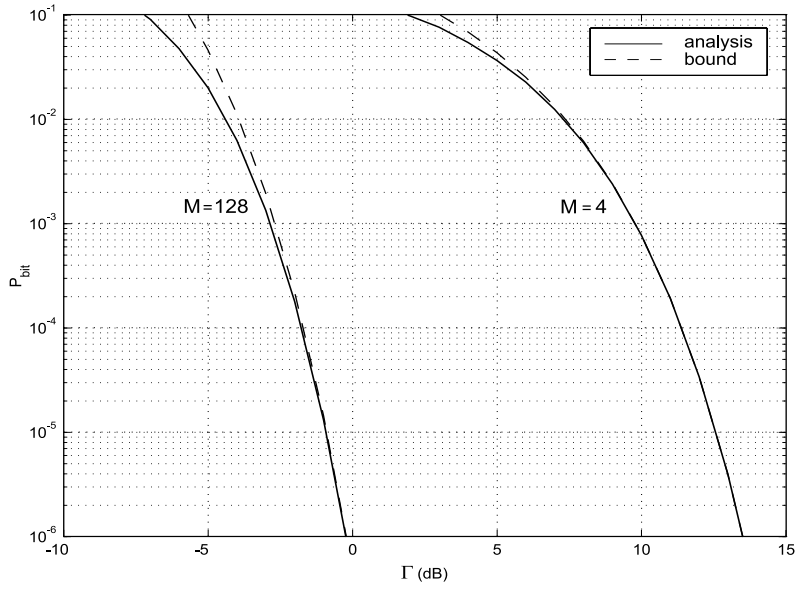


Figure 6.49. Comparison between the exact error probability and the limit (6.251) for transmission with M biorthogonal signals.

6.8 Binary sequences and coding

We consider a baseband signalling scheme where the transmitted signal is given by

$$s_n(t) = \sqrt{E_{\mathrm{w}}} \sum_{j=0}^{n_0-1} c_{n,j}\, \tilde{\mathrm{w}}_T(t - jT) \qquad n = 1, \ldots, M \quad 0 < t < n_0 T = T_s \tag{6.252}$$

where $c_{n,j} \in \{-1, +1\}$, and $\tilde{\mathrm{w}}_T(t) = \frac{1}{\sqrt{T}}\, rect\, \frac{t-T/2}{T}$ is the normalized rectangular window of duration T (see (1.456)) with unit energy. Then E_{w} is the energy of the pulse s_n *evaluated on a generic subperiod* T. Moreover, we have $B_{min} = \frac{1}{2T}$.

Interpreting the n_0 pulses

$$\tilde{\mathrm{w}}_T(t), \ldots, \tilde{\mathrm{w}}_T(t - (n_0 - 1)T) \tag{6.253}$$

as elements of an orthonormal basis, we derive the structure of the optimum receiver.

Uncoded sequences. Every sequence of n_0 binary coefficients

$$\mathbf{c}_n = [c_{n,0}, \ldots, c_{n,n_0-1}]^T \qquad c_{n,j} \in \{-1, 1\} \tag{6.254}$$

is allowed, hence $M = 2^{n_0}$. For a modulation interval T_s, we have $L_b = \log_2 M = n_0$. As $I = n_0$, it follows

$$R_I = \frac{\log_2 M}{I} = 1 \tag{6.255}$$

$$E_s = n_0 \, E_{\mathrm{w}} \tag{6.256}$$

$$E_I = \frac{E_s}{I} = E_{\mathrm{w}} \tag{6.257}$$

$$E_b = \frac{E_I}{R_I} = E_{\mathrm{w}} \tag{6.258}$$

$$\Gamma_I = \frac{E_I}{N_0/2} = \frac{2E_{\mathrm{w}}}{N_0} = \frac{2E_b}{N_0} \tag{6.259}$$

$$\Gamma = \frac{E_s}{N_0 \dfrac{1}{2T} T_s} = \frac{2E_{\mathrm{w}}}{N_0} = \Gamma_I \tag{6.260}$$

Moreover, the minimum distance between two elements of the set of signals (6.252) is equal to $d_{min} = \sqrt{4E_{\mathrm{w}}}$. The error probability is determined by the ratio (6.57)

$$\gamma_u = \frac{d_{min}^2}{(2\sigma_I)^2} = \frac{d_{min}^2}{2N_0} = \frac{2E_{\mathrm{w}}}{N_0} = \frac{2E_b}{N_0} \tag{6.261}$$

where in the last step equation (6.258) is used.

Coded sequences. We consider a set of signals (6.252) corresponding to $M = 2^{k_0}$ binary sequences $\mathbf{c}_n$ with n_0 components, assuming that only k_0 components in (6.254), as for example those with index $j = 0, 1, \ldots, k_0 - 1$, can assume values in $\{-1, 1\}$ arbitrarily: these components determine, through appropriate binary functions, also the remaining $n_0 - k_0$ components. Because the number of elements of the basis is always $I = n_0$, we have

$$R_I = \frac{\log_2 M}{I} = \frac{k_0}{n_0} \tag{6.262}$$

$$E_s = n_0 \, E_{\mathrm{w}} \tag{6.263}$$

$$E_I = \frac{n_0 \, E_{\mathrm{w}}}{I} = E_{\mathrm{w}} \tag{6.264}$$

$$E_b = \frac{E_I}{R_I} = \frac{n_0}{k_0} \, E_{\mathrm{w}} \tag{6.265}$$

$$\Gamma_I = \frac{E_I}{N_0/2} = \frac{2E_{\mathrm{w}}}{N_0} = \frac{k_0}{n_0} \, \frac{2E_b}{N_0} \tag{6.266}$$

$$\Gamma = \frac{E_s}{N_0 \dfrac{1}{2T} T_s} = \Gamma_I \tag{6.267}$$

Indicating with d_{min}^H the minimum number of positions in which two vectors $\mathbf{c}_n$ differ, we find

$$d_{min}^2 = 4E_{\mathrm{w}} \, d_{min}^H \tag{6.268}$$

In the case of uncoded sequences $d_{min}^H = 1$. An example of coding is given by the choice of the following vectors (code sequences or code words) for $n_0 = 4$ and $k_0 = 2$,

$$\mathbf{c}_0 = \begin{bmatrix} -1 \\ -1 \\ -1 \\ -1 \end{bmatrix} \quad \mathbf{c}_1 = \begin{bmatrix} -1 \\ -1 \\ +1 \\ +1 \end{bmatrix} \quad \mathbf{c}_2 = \begin{bmatrix} +1 \\ +1 \\ -1 \\ -1 \end{bmatrix} \quad \mathbf{c}_3 = \begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \end{bmatrix} \tag{6.269}$$

For this signalling system, we have $d_{min}^H = 2$, and therefore $d_{min}^2 = 8E_{\mathrm{w}}$. Using (6.265), the signal-to-noise ratio at the decision point is given by

$$\gamma_c = \frac{4d_{min}^H E_{\mathrm{w}}}{2N_0} = \frac{d_{min}^H R_I 2E_b}{N_0} \tag{6.270}$$

We note that for a given value of E_b/N_0 the coded system presents a larger γ, and consequently a lower bit error probability, if $d_{min}^H R_I > 1$.

We will discuss in Chapter 11 the design of codes that yield a large d_{min}^H for given values of the parameters n_0 and k_0. A drawback of these systems is represented by the reduction of the transmission bit rate R_b for a given modulation interval T_s; alternative coding methods will be examined in Chapter 12.

Optimum receiver

With reference to the implementation of Figure 6.8, as the elements of the orthonormal basis (6.253) are obtained by shifting the pulse $\tilde{w}_T(t)$, the optimum receiver can be simplified as illustrated in Figure 6.50, where the projections of the received signal $r(t)$ onto the

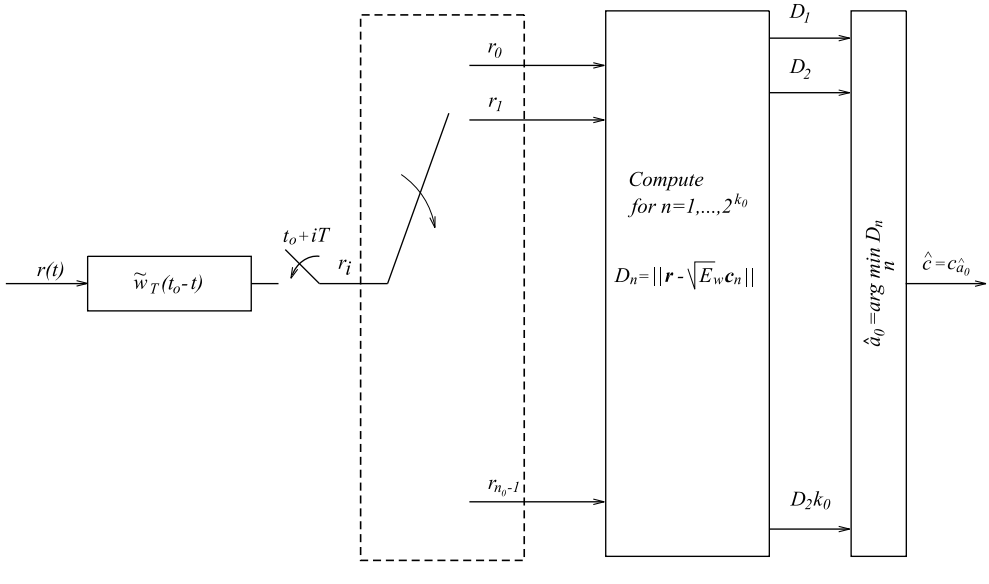


Figure 6.50. ML receiver for the signal set (6.252).

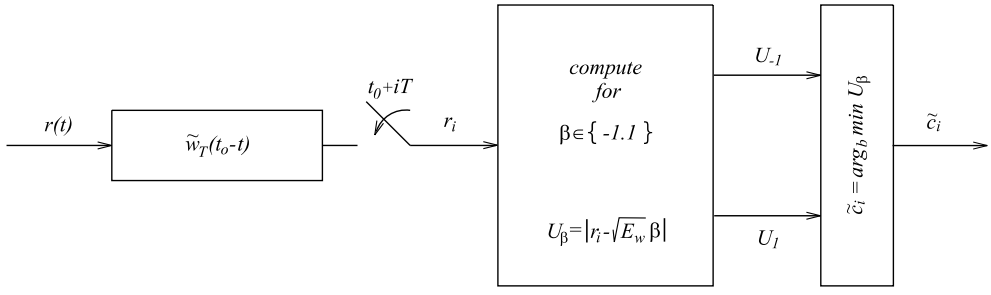


Figure 6.51. ML receiver for the signal set (6.252) under the assumption of uncoded sequences.

components of the basis (6.253) are obtained sequentially. The vector components $\mathbf{r} = [r_0, r_1, \ldots, r_{n_o-1}]^T$ are then used to compute the Euclidean distances with each of the possible code sequences. The scheme of Figure 6.50 yields the detected signal of the type (6.252), or equivalently the detected code sequence $\hat{\mathbf{c}} = [\hat{c}_0, \hat{c}_1, \ldots, \hat{c}_{n_o-1}]^T$, according to the ML criterion. This procedure is usually called *soft*-input decoding.

For the uncoded system, the receiver can be simplified by computing the Euclidean distance component by component, as illustrated in Figure 6.51. In the binary case under examination,

$$\hat{c}_i = \tilde{c}_i = \text{sgn}(r_i) \qquad i = 0, \ldots, n_0 - 1 \tag{6.271}$$

The resulting channel model (memoryless binary symmetric) is that of Figure 6.21.

In some receivers for coded systems, a simplification of the scheme of Figure 6.50 is obtained by first detecting the single components $\tilde{c}_i \in \{-1, 1\}$ according to the scheme of Figure 6.51. Successively, the binary vector $\tilde{\mathbf{c}} = [\tilde{c}_0, \ldots, \tilde{c}_{n_0-1}]^T$ is formed. Then we choose among the possible code sequences $\mathbf{c}_n$, $n = 1, \ldots, 2^{k_0}$, the one that differs in the smallest number of positions with respect to the sequence $\tilde{\mathbf{c}}$. This scheme is usually called *hard*-input decoding and is clearly suboptimum as compared to the scheme with *soft* input.

6.9 Comparison between coherent modulation methods

Table 6.9 summarizes some important results derived in the previous sections. Passband PAM is considered as *single sideband* (SSB) modulation or *double sideband* (DSB) modulation (see Appendix 7.C). In the latter case B_{min} is equal to $1/T$, hence $\Gamma = E_s/N_0$. We note that, for a given noise level, PAM, PAM+SSB and PAM+DSB methods require the same statistical power to achieve a certain P_e; however the PAM+DSB technique has a B_{min} that is double as compared to PAM or PAM+SSB methods.

For a given value of the symbol error probability, we now derive Γ_I as a function of R_I for some multilevel modulations. The result will be compared with the Shannon limit given by $\Gamma_I = 2^{2R_I} - 1$, that represents the *minimum theoretical value* of Γ_I, in correspondence of a given R_I, for which P_{bit} can be made arbitrarily small by using channel coding without constraints in complexity and latency (see Section 6.10). We note

Table 6.9 Comparison of various modulation methods in terms of performance, bandwidth, and spectral efficiency.

Modulation	Approximated symbol error probability P_e	Minimum bandwidth B_{min} (Hz)	Spectral efficiency $\nu = (1/T_b)/B_{min}$ (bit/s/Hz)	Encoder-modulator rate R_I (bit/dim)	Signal-to-noise ratio Γ
binary antipodal (BB)	$Q\left(\sqrt{\Gamma}\right)$	$\frac{1}{2T}$	2	1	$\frac{2E_s}{N_0}$
M-PAM	$\left(1-\frac{1}{M}\right)2Q\left(\sqrt{\frac{3}{M^2-1}\Gamma}\right)$	$\frac{1}{2T}$	$2\log_2 M$	$\log_2 M$	$\frac{2E_s}{N_0}$
M-PAM + SSB					
M-PAM + DSB	$\left(1-\frac{1}{M}\right)2Q\left(\sqrt{\frac{6}{M^2-1}\Gamma}\right)$	$\frac{1}{T}$	$\log_2 M$		$\frac{E_s}{N_0}$
M-QAM ($M = L^2$)	$\left(1-\frac{1}{\sqrt{M}}\right)4Q\left(\sqrt{\frac{3}{M-1}\Gamma}\right)$	$\frac{1}{T}$	$\log_2 M$	$\frac{1}{2}\log_2 M$	$\frac{E_s}{N_0}$
BPSK o 2-PSK	$Q\left(\sqrt{2\Gamma}\right)$	$\frac{1}{T}$	1	1	$\frac{E_s}{N_0}$
QPSK o 4-PSK	$2Q\left(\sqrt{2\sin^2\left(\frac{\pi}{M}\right)\Gamma}\right)$		2	1	
M-PSK ($M > 2$)			$\log_2 M$	$\frac{1}{2}\log_2 M$	
orthogonal (BB)	$(M-1)Q\left(\sqrt{\frac{M}{2}\Gamma}\right)$	$\frac{M}{2T}$	$2\frac{\log_2 M}{M}$	$\frac{1}{M}\log_2 M$	$\frac{2E_s}{N_0 M}$
biorthogonal (BB)	$(M-2)Q\left(\sqrt{\frac{M}{4}\Gamma}\right)+Q\left(\sqrt{\frac{M}{2}\Gamma}\right)$	$\frac{M}{4T}$	$4\frac{\log_2 M}{M}$	$\frac{2}{M}\log_2 M$	$\frac{4E_s}{N_0 M}$

that an equivalent approach often adopted in the literature is to give E_b/N_0, related to Γ_I through (6.107), as a function of ν, related to R_I through (6.104).

A first comparison is made by assuming the same symbol error probability, $P_e = 10^{-6}$, for all systems. As $Q(\sqrt{z_0}) = 10^{-6}$ implies $z_0 \simeq 22$, considering only the argument of the Q function in Table 6.9, we have the following results.

1. *M-PAM.* From

$$\frac{3}{M^2-1}\Gamma = z_0 \tag{6.272}$$

and

$$\Gamma_I = \Gamma \qquad R_I = \log_2 M \tag{6.273}$$

we obtain the following relation

$$\Gamma_I = \frac{z_0}{3}(2^{2R_I} - 1) \tag{6.274}$$

2. *M-QAM.* From

$$\frac{3}{M-1}\Gamma = z_0 \tag{6.275}$$

and

$$\Gamma_I = \Gamma \qquad R_I = \tfrac{1}{2}\log_2 M \tag{6.276}$$

we obtain

$$\Gamma_I = \frac{z_0}{3}(2^{2R_I} - 1) \tag{6.277}$$

We note that for QAM a certain R_I is obtained with a number of symbols equal to $M_{QAM} = 2^{2R_I}$, whereas for PAM the same efficiency is reached for $M_{PAM} = 2^{R_I} = \sqrt{M_{QAM}}$.

3. *M-PSK.* It turns out

$$\Gamma_I = \frac{z_0}{20}2^{4R_I} \tag{6.278}$$

Equation (6.278) holds for $M \geq 4$, and is obtained by approximating $\sin(\pi/M)$ with π/M, and π^2 with 10.

4. *Orthogonal modulation.* Using the approximation

$$P_e \simeq (M-1)\,Q\left(\sqrt{\frac{M}{2}\Gamma}\right) \tag{6.279}$$

we note that the multiplicative constant in front of the Q function cannot be ignored: therefore a closed-form analytical expression for Γ_I as a function of R_I for a given P_e cannot be found.

5. *Biorthogonal modulation.* The symbol error probability is approximately the same as that of orthogonal modulation for half the number of signals. Both R_I and ν are doubled.

We note that, for a given value of R_I, PAM and QAM require the same value of Γ_I, whereas PSK requires a much larger value of Γ_I.

An exact comparison is now made for a given bit error probability. Using the P_{bit} curves previously obtained, the behavior of R_I as a function of Γ_I for $P_{bit} = 10^{-6}$ is illustrated in Figure 6.52.

We observe that the required Γ_I is much larger than the minimum value obtained by the Shannon limit. As will be discussed in Section 6.10, the gap can be reduced by channel coding.

We also note that, for large R_I, PAM and QAM allow a lower Γ_I with respect to PSK; moreover, orthogonal and biorthogonal modulation operate with $R_I < 1$, and corresponding very small values of Γ_I.

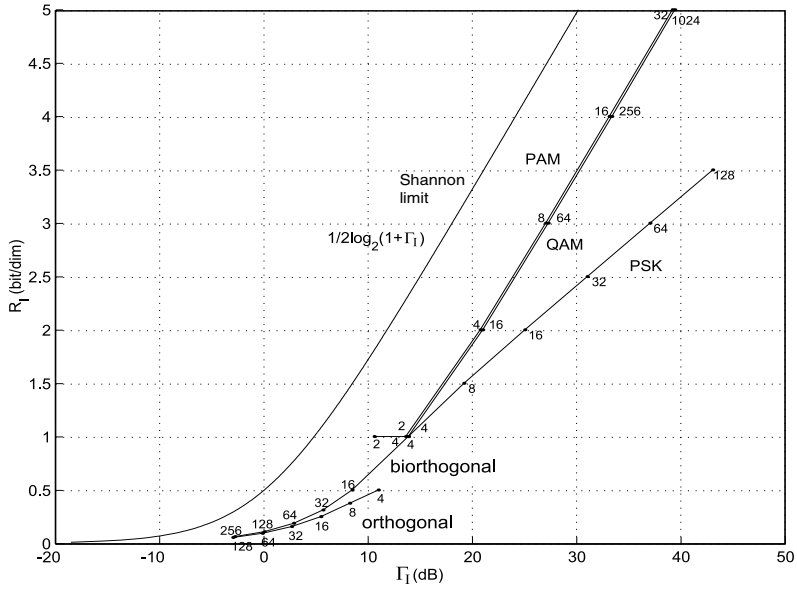


Figure 6.52. Γ_I required for a given rate R_I, for different modulation methods and bit error probability equal to $P_{bit} = 10^{-6}$. The parameter in the figure denotes the number of symbols M of the constellation.

Trade-offs for QAM systems

There are various trade-offs that are possible among the parameters of a modulation method. We consider for example Figure 6.41 for M-QAM, where the parameter is $\nu = \log_2 M = (1/T_b)/B_{min}$.

We assume that $1/T_b$ is fixed. For a given P_{bit}, we obtain Γ as a function of ν, from which the required bandwidth is also obtained; given ν (and the bandwidth), the trade-off is between P_{bit} and Γ; finally, fixed Γ, we get ν as a function of P_{bit}. We note that to modify ν a modulator with a different constellation must be adopted.

Comparison of modulation methods

PAM, QAM, and PSK are bandwidth efficient modulation methods as they cover the region for $R_I > 1$, or equivalently $\nu > 2$, or $B_{min} < 1/(2T_b)$, as illustrated in Figure 6.52. The bandwidth is traded off with the power, that is Γ, by increasing the number of levels: we note that, in this region, higher values of Γ are required to increase ν.

Orthogonal and biorthogonal modulation are not very efficient in bandwidth ($R_I < 1$), but require much lower values of Γ. As illustrated in Figure 6.52, biorthogonal modulation (see (6.249)) has the same performance as orthogonal modulation (see (6.234)), but requires half the bandwidth; in this region, by increasing the bandwidth it is possible to decrease Γ. However, a slight decrease in Γ may determine a large increase of the bandwidth. The P_{bit} of orthogonal or biorthogonal modulation is almost independent of M

and depends mainly on the energy E_s of the signal and on the spectral density $N_0/2$ of the noise.

In addition to the required power and bandwidth, the choice of a modulation scheme is based on the channel characteristics and on the cost of the implementation: until recently, for example, non-coherent receivers were preferred in radio mobile systems because of their simplicity, even though the performance is inferior to that of coherent receivers (see Chapter 18) [2].

6.10 Limits imposed by information theory

We consider the transmission of signals with a given power over an AWGN channel having noise power spectral density equal to $N_0/2$.

We recall the definition (6.93) of the encoder-modulator rate, $R_I = L_b/I$, where I is the number of signal space dimensions. For example, the encoder-modulator for the 8-PAM system with bit map defined in Table 6.1 has rate $R_I = 3$ (bit/dim), as $L_b = 3$ and $I = 1$. From (6.95), we have the cardinality of alphabet $\mathcal{A}$ is equal to $M = 2^{R_I} = 8$.

Let us consider for example a monodimensional transmission system (PAM) with an alphabet of cardinality $\mathcal{A}$ for a given rate R_I, such that $L_b < \log_2 M$, that is $M > 2^{R_I}$ from (6.93); the redundancy of the alphabet can be used to encode sequences of information bits: in this case we speak of *coded systems* (see Example 6.7.5). Let us take a PAM system with $R_I = 3$ and $M = 16$: redundancy may be introduced in the sequence of transmitted symbols. The mapping of sequences of information bits into sequences of coded output symbols may be described by a finite state sequential machine. Some specific examples will be illustrated in Chapter 12.

We recall the definition (1.135) of the passband $\mathcal{B}$ associated with the frequency response of a channel, with bandwidth given by (1.140), $B = \int_{\mathcal{B}} df$. *Channel capacity* is defined as the maximum of the *average mutual information* between the input and output signals of the channel [6, 7]. For transmission over an ideal AWGN channel, channel capacity is given in *bits per second* by

$$C_{[b/s]} = B \log_2(1 + \Gamma) \text{ (bit/s)} \tag{6.280}$$

where Γ is obtained from (6.105) by choosing $B_{min} = B$.

Equation (6.280) is a limit derived by Shannon assuming the transmitted signal $s(t)$ is a Gaussian random process with zero mean and constant power spectral density in the passband $\mathcal{B}$.

Using (6.280) and (6.103), we define the *maximum spectral efficiency* as

$$\nu_{max} = \frac{C_{[b/s]}}{B} = \log_2(1 + \Gamma) \text{ (bit/s/Hz)} \tag{6.281}$$

With reference to a message composed of a sequence of symbols, which belong to an I-dimensional space, the capacity can be expressed in *bits per dimension* as

$$C = \tfrac{1}{2} \log_2(1 + \Gamma_I) \text{ (bit/dim)} \tag{6.282}$$

obtained assuming a Gaussian distribution of the transmitted symbol sequence, where Γ_I is given by (6.106).

We give without proof the following fundamental theorem [8, 6].

Theorem 6.2 (Shannon's theorem)
For any rate $R_I < C$, there exists channel coding that allows transmission of information with an arbitrarily small probability of error; such coding does not exist if $R_I > C$.

We note that Shannon's theorem indicates the limits, in terms of encoder-modulator rate or, equivalently, in terms of *transmission bit rate* (see (6.280)), within which we can develop systems that allow reliable transmission of information, but it does not give any indication about the practical realization of channel coding.

The capacity can be upper limited and approximated for small values of Γ_I by a linear function, and also lower limited and approximated for large values of Γ_I by a logarithmic function as follows:

$$\Gamma_I \ll 1 : C \leq \tfrac{1}{2} \log_2(e)\, \Gamma_I \tag{6.283}$$

$$\Gamma_I \gg 1 : C \geq \tfrac{1}{2} \log_2(\Gamma_I) \tag{6.284}$$

Extension of the capacity formula for an AWGN channel to multi-input multi-output (MIMO) systems can be found in [9, 10].

Capacity of a system using amplitude modulation

Let us consider an M-PAM system with $M \geq 2$. The capacity of a real-valued AWGN channel having as input an M-PAM signal is given in bits per dimension by [11]

$$C = \max_{p_1,\ldots,p_M} \sum_{n=1}^{M} p_n \int_{-\infty}^{+\infty} p_{r|a_0}(\eta \mid \alpha_n) \log_2 \left[\frac{p_{r|a_0}(\eta \mid \alpha_n)}{\sum_{i=1}^{M} p_i\, p_{r|a_0}(\eta \mid \alpha_i)} \right] d\eta \tag{6.285}$$

where p_n indicates the probability of transmission of the symbol $a_0 = \alpha_n$. By the hypothesis of white Gaussian noise, we have

$$p_{r|a_0}(\eta \mid \alpha_n) \propto \exp\left\{ -\frac{(\eta - \alpha_n)^2}{2\sigma_I^2} \right\} \tag{6.286}$$

With the further hypothesis that only codes with equally likely symbols are of practical interest, the computation of the maximum of C with respect to the probability distribution of the input signal can be omitted. The channel capacity is therefore given by

$$\tilde{C} = \log_2 M - \frac{1}{M} \sum_{n=1}^{M} \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_I} e^{-\frac{\xi^2}{2\sigma_I^2}} \log_2 \left[\sum_{i=1}^{M} \exp\left(-\frac{(\alpha_n + \xi - \alpha_i)^2 - \xi^2}{2\sigma_I^2} \right) \right] d\xi \tag{6.287}$$

The capacity $\tilde{C}$ is illustrated in Figure 6.53, where the Shannon limit given by (6.282), as well as the signal-to-noise ratio given by (6.124) for which a symbol error probability

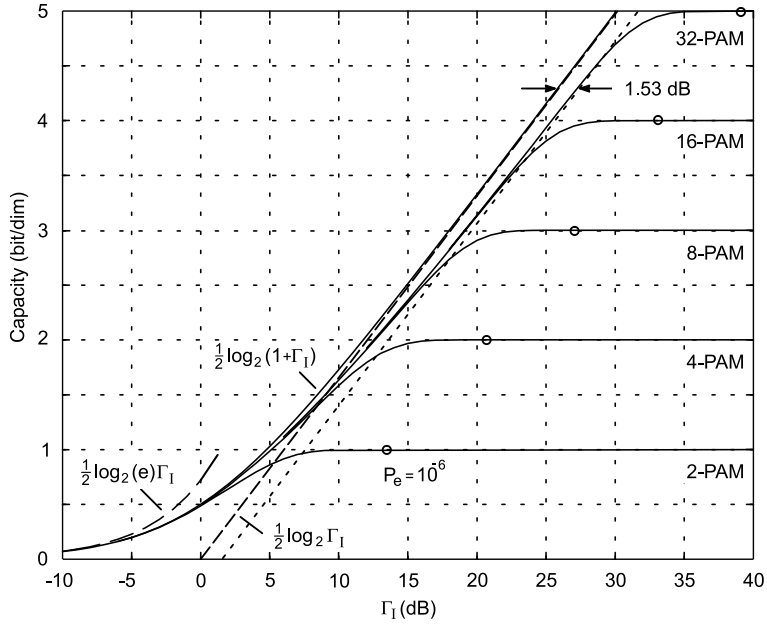


Figure 6.53. Capacity of an ideal AWGN channel for Gaussian and *M*-PAM input signals. [*From Forney and Ungerboeck (1998). © 1998 IEEE.*]

equal to 10^{-6} is obtained for uncoded transmission, are also indicated [12]. We note that the curves saturate as information cannot be transmitted with a rate larger than $R_I = \log_2 M$.

Let us consider, for example, the uncoded transmission of 1 bit of information per modulation interval by a 2-PAM system, where we have a symbol error probability equal to 10^{-6} for $\Gamma_I = 13.5$ dB. If the number of symbols in the alphabet $\mathcal{A}$ doubles, choosing 4-PAM modulation, we see that the coded transmission of 1 bit of information per modulation interval with rate $R_I = 1$ is possible, and an arbitrarily small error probability can be obtained for $\Gamma_I = 5$ dB. This indicates that a coded 4-PAM system may achieve a gain of about 8.5 dB in signal-to-noise ratio over an uncoded 2-PAM system, at an error probability of 10^{-6}. If the number of symbols is further increased, the additional achievable gain is negligible. Therefore we conclude that, by doubling the number of symbols with respect to an uncoded system, we obtain in practice the entire gain that would be expected from the expansion of the input alphabet.

We see from Figure 6.53 that for small values of Γ_I the choice of a binary alphabet is almost optimum: in fact for $\Gamma_I < 1$ (0 dB) the capacity given by (6.282) is essentially equivalent to the capacity given by (6.287) with a binary alphabet of input symbols.

For large values of Γ_I, the capacity of multilevel systems asymptotically approximates a straight line that is parallel to the capacity of the AWGN channel. The asymptotic loss of $\pi e/6$ (1.53 dB) is due to the choice of a uniform distribution rather than Gaussian for the set of input symbols. To achieve the Shannon limit it is not sufficient to use coding techniques with equally likely input symbols, no matter how sophisticated they are: to bridge the gap

of 1.53 dB, *shaping* techniques are required [13] that produce a distribution of the input symbols similar to a Gaussian distribution.

Coding techniques for small Γ_I and large Γ_I are therefore quite different: for low Γ_I, the binary codes are almost optimum and the shaping of the constellation is not necessary; for high Γ_I instead constellations with more than two elements must be used. To reach capacity, coding must be extended with shaping techniques; moreover, to reach the capacity in channels with limited bandwidth, techniques are required that combine coding, shaping and equalization, as we will see in Chapter 13.

Coding strategies depending on the signal-to-noise ratio

The formula of the capacity (6.282) can be expressed as $\Gamma_I/(2^{2C}-1)=1$. This relation suggests the definition of the normalized signal-to-noise ratio

$$\overline{\Gamma}_I = \frac{\Gamma_I}{2^{2R_I}-1} \tag{6.288}$$

for a given R_I given by (6.93). For a scheme that achieves the capacity, R_I is equal to the capacity of the channel C and $\overline{\Gamma}_I = 1$ (0 dB); if $R_I < C$, as it must be in practice, then $\overline{\Gamma}_I > 1$. Therefore the value of $\overline{\Gamma}_I$ indicates how far from the Shannon limit a system operates, or, in other words, the gap that separates the system from capacity. We now consider two cases.

High signal-to-noise ratios. We note from Figure 6.53 that for high values of Γ_I it is possible to find coding methods that allow reliable transmission of several bits per dimension. For an *uncoded* M-PAM system,

$$R_I = \log_2 M \tag{6.289}$$

bits of information are mapped into each transmitted symbol. The average symbol error probability is given by (6.124),

$$P_e = 2\left(1-\frac{1}{M}\right) Q\left(\sqrt{\frac{3}{M^2-1}\Gamma_I}\right) \tag{6.290}$$

We note that P_e is function only of M and Γ_I. Moreover, using (6.289) and (6.288) we obtain

$$\overline{\Gamma}_I = \frac{\Gamma_I}{M^2-1} \tag{6.291}$$

For large M, P_e can therefore be expressed as

$$P_e = 2\left(1-\frac{1}{M}\right) Q\left(\sqrt{3\bar{\Gamma}_I}\right) \simeq 2Q\left(\sqrt{3\overline{\Gamma}_I}\right) \tag{6.292}$$

We note that the relation between P_e and $\overline{\Gamma}_I$ is almost independent of M, if M is large. This relation is used in the comparison illustrated in Figure 6.54 between uncoded systems and the Shannon limit given by $\overline{\Gamma}_I = 1$.

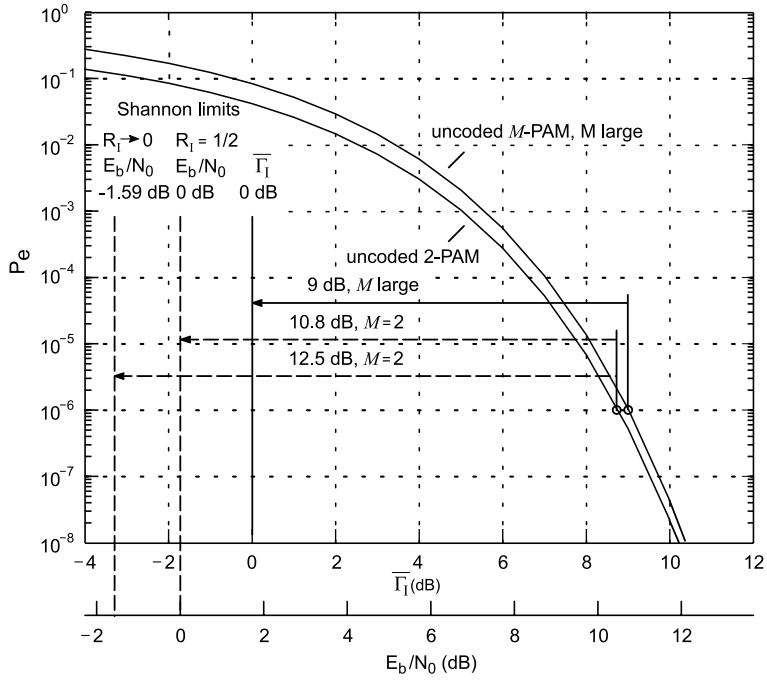


Figure 6.54. Bit error probability as a function of E_b/N_0 for an uncoded 2-PAM system, and symbol error probability as a function of $\overline{\Gamma}_I$ for an uncoded M-PAM system. [*From Forney and Ungerboeck (1998). © 1998 IEEE.*]

Low signal-to-noise ratios. For low values of Γ_I the capacity is less than 1 and can be approximated by binary transmission systems: consequently we refer to coding methods that employ more binary symbols to obtain the reliable transmission of 1 bit (see Section 6.8). For low values of Γ_I it is customary to introduce the following ratio (see (6.107)):

$$\frac{E_b}{N_0} = \frac{2^{2R_I} - 1}{2R_I}\,\overline{\Gamma}_I \tag{6.293}$$

We note the following particular cases:

- if $R_I \ll 1$, then $E_b/N_0 \approx (\ln 2)\,\overline{\Gamma}_I$;
- if $R_I = 1/2$, then $E_b/N_0 = \overline{\Gamma}_I$;
- if $R_I = 1$, then $E_b/N_0 = (3/2)\,\overline{\Gamma}_I$.

For low Γ_I, if the bandwidth can be extended without limit for a given power, for example, by using an orthogonal modulation with $T \to 0$ (see Example 6.7.3), then by increasing the bandwidth, or equivalently the number of dimensions M of input signals, both Γ_I and R_I tend to zero. For systems with limited power and unlimited bandwidth, usually E_b/N_0 is adopted as a figure of merit.

From (6.293) and the Shannon limit $\overline{\Gamma}_I > 1$, we obtain the Shannon limit in terms of E_b/N_0 for a given rate R_I as

$$\frac{E_b}{N_0} > \frac{2^{2R_I} - 1}{2R_I} \tag{6.294}$$

This lower limit monotonically decreases with R_I.

In particular, we examine again the three cases:

- if R_I tends to zero, the *ultimate Shannon limit* is given by

$$\frac{E_b}{N_0} > \ln 2 \quad (-1.59\,\text{dB}) \tag{6.295}$$

  in other words, equation (6.295) affirms that even though an infinitely large bandwidth is used, reliable transmission can be achieved only if $E_b/N_0 > -1.59\,\text{dB}$;

- if the bandwidth is limited, from (6.294) we find that the Shannon limit in terms of E_b/N_0 is higher; for example, if $R_I = 1/2$ the limit becomes $E_b/N_0 > 1$ (0 dB);

- if $R_I = 1$, as $E_b/N_0 = (3/2)\,\overline{\Gamma}_I$, the symbol error probability or bit error probability for an *uncoded* 2-PAM system can be expressed in two equivalent ways:

$$P_{bit} \approx Q\left(\sqrt{3\overline{\Gamma}_I}\right) = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \tag{6.296}$$

Coding gain

Definition 6.2
The *coding gain* of a coded modulation scheme is equal to the reduction in the value of E_b/N_0, or in the value of Γ or $\overline{\Gamma}_I$ (see (11.9)), that is required to obtain a given probability of error relative to a reference uncoded system. If the modulation rate of the coded system remains unchanged, we typically refer to Γ or $\overline{\Gamma}_I$.

Let us consider as reference systems a 2-PAM system and an M-PAM system with $M \gg 1$, for small and large values of Γ_I, respectively. Figure 6.54 illustrates the bit error probability for an uncoded 2-PAM system as a function of both E_b/N_0 and $\overline{\Gamma}_I$. For $P_{bit} = 10^{-6}$, the reference uncoded 2-PAM system operates at about 12.5 dB from the ultimate Shannon limit. Thus a coding gain up to 12.5 dB is possible, in principle, at this probability of error, if the bandwidth can be sufficiently extended to allow the use of binary codes with $R_I \ll 1$; if, instead, the bandwidth can be extended only by a factor 2 with respect to an uncoded system, then a binary code with rate $R_I = 1/2$ can yield a coding gain up to about 10.8 dB.

Figure 6.54 also shows the symbol error probability for an uncoded M-PAM system as a function of $\overline{\Gamma}_I$ for large M. For $P_e = 10^{-6}$, a reference uncoded M-PAM system operates at about 9 dB from the Shannon limit: in other words, assuming a limited bandwidth system, the Shannon limit can be achieved by a code having a gain of about 9 dB.

Cut-off rate

It is useful to introduce the notion of *cut-off rate* R_0 associated with a channel, for a given modulation and class of codes [2].

We sometimes refer to R_0 as a practical upper bound of the transmission bit rate. Therefore for a given channel we can determine the minimum signal-to-noise ratio $(E_b/N_0)_0$ below which reliable transmission is not possible, assuming a certain class of coding and decoding techniques. Typically, for codes with rate $R_c = \frac{1}{2}$ (see Chapter 11), $(E_b/N_0)_0$ is about 2 dB above the signal-to-noise ratio at which capacity is achieved.

6.11 Optimum receivers for signals with random phase

Let us consider transmission over an AWGN channel of one of the signals

$$s_n(t) = Re[s_n^{(bb)}(t)\, e^{j2\pi f_0 t}] \qquad n = 1, 2, \ldots, M \tag{6.297}$$

where $s_n^{(bb)}$ is the complex envelope of s_n, relative to the carrier frequency f_0, with support $(0, t_0)$. If in (6.297) every signal $s_n^{(bb)}$ has a bandwidth smaller than f_0, then the energy of s_n is given by

$$E_n = \int_0^{t_0} s_n^2(t)\, dt = \int_0^{t_0} \frac{1}{2} |s_n^{(bb)}(t)|^2\, dt \tag{6.298}$$

At the receiver, we observe the signal

$$r(t) = s_n(t;\varphi) + w(t) \tag{6.299}$$

where

$$\begin{aligned} s_n(t;\varphi) &= Re[s_n^{(bb)}(t) e^{j\varphi} e^{j2\pi f_0 t}] \\ &= Re[s_n^{(bb)*}(t) e^{-j\varphi} e^{-j2\pi f_0 t}] \qquad n = 1, 2, \ldots, M \end{aligned} \tag{6.300}$$

In other words, at the receiver we assume the carrier is known, except, however, for a phase φ that we assume to be a uniform r.v. in $[-\pi, \pi)$. Receivers, which do not rely on the knowledge of the carrier phase, are called *non-coherent receivers*.

We give three examples of signalling schemes that employ non-coherent receivers.

Example 6.11.1 (Non-coherent binary FSK)
The received signals are expressed as (see also (6.204)):

$$\begin{aligned} s_1(t;\varphi_1) &= A\cos(2\pi f_1 t + \varphi_1) \qquad 0 < t < T \\ s_2(t;\varphi_2) &= A\cos(2\pi f_2 t + \varphi_2) \qquad 0 < t < T \end{aligned} \tag{6.301}$$

where

$$A = \sqrt{\frac{2E_s}{T}} \qquad \varphi_1, \varphi_2 \in U[-\pi, \pi) \tag{6.302}$$

and

$$f_1 = f_0 - f_d \qquad f_2 = f_0 + f_d \tag{6.303}$$

where f_d is the frequency deviation with respect to the carrier f_0. We recall that if

$$f_1 + f_2 = k_1 \frac{1}{T} \ (k_1 \text{ integer}) \quad \text{or else} \quad f_0 \gg \frac{1}{T} \tag{6.304}$$

and if

$$2 f_d T = k \ (k \text{ integer}) \tag{6.305}$$

then $s_1(t, \varphi_1)$ and $s_2(t, \varphi_2)$ are orthogonal.

The minimum value of f_d is given by

$$(f_d)_{min} = \frac{1}{2T} \tag{6.306}$$

which is twice the value we find for the coherent demodulation case (6.203).

Example 6.11.2 (On-off keying)
On-off keying (OOK) is a binary modulation scheme where, for example,

$$s_1(t;\varphi) = A \cos(2\pi f_0 t + \varphi) \qquad 0 < t < T \tag{6.307}$$

and

$$s_2(t;\varphi) = 0 \tag{6.308}$$

where $A = \sqrt{4E_s/T}$, and E_s is the average energy of a pulse.

Example 6.11.3 (DSB modulated signalling with random phase)
We consider an M-ary baseband signalling scheme, $\{s_n^{(bb)}(t)\}$, $n = 1, \ldots, M$, that is modulated in the passband by the *double sideband* technique (see Example 1.7.3 on page 58).

The received signals are expressed as

$$s_n(t;\varphi) = s_n^{(bb)}(t) \cos(2\pi f_0 t + \varphi) \qquad n = 1, \ldots, M \tag{6.309}$$

ML criterion

Given $\varphi = p$, that is for known φ, the ML criterion to detect the transmitted signal has been previously developed starting from (6.26).

The conditional probability density function of the vector $\mathbf{r}$ is given by

$$\begin{aligned} p_{\mathbf{r}|a_0,\varphi}(\boldsymbol{\rho} \mid n, p) &= \frac{1}{(\sqrt{2\pi (N_0/2)})^I} e^{-\frac{1}{N_0}\|\boldsymbol{\rho} - \mathbf{s}_n\|^2} \\ &= \mathcal{K} \exp\left(\frac{2}{N_0} \int_0^{t_0} r(t)\, s_n(t;p)\, dt - \frac{1}{N_0} \int_0^{t_0} s_n^2(t;p)\, dt \right) \end{aligned} \tag{6.310}$$

Using the result

$$\int_0^{t_0} s_n^2(t;\varphi)\,dt = E_n \tag{6.311}$$

we define the following likelihood function, which is equivalent, but not equal, to that defined in (6.27):

$$\mathtt{L}_n[p] = \exp\left(-\frac{E_n}{N_0}\right)\exp\left(\frac{2}{N_0}\int_0^{t_0} r(t)\,s_n(t;p)\,dt\right) \qquad n = 1,\ldots,M \tag{6.312}$$

Given $\varphi = p$, the *maximum likelihood* criterion yields the decision rule

$$\hat{a}_0 = \arg\max_n \mathtt{L}_n[p] \tag{6.313}$$

The dependency on the r.v. φ is removed by taking the expectation of $\mathtt{L}_n[p]$ with respect to φ:[16]

$$\begin{aligned}\mathtt{L}_n &= \int_{-\pi}^{\pi} \mathtt{L}_n[p]\,p_\varphi(p)\,dp \\ &= e^{-\frac{E_n}{N_0}}\frac{1}{2\pi}\int_{-\pi}^{\pi}\exp\left(\frac{2}{N_0}Re\left[\int_0^{t_0} r(t)\,s_n^{(bb)*}(t)\,e^{-j(p+2\pi f_0 t)}\,dt\right]\right)dp\end{aligned} \tag{6.314}$$

using (6.300). We define

$$L_n = \frac{1}{\sqrt{E_n}}\int_0^{t_0} r(t)\left[s_n^{(bb)}(t)\,e^{j2\pi f_0 t}\right]^* dt \tag{6.315}$$

Introducing the polar notation $L_n = |L_n|e^{j\arg L_n}$, (6.314) becomes

$$\begin{aligned}\mathtt{L}_n &= e^{-\frac{E_n}{N_0}}\frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\frac{2\sqrt{E_n}}{N_0}Re[L_n e^{-jp}]}\,dp \\ &= e^{-\frac{E_n}{N_0}}\frac{1}{2\pi}\int_{-\pi}^{\pi} e^{\frac{2\sqrt{E_n}}{N_0}|L_n|\cos(p-\arg L_n)}\,dp\end{aligned} \tag{6.316}$$

We recall the following properties of the Bessel functions (4.216):

1.
$$I_0(x) = \frac{1}{2\pi}\int_{-\pi}^{\pi} e^{x\cos(p-\theta)}\,dp \qquad \forall\theta \tag{6.317}$$

2. $I_0(x)$ is a monotonic increasing function for $x > 0$.

[16] Averaging with respect to the phase φ cannot be considered for PSK and QAM systems, where information is also carried by the phase of the signal.

Then (6.316) becomes

$$\mathtt{L}_n = e^{-\frac{E_n}{N_0}} I_0\left(\frac{2\sqrt{E_n}}{N_0}|L_n|\right) \qquad n = 1, \ldots, M \tag{6.318}$$

Taking the logarithm we obtain the *log-likelihood function*

$$\ell_n = \ln I_0\left(\frac{2\sqrt{E_n}}{N_0}|L_n|\right) - \frac{E_n}{N_0} \tag{6.319}$$

If the signals have all the same energy, and considering that both ln and I_0 are monotonic functions, the ML decision criterion can be expressed as

$$\hat{a}_0 = \arg\max_n |L_n| \tag{6.320}$$

Implementation of a non-coherent ML receiver

The scheme that implements the criterion (6.320) is illustrated in Figure 6.55 for the case of all E_n equal. Polar notation is adopted for the complex envelope:

$$s_n^{(bb)}(t) = |s_n^{(bb)}(t)|e^{j\Phi_n(t)} \tag{6.321}$$

From (6.315), the scheme first determines the real and the imaginary parts of L_n starting from $s_n^{(bb)}$, and then determines the squared magnitude. Note that the available signal is $s_n^{(bb)}(t)\, e^{j\varphi_0}$, where φ_0 is a constant, rather than $s_n^{(bb)}(t)$: this, however, does not modify the magnitude of L_n. As shown in Figure 6.56, the generic branch of the scheme in Figure 6.55, composed of the I branch and the Q branch, can be implemented by a complex-valued passband filter (see (6.315)); the bold line denotes a complex-valued signal.

Alternatively, the matched filter can be real valued if it is followed by a *phase-splitter*: in this case the receiver is illustrated in Figure 6.57. For the generic branch, the desired value $|L_n|$ coincides with the absolute value of the output signal of the phase-splitter at instant t_0,

$$|L_n| = |y_n^{(a)}(t)|_{t=t_0} \tag{6.322}$$

The cascade of the phase-splitter and the "modulo" transformation is called the *envelope detector* of the signal $y_n(t)$ (see (1.196) and (1.202)).

A simplification arises if the various signals $s_n^{(bb)}$ have a bandwidth B much lower than f_0. In this case, recalling (1.202), if $y_n^{(bb)}$ is the complex envelope of y_n, at the matched filter output the following relation holds

$$\begin{aligned} y_n(t) &= Re[y_n^{(bb)}(t)e^{j2\pi f_0 t}] \\ &= |y_n^{(bb)}(t)|\cos(2\pi f_0 t + \arg y_n^{(bb)}(t)) \end{aligned} \tag{6.323}$$

Moreover, from (6.322) and (1.199) we have

$$|L_n| = |y_n^{(bb)}(t)|_{t=t_0} \tag{6.324}$$

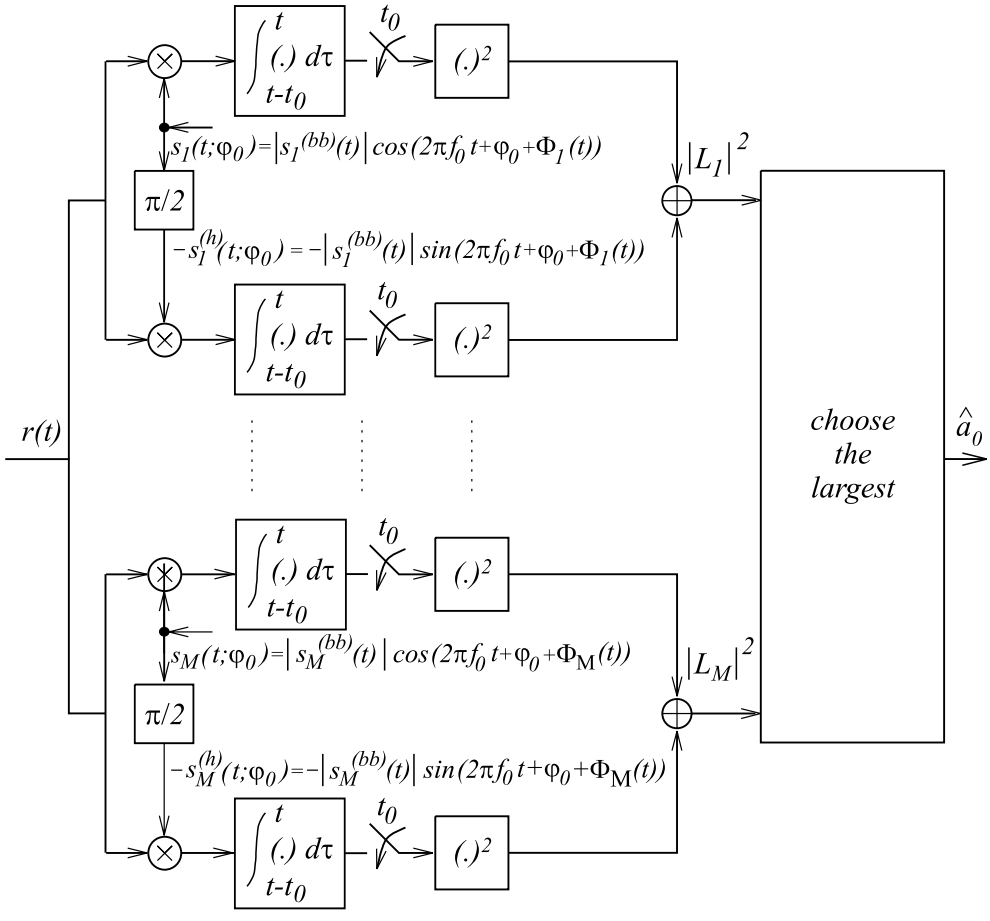


Figure 6.55. Non-coherent ML receiver of the type *square-law detector*.

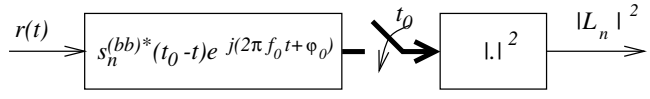


Figure 6.56. Implementation of a branch of the scheme of Figure 6.55 by a complex-valued passband matched filter.

Now, if $f_0 \gg B$, to determine the amplitude $|y_n^{(bb)}(t)|$, we can use one of the schemes of Figure 6.58.

Example 6.11.4 (Non-coherent binary FSK)
We show in Figure 6.59 two alternative schemes of the ML receiver for the modulation system considered in Example 6.11.1.

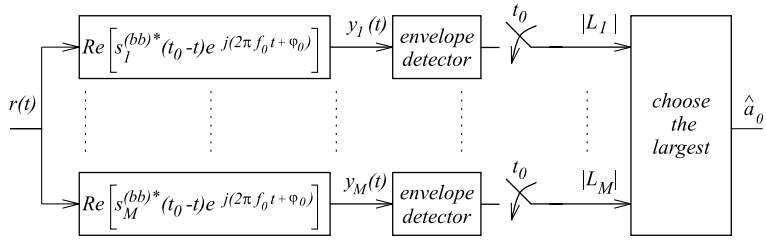


Figure 6.57. Non-coherent ML receiver of the type *envelope detector*, using passband matched filters.

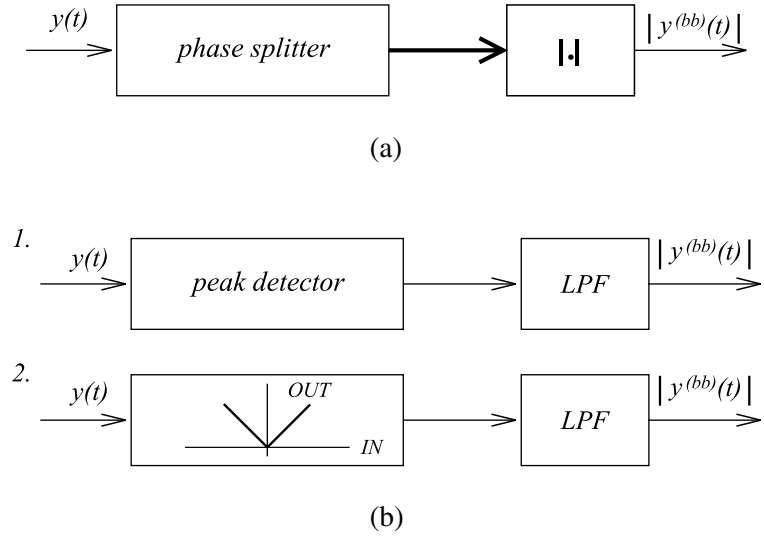


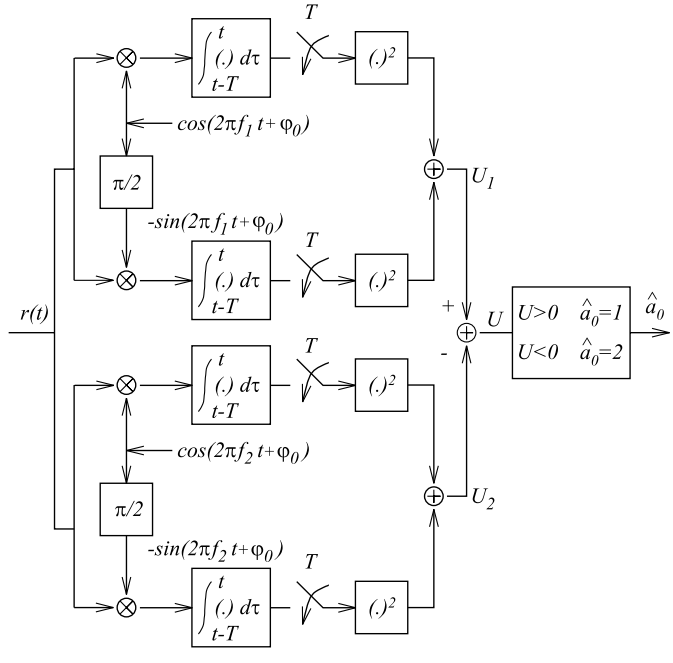
Figure 6.58. (a) Ideal implementation of an envelope detector, and (b) two simpler approximate implementations.

Example 6.11.5 (On-off keying)
We illustrate in Figure 6.60 the receiver for the modulation system of Example 6.11.2 where, recalling (6.318), we have

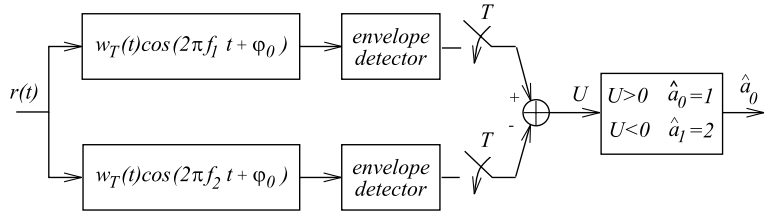
$$U_{Th} = \frac{N_0}{2\sqrt{E_1}} I_0^{-1}(e^{E_1/N_0}) \tag{6.325}$$

where

$$E_1 = \frac{A^2 T}{2} \tag{6.326}$$



(a) Square-law detector.



(b) Envelope detector.

Figure 6.59. Two ML receivers for a non-coherent 2-FSK system.

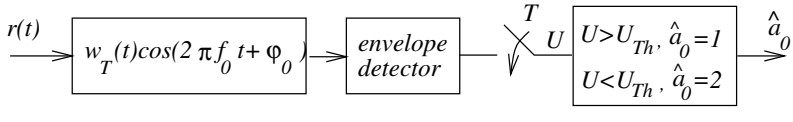


Figure 6.60. Envelope detector receiver for an on-off keying system.

Example 6.11.6 (DSB modulated signalling with random phase)
With reference to the Example 6.11.3, we show in Figure 6.61 the receiver for a baseband M-ary signalling scheme that is DSB modulated with random phase. Depending upon the signalling type, further simplifications can be done by extracting functions that are common to the different branches.

Error probability for a non-coherent binary FSK system

We now derive the error probability of the system of Example 6.11.4. We assume that s_1 is transmitted: then

$$r(t) = s_1(t) + w(t) \tag{6.327}$$

and

$$P_{bit} = P[U_1 < U_2 \mid s_1] \tag{6.328}$$

Equivalently, if we define

$$V_1 = \sqrt{U_1} \quad \text{and} \quad V_2 = \sqrt{U_2} \tag{6.329}$$

we have

$$P_{bit} = P[V_1 < V_2 \mid s_1] \tag{6.330}$$

Now, recalling assumption (6.305), that is $s_1(t, \varphi_1) \perp s_2(t, \varphi_2)$, we have

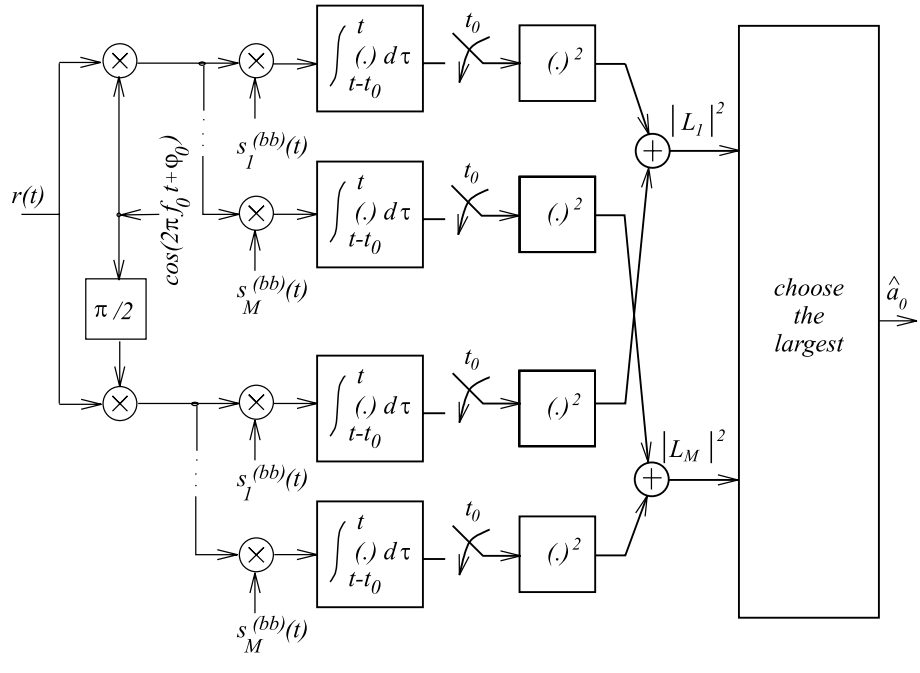
$$\begin{aligned} U_2 &= \left(\int_0^T r(t) \cos(2\pi f_2 t + \varphi_0)\, dt \right)^2 + \left(\int_0^T r(t) \sin(2\pi f_2 t + \varphi_0)\, dt \right)^2 \\ &= w_{2,c}^2 + w_{2,s}^2 \end{aligned} \tag{6.331}$$

where

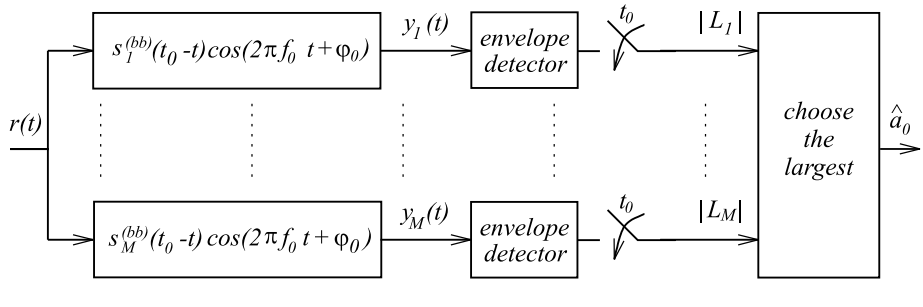
$$\begin{aligned} w_{2,c} &= \int_0^T w(t) \cos(2\pi f_2 t + \varphi_0)\, dt \\ w_{2,s} &= \int_0^T w(t) \sin(2\pi f_2 t + \varphi_0)\, dt \end{aligned} \tag{6.332}$$

If we define

$$\begin{aligned} w_{1,c} &= \int_0^T w(t) \cos(2\pi f_1 t + \varphi_0)\, dt \\ w_{1,s} &= \int_0^T w(t) \sin(2\pi f_1 t + \varphi_0)\, dt \end{aligned} \tag{6.333}$$



(a) Square-law detector.



(b) Envelope detector.

Figure 6.61. Two receivers for a DSB modulation system with *M*-ary signalling and random phase.

we have

$$\begin{aligned} U_1 &= \left(\int_0^T r(t)\cos(2\pi f_1 t + \varphi_0)\,dt\right)^2 + \left(\int_0^T r(t)\sin(2\pi f_1 t + \varphi_0)\,dt\right)^2 \\ &= \left(\frac{AT}{2}\cos(\varphi_0 - \varphi_1) + w_{1,c}\right)^2 + \left(\frac{AT}{2}\sin(\varphi_0 - \varphi_1) + w_{1,s}\right)^2 \end{aligned} \tag{6.334}$$

where from (6.302) we also have

$$\frac{AT}{2} = \sqrt{\frac{E_s T}{2}} \tag{6.335}$$

As $w(t)$ is a white Gaussian random process with zero mean, $w_{2,c}$ and $w_{2,s}$ are two jointly Gaussian r.v.s with

$$\begin{aligned} &E[w_{2,c}] = E[w_{2,s}] = 0 \\ &E[w_{2,c}^2] = E[w_{2,s}^2] = \frac{N_0}{2}\,\frac{T}{2} \\ &E[w_{2,c}\,w_{2,s}] = \int_0^T \int_0^T \frac{N_0}{2}\,\delta(t_1 - t_2)\cos(2\pi f_2 t_1 + \varphi_0)\sin(2\pi f_2 t_2 + \varphi_0)\,dt_1\,dt_2 = 0 \end{aligned} \tag{6.336}$$

Similar considerations hold for $w_{1,c}$ and $w_{1,s}$.

Therefore V_2, with statistical power $2(N_0 T/4)$, has a Rayleigh probability density

$$p_{V_2}(v_2) = \frac{v_2}{N_0 T/4}\, e^{-\frac{v_2^2}{2(N_0 T/4)}}\, 1(v_2) \tag{6.337}$$

whereas V_1 has a Rice probability density function

$$p_{V_1}(v_1) = \frac{v_1}{N_0 T/4}\, e^{-\frac{(v_1^2 + (AT/2)^2)}{2(N_0 T/4)}}\, I_0\left(\frac{v_1 (AT/2)}{N_0 T/4}\right) 1(v_1) \tag{6.338}$$

Consequently equation (6.330) assumes the expression[17]

$$\begin{aligned} P_{bit} &= \int_0^{+\infty} P[V_1 < v_2 \mid V_2 = v_2]\, p_{V_2}(v_2)\,dv_2 \\ &= \int_0^{+\infty} \left(\int_0^{v_2} p_{V_1}(v_1)\,dv_1\right) p_{V_2}(v_2)\,dv_2 \\ &= \frac{1}{2}\, e^{-\frac{1}{2}\frac{E_s}{N_0}} \end{aligned} \tag{6.340}$$

[17] To compute the following integrals we recall the Weber-Sonine formula:

$$\int_0^{+\infty} x\, e^{-\alpha(x^2/2)}\, I_0(\beta x)\,dx = \frac{1}{\alpha}\, e^{\beta^2/2\alpha} \tag{6.339}$$

It can be shown that this result is not limited to FSK systems and is valid for any pair of non-coherent orthogonal signals with energy E_s.

Performance comparison of binary systems

The received signals are given by (6.301), where f_d satisfies the constraint (6.305). The correlation coefficient between the two signals is equal to zero, and from (6.340) we have

$$\text{FSK(NC):} \qquad P_{bit} = \frac{1}{2} e^{-\frac{1}{2}\frac{E_s}{N_0}} \tag{6.341}$$

A comparison with a non-coherent binary system with differentially encoded bits, such as DBPSK, is illustrated in Figure 6.62. The differential receiver for DBPSK directly gives the original uncoded bits, thus from (6.163) we have

$$\text{DBPSK:} \qquad P_{bit} = \tfrac{1}{2} e^{-\frac{E_s}{N_0}} \tag{6.342}$$

A comparison between (6.341) and (6.342) indicates that DBPSK is better than FSK by about 3 dB in Γ, for the same P_{bit}.

The performance of a binary FSK system and that of a differentially encoded BPSK system with coherent detection are compared in Figure 6.62. In particular, from (6.75) it follows

$$\text{FSK(CO):} \qquad P_{bit} = Q\left(\sqrt{\frac{E_s}{N_0}}\right) \tag{6.343}$$

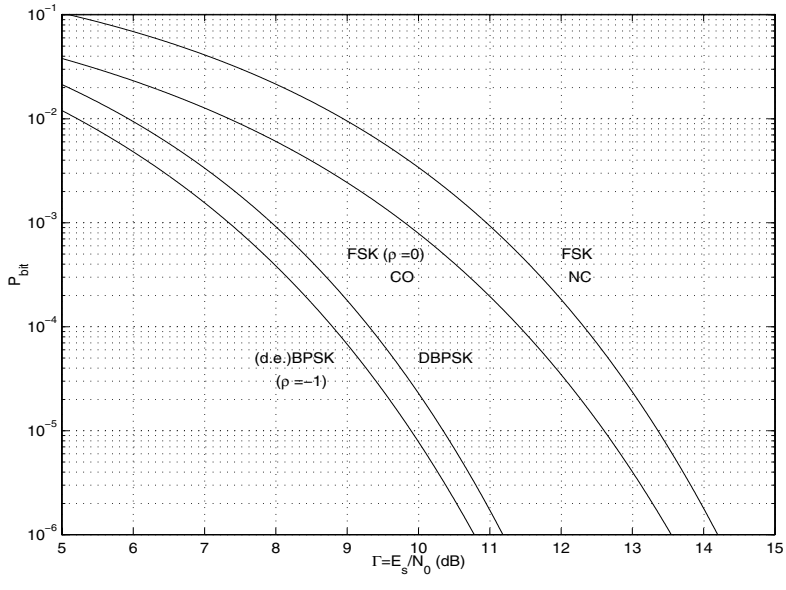


Figure 6.62. Bit error probability as a function of Γ for BPSK and binary FSK systems, with coherent (CO) and non-coherent (NC) detection.

Taking into account differential decoding, from (6.64) we have[18]

$$\text{(d.e.)BPSK:} \qquad P_{bit} \simeq 2Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \tag{6.344}$$

We observe that the difference between coherent FSK and non-coherent FSK is less than 2 dB for $P_e \leq 10^{-3}$, and becomes less than 1 dB for $P_e \leq 10^{-5}$. We also note that because of the large bandwidth required, see (6.206), FSK systems with $M > 2$ are not widely used.

6.12 Binary modulation systems in the presence of flat fading

We assume now that the channel introduces, in addition to a random phase, also a random attenuation (see Section 4.6.5). The received signal is expressed as

$$r(t) = Re[s_n^{(bb)}(t) g_1 e^{j2\pi f_0 t}] + w(t) \qquad n \in \{1, \ldots, M\} \qquad 0 < t < T \tag{6.345}$$

where $g_1 = g_{1,I} + j g_{1,Q}$ is a Rayleigh r.v., that is $g_{1,I}$ and $g_{1,Q}$ are uncorrelated Gaussian r.v.s, with zero mean and equal variance. In polar notation $g_1 = |g_1| e^{j\varphi}$, where $\varphi \in U(-\pi, \pi)$ and $p_{|g_1|}$ is given by (4.215). As $|g_1|$ determines the signal level at the input of the receiver, the signal-to-noise ratio at the receiver input,

$$\Gamma = \frac{E_s}{N_0} |g_1|^2 \tag{6.346}$$

where E_s is the average energy of the transmitted signal, is a function of $|g_1|$ and therefore it is a random variable. We define the average signal-to-noise ratio

$$\Gamma_{avg} = \frac{E_s}{N_0} E[|g_1|^2] \tag{6.347}$$

Then the probability density of Γ is that of a *chi-square* r.v.:

$$p_\Gamma(a) = \frac{1}{\Gamma_{avg}} e^{-a/\Gamma_{avg}} 1(a) \tag{6.348}$$

To compute the performance of a signalling scheme in the presence of flat fading, we regard the expressions of P_e obtained in the previous sections as functions of Γ. Therefore we consider P_e as the conditional error probability for a given value of $|g_1|$. To evaluate the mean error probability we apply the total probability theorem, that yields

$$P_e = \int_0^{+\infty} P_e(a)\, p_\Gamma(a)\, da \tag{6.349}$$

Limiting ourselves to binary signalling schemes and substituting P_e given by (6.341), (6.342), (6.343) and (6.344) in (6.349), we obtain:[19]

---

[18] For a more accurate evaluation of the probability of error see footnote 14 on page 478.

[19] For the computation of the integral in (6.349) we recall the following result:

$$\int_0^{+\infty} Q\left(\sqrt{\alpha x}\right) \frac{1}{\beta} e^{-\frac{x}{\beta}} dx = \frac{1}{2}\left(1 - \sqrt{\frac{\beta}{\beta + \frac{2}{\alpha}}}\right) \tag{6.350}$$

1. *Orthogonal binary FSK with coherent detection*

$$P_{bit} = \frac{1}{2}\left(1 - \sqrt{\frac{\Gamma_{avg}}{2 + \Gamma_{avg}}}\right) \tag{6.351}$$

2. *Differentially encoded BPSK with coherent detection*

$$P_{bit} = 1 - \sqrt{\frac{\Gamma_{avg}}{1 + \Gamma_{avg}}} \tag{6.352}$$

   We note that both the above expressions are in practice a lower limit to P_{bit}, as it is assumed that an estimate of the phase φ is available, which is very hard to obtain under fading conditions. In case the uncertainty on the phase is relevant, non-coherent DPSK and FSK systems are valid alternatives.

3. *Orthogonal binary FSK with non-coherent detection*

$$P_{bit} = \frac{1}{2 + \Gamma_{avg}} \tag{6.353}$$

4. *DBPSK*

$$P_{bit} = \frac{1}{2(1 + \Gamma_{avg})} \tag{6.354}$$

   The various expressions of P_{bit} as a function of Γ_{avg} are plotted in Figure 6.63 and compared with the case of transmission over an AWGN channel.

We note that to achieve a certain P_{bit}, it is required a substantially larger E_s as compared to the case of transmission over an AWGN channel, for the same N_0.

For a systematic method to determine the performance of systems in the presence of a channel affected by *multipath fading*, we refer the reader to [14], and to the references therein.

Diversity

In the previous section it became apparent that the probability of error for transmission over channels with Rayleigh fading varies *inversely proportional* to the signal-to-noise ratio, rather than exponentially as in the AWGN channel case: therefore a large transmission power is needed to obtain good system performance. To mitigate this problem it is useful to resort to the concept of *diversity*, that is exploiting channels that are *independent*, or at least highly uncorrelated, for communication. The basic idea consists in providing the receiver with several replicas of the signal via independent channels, so that the probability is small that the attenuation due to fading is high for all the channels. There are various diversity techniques.

1. *Frequency diversity*: the same signal is transmitted using several carriers, separated from each other in frequency by an interval that is larger than the *coherence bandwidth* of the channel.

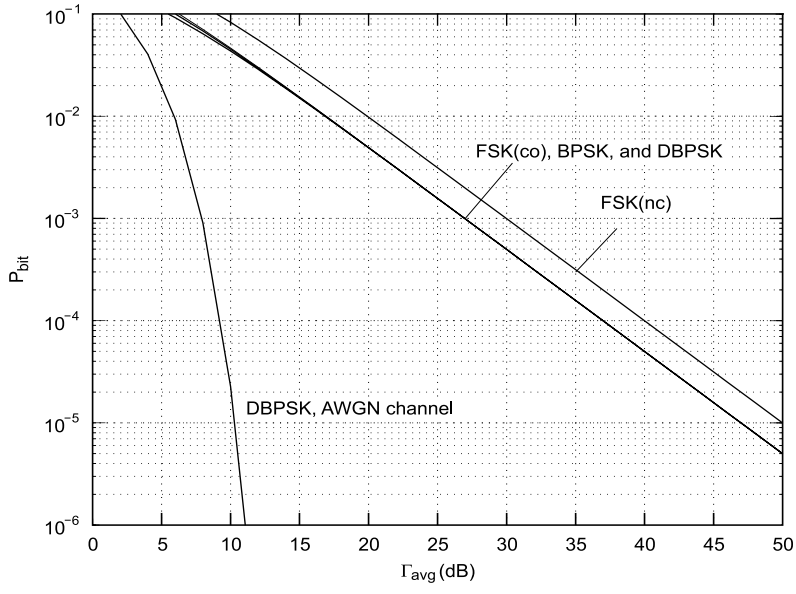


Figure 6.63. Bit error probability as a function of Γ_{avg} for BPSK, DBPSK, and binary FSK systems, for a flat Rayleigh fading channel.

2. *Time diversity*: the same signal is transmitted over different time slots, spaced by an interval that is larger than the *coherence time* of the channel.

3. *Space diversity*: multiple reflections from ground and surrounding buildings can make the power of the received signal change rapidly; by setting two or more antennas close to each other, we can select the antenna that provides the signal with higher power.

4. *Polarization diversity*: several channels are obtained for transmission by orthogonal polarization.

5. *Combinations of the previous techniques*: for the many techniques of combining available, both linear (*equal gain*, *selection*, *maximal ratio*) and non-linear (*square law*), we refer to Section 8.18 and to the bibliography [15, 16, 17].

6.13 Transmission methods

6.13.1 Transmission methods between two users

A transmission link between two users of a communication network may be classified as

a) *Full duplex*, when two users A and B can send information to each other simultaneously, not necessarily by using the same transmission channels in the two directions.

b) *Half duplex*, when two users A and B can send information in only one direction at a time, from A to B or from B to A, alternatively.

c) *Simplex*, when only A can send information to B, that is the link is unidirectional.

Three methods

In the following we give three examples of transmission methods which are used in practice.

a) *Frequency-division duplexing* (FDD): in this case the two users are assigned different transmission bands using the same transmission medium, thus allowing full-duplex transmission. Examples of FDD systems are the GSM, which uses a radio channel (see Section 17.A.2), and the VDSL, which uses a twisted pair cable (see page 1146).

b) *Time-division duplexing* (TDD): in this case the two users are assigned different slots in a time frame (see Section 6.13.2). If the duration of one slot is small with respect to that of the message, we speak of *full-duplex* TDD systems. Examples of TDD systems are the DECT, which uses a radio channel (see Section 17.A.6), and the *ping-pong* BR-ISDN, which uses a twisted pair cable.

c) *Full-duplex systems over a single band*: in this case the two users transmit simultaneously in two directions using the same transmission band; examples are the HDSL (see Section 17.1.1), and in general high-speed transmission systems over twisted-pair cables for LAN applications (see Section 17.1.2). The two directions of transmission are separated by a hybrid; the receiver eliminates echo signals by echo cancellation techniques. We note that *full-duplex* transmission over a single band is possible also over radio channels, but in practice alternative methods are still preferred because of the complexity required by echo cancellation.

6.13.2 Channel sharing: deterministic access methods

We distinguish three cases for channel access by N users:

1. Subdivision of the channel passband into N_B separate sub-bands that may be used for transmission (see Figure 6.64a).

2. Subdivision of a sequence of modulation intervals into adjacent subsets called *frames*, each in turn subdivided into N_S adjacent subsets called *slots*. Within a *frame*, each *slot* is identified by an index i, $i = 0, \ldots, N_S - 1$ (see Figure 6.64b).

3. Signalling by N_0 orthogonal signals (see for example Figure 6.71).

N users may share the channel using one of the following methods.[20]

1. *Frequency division multiple access* (FDMA): to each user is assigned one of the N_B sub-bands.

[20] The access methods discussed in this section are deterministic, as each user knows exactly at which point in time the channel resources are reserved for transmission; an alternative approach is represented by random access techniques, e.g., ALOHA, CSMA/CD, collision resolution protocols [18] (see also Chapter 17).

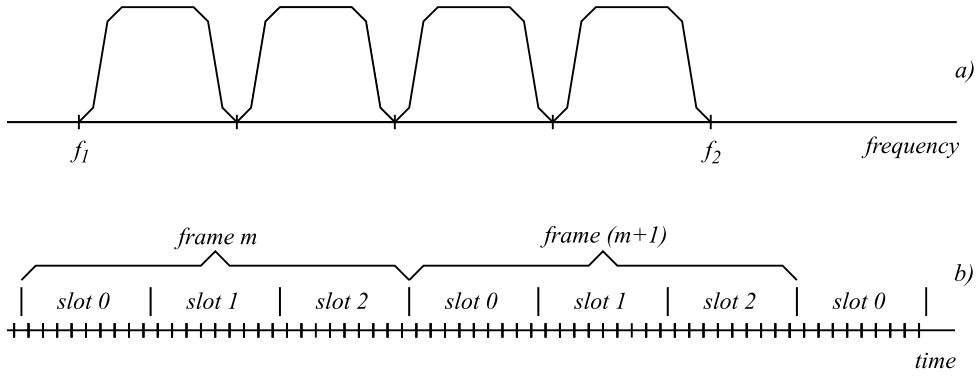


Figure 6.64. Illustration of (a) FDMA, and (b) TDMA.

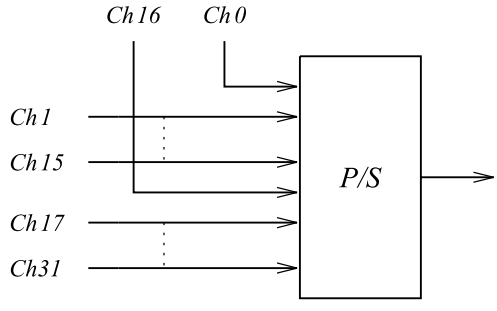
2. *Time division multiple access* (TDMA): to each user is assigned one of the N_S time sequences (slots), whose elements identify the modulation intervals.

3. *Code division multiple access* (CDMA): to each user is assigned a modulation scheme that employs one of the N_0 orthogonal signals, preserving the orthogonality between modulated signals of the various users. For example, for the case $N_0 = 8$, to each user may be assigned one orthogonal signal of those given in Figure 6.71; for binary modulation, within a modulation interval each user then transmits the assigned orthogonal signal or the antipodal signal.

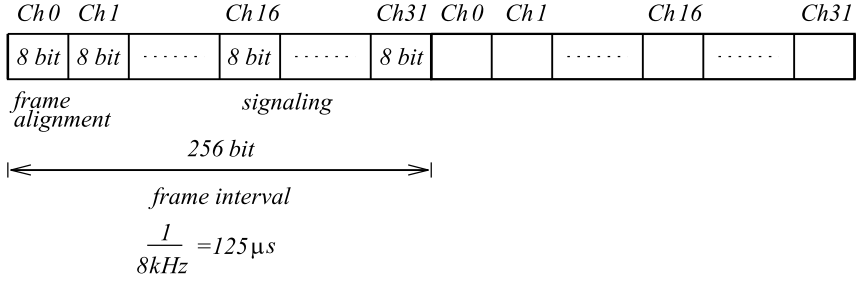
We give an example of implementation of the TDMA principle.

Example 6.13.1 (Time-division multiplexing)
Time-division multiplexing (TDM) is the interleaving of several digital messages into one digital message with a higher bit rate; as an example we illustrate the generation of the European base group, called E1, at 2.048 Mbit/s, that is obtained by multiplexing 30 PCM coded speech signals (or channels) at 64 kbit/s. As shown in Figure 6.65, each 8-bit sample of each channel is inserted into a pre-assigned *slot* of a frame composed of 32 slots, equivalent to $32 \cdot 8 = 256$ bits. The frame structure must contain information bits to identify the beginning of a frame (channel ch0) by 8 known framing bits; 8 bits are employed for signalling between central offices (channel ch16). The remaining 30 channels are for the transmission of signals. As the duration of a frame is of 125 µs, equal to the interval between two PCM samples of the same channel, the overall digital message has a bit rate of 256 bit/125 µs = 2.048 Mbit/s; we note, however, that of the 256 bits of the frame only $30 \cdot 8 = 240$ bits carry information related to signals.

In the United States, Canada, and Japan the base group, analog to E1, is called *T1 carrier system* and has a bit rate of 1.544 Mbit/s, obtained by multiplexing 24 PCM speech coded signals at 64 kbit/s. In this case the frame is such that one bit per channel is employed for signalling: this bit is "robbed" from the least important bit of the 8-bit PCM sample, thus making it a 7-bit code word per sample; there is then only one bit for the synchronization of the whole frame. The entire frame is formed of $24 \cdot 8 + 1 = 193$ bits.



(a) Basic scheme



(b) Frame structure

Figure 6.65. TDM in the European base group at 2.048 Mbit/s.

Bibliography

[1] J. M. Wozencraft and I. M. Jacobs, *Principles of communication engineering*. New York: John Wiley & Sons, 1965.

[2] S. Benedetto and E. Biglieri, *Principles of digital transmission with wireless applications*. New York: Kluwer Academic Publishers, 1999.

[3] J. G. Proakis, *Digital communications*. New York: McGraw-Hill, 3rd ed., 1995.

[4] D. Divsalar, M. K. Simon, and M. Shahshahani, "The performance of trellis-coded MDPSK with multiple symbol detection", *IEEE Trans. on Communications*, vol. 38, pp. 1391–1403, Sept. 1990.

[5] M. Abramovitz and I. A. Stegun, eds, *Handbook of mathematical functions*. New York: Dover Publications, 1965.

[6] R. G. Gallager, *Information theory and reliable communication*. New York: John Wiley & Sons, 1968.

[7] T. M. Cover and J. Thomas, *Elements of information theory*. New York: John Wiley & Sons, 1991.

[8] C. E. Shannon, "A mathematical theory of communication", *Bell System Technical Journal*, vol. 27, pp. 379–427 (Part I) and 623–656 (Part II), 1948.

[9] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas", *Wireless Person. Commun.*, vol. 6, pp. 311–335, June 1998.

[10] E. Telatar, "Capacity of multi–antenna Gaussian channels", *Europ. Trans. on Telecomm.*, vol. 10, pp. 585–595, Nov.–Dec. 1999.

[11] G. Ungerboeck, "Channel coding with multilevel/phase signals", *IEEE Trans. on Information Theory*, vol. 28, pp. 55–67, Jan. 1982.

[12] G. D. Forney, Jr. and G. Ungerboeck, "Modulation and coding for linear Gaussian channels", *IEEE Trans. on Information Theory*, vol. 44, pp. 2384–2415, Oct. 1998.

[13] G. D. Forney, Jr., "Trellis shaping", *IEEE Trans. on Information Theory*, vol. 38, pp. 281–300, Mar. 1992.

[14] M. K. Simon and M.-S. Alouini, "Exponential-type bounds on the generalized Marcum Q-function with application to error probability analysis over fading channels", *IEEE Trans. on Communications*, vol. 48, pp. 359–366, Mar. 2000.

[15] M. Schwartz, W. R. Bennett, and S. Stein, *Communication systems and techniques*. New York: McGraw-Hill, 1966.

[16] T. S. Rappaport, *Wireless communications: principles and practice*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[17] G. L. Stuber, *Principles of mobile communication*. Norwell, MA: Kluwer Academic Publishers, 1996.

[18] *Multiple access communications: foundations for emerging technologies*. (N. Abramson, ed.), Piscataway: IEEE Press, 1993.

[19] W. F. McGee, "Another recursive method of computing the Q-function", *IEEE Trans. on Information Theory*, vol. 16, pp. 500–501, July 1970.

[20] R. E. Ziemer and W. H. Tranter, *Principles of communications: systems, modulation, and noise*. New York: John Wiley & Sons, 4th ed., 1995.

Appendix 6.A *Gaussian distribution function and Marcum function*

6.A.1 The Q function

The probability density function of a Gaussian variable w with mean $\mathtt{m}$ and variance σ^2 is given by

$$p_w(b) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(b-\mathtt{m})^2}{2\sigma^2}} \tag{6.355}$$

We define *normalized Gaussian distribution* ($\mathtt{m} = 0$ and $\sigma^2 = 1$) as the function

$$\Phi(a) = \int_{-\infty}^{a} p_w(b)\, db = \int_{-\infty}^{a} \frac{1}{\sqrt{2\pi}} e^{-b^2/2}\, db \tag{6.356}$$

It is often convenient to use the *complementary Gaussian distribution* function, defined as

$$Q(a) = 1 - \Phi(a) = \int_{a}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-b^2/2}\, db \tag{6.357}$$

Two other functions that are widely used are the *error function*

$$\begin{aligned} \operatorname{erf}(a) &= -1 + 2\int_{-\infty}^{a\sqrt{2}} p_w(b)\, db \\ &= 1 - 2\int_{a\sqrt{2}}^{+\infty} \frac{1}{\sqrt{2\pi}} e^{-b^2}\, db \end{aligned} \tag{6.358}$$

and the *complementary error function*

$$\operatorname{erfc}(a) = 1 - \operatorname{erf}(a) \tag{6.359}$$

which are related to Φ and Q by the following equations

$$\Phi(a) = \frac{1}{2}\left[1 + \operatorname{erf}\left(\frac{a}{\sqrt{2}}\right)\right] \tag{6.360}$$

$$Q(a) = \frac{1}{2}\operatorname{erfc}\left(\frac{a}{\sqrt{2}}\right) \tag{6.361}$$

Table 6.10 Complementary gaussian distribution.

a	$Q(a)$	a	$Q(a)$	a	$Q(a)$
0.0	5.0000(−01)*	2.7	3.4670(−03)	5.4	3.3320(−08)
0.1	4.6017(−01)	2.8	2.5551(−03)	5.5	1.8990(−08)
0.2	4.2074(−01)	2.9	1.8658(−03)	5.6	1.0718(−08)
0.3	3.8209(−01)	3.0	1.3499(−03)	5.7	5.9904(−09)
0.4	3.4458(−01)	3.1	9.6760(−04)	5.8	3.3157(−09)
0.5	3.0854(−01)	3.2	6.8714(−04)	5.9	1.8175(−09)
0.6	2.7425(−01)	3.3	4.8342(−04)	6.0	9.8659(−10)
0.7	2.4196(−01)	3.4	3.3693(−04)	6.1	5.3034(−10)
0.8	2.1186(−01)	3.5	2.3263(−04)	6.2	2.8232(−10)
0.9	1.8406(−01)	3.6	1.5911(−04)	6.3	1.4882(−10)
1.0	1.5866(−01)	3.7	1.0780(−04)	6.4	7.7688(−11)
1.1	1.3567(−01)	3.8	7.2348(−05)	6.5	4.0160(−11)
1.2	1.1507(−01)	3.9	4.8096(−05)	6.6	2.0558(−11)
1.3	9.6800(−02)	4.0	3.1671(−05)	6.7	1.0421(−11)
1.4	8.0757(−02)	4.1	2.0658(−05)	6.8	5.2310(−12)
1.5	6.6807(−02)	4.2	1.3346(−05)	6.9	2.6001(−12)
1.6	5.4799(−02)	4.3	8.5399(−06)	7.0	1.2798(−12)
1.7	4.4565(−02)	4.4	5.4125(−06)	7.1	6.2378(−13)
1.8	3.5930(−02)	4.5	3.3977(−06)	7.2	3.0106(−13)
1.9	2.8717(−02)	4.6	2.1125(−06)	7.3	1.4388(−13)
2.0	2.2750(−02)	4.7	1.3008(−06)	7.4	6.8092(−14)
2.1	1.7864(−02)	4.8	7.9333(−07)	7.5	3.1909(−14)
2.2	1.3903(−02)	4.9	4.7918(−07)	7.6	1.4807(−14)
2.3	1.0724(−02)	5.0	2.8665(−07)	7.7	6.8033(−15)
2.4	8.1975(−03)	5.1	1.6983(−07)	7.8	3.0954(−15)
2.5	6.2097(−03)	5.2	9.9644(−08)	7.9	1.3945(−15)
2.6	4.6612(−03)	5.3	5.7901(−08)	8.0	6.2210(−16)

* Writing 5.0000(−01) means 5.0000×10^{-1}.

In Table 6.10 the values assumed by the complementary Gaussian distribution are given for values of the argument between 0 and 8. We present below some bounds of the Q function.

$$bound_1: \quad Q_1(a) = \frac{1}{\sqrt{2\pi}\,a}\left(1 - \frac{1}{a^2}\right)\exp\left(-\frac{a^2}{2}\right) \tag{6.362}$$

$$bound_2: \quad Q_2(a) = \frac{1}{\sqrt{2\pi}\,a}\exp\left(-\frac{a^2}{2}\right) \tag{6.363}$$

$$bound_3: \quad Q_3(a) = \frac{1}{2}\exp\left(-\frac{a^2}{2}\right) \tag{6.364}$$

The Q function and the above bounds are illustrated in Figure 6.66.

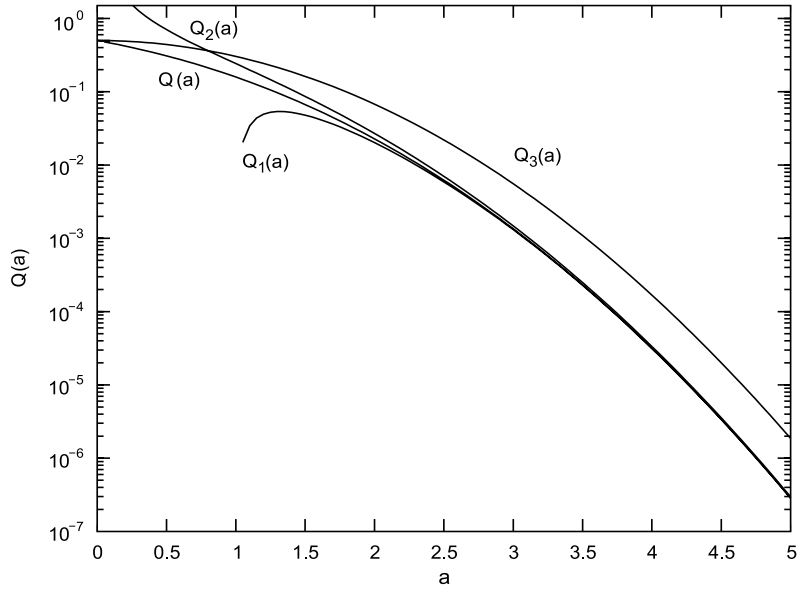


Figure 6.66. The Q function and relative bounds.

6.A.2 The Marcum function

We define the first-order *Marcum function* as

$$Q_1(a,b) = \int_b^{+\infty} x\, e^{-\frac{x^2+a^2}{2}} I_0(ax)\, dx \tag{6.365}$$

where I_0 is the *modified Bessel function* of the first type and order zero, defined in (4.216).

From (6.365), two particular cases follow:

$$Q_1(0,b) = e^{-\frac{b^2}{2}} \tag{6.366}$$

$$Q_1(a,0) = 1 \tag{6.367}$$

Moreover, for $b \gg 1$ and $b \gg b-a$ the following approximation holds

$$Q_1(a,b) \simeq Q(b-a) \tag{6.368}$$

where the Q function is given by (6.357).

A useful approximation valid for $b \gg 1$, $a \gg 1$, $b \gg b-a > 0$, is given by

$$1 + Q_1(a,b) - Q_1(b,a) \simeq 2Q(b-a) \tag{6.369}$$

We also give the Simon bound [14]

$$e^{-\frac{(b+a)^2}{2}} \leq Q_1(a,b) \leq e^{-\frac{(b-a)^2}{2}} \qquad b > a > 0 \tag{6.370}$$

and

$$1 - \frac{1}{2}\left[e^{-\frac{(a-b)^2}{2}} - e^{-\frac{(a+b)^2}{2}}\right] \leq Q_1(a, b) \qquad a > b \geq 0 \tag{6.371}$$

We observe that in (6.370) the upper bound is very tight, and the lower bound for a given value of b becomes looser as a increases. In (6.371) the lower bound is very tight. A recursive method for computing the Marcum function is given in [19].

Appendix 6.B *Gray coding*

In this appendix we give the procedure to construct a list of 2^n binary words of n bits, where adjacent words differ in only one bit.

The case for $n = 1$ is immediate. We have two words with two possible values

$$\begin{matrix} 0 \\ 1 \end{matrix} \tag{6.372}$$

The list for $n = 2$ is constructed by considering first the list of $(1/2)2^2 = 2$ words that are obtained by appending a 0 in front of the words of the list (6.372):

$$\begin{matrix} 0 & 0 \\ 0 & 1 \end{matrix} \tag{6.373}$$

The remaining two words are obtained by inverting the order of the words in (6.372) and appending a 1 in front:

$$\begin{matrix} 1 & 1 \\ 1 & 0 \end{matrix} \tag{6.374}$$

The final result is the following list of words:

$$\begin{matrix} 0 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 0 \end{matrix} \tag{6.375}$$

Iterating the procedure for $n = 3$, the first 4 words are obtained by repeating the list (6.375) and appending a 0 in front of the words of the list.

Inverting then the order of the list (6.375) and appending a 1 in front, the final result is the list of 8 words

$$\begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{matrix} \tag{6.376}$$

It is easy to extend this procedure to any value of n. By induction it is just as easy to prove that two adjacent words in each list differ by one bit at most.

Appendix 6.C *Baseband PPM and PDM*

In addition to the widely known PAM, two other baseband pulse modulation techniques are *pulse position modulation* (PPM) and *pulse duration modulation* (PDM).

PPM consists of a set of pulses whose shift, with respect to a given time reference, depends on the value of the transmitted symbol. We consider the fundamental pulse shape of Figure 6.67 and an alphabet given by

$$\mathcal{A} = \{0, 1, 2, 3, \ldots, M-1\} \tag{6.377}$$

The transmitted isolated pulse is

$$s_n(t) = g_0\left(t - n\frac{T}{M}\right) \qquad n \in \mathcal{A} \tag{6.378}$$

For $M = 4$ the set of waveforms is represented in Figure 6.68.

In PDM, instead, the input symbol determines the duration of the transmitted pulse, that is a multiple of a minimum time duration equal to T/M.

For an alphabet

$$\mathcal{A} = \{1, 2, \ldots, M\} \tag{6.379}$$

the transmitted isolated pulse is given by

$$s_n(t) = g_0\left(\frac{t}{n}\right) \qquad n \in \mathcal{A} \tag{6.380}$$

where g_0 is given in Figure 6.67.

The set of PDM waveforms for $M = 4$ is illustrated in Figure 6.69.

Signal-to-noise ratio

In both PPM and PDM the information lies in the position of the fronts of the transmitted pulses: therefore demodulation consists in finding the fronts of the pulses, which are disturbed by noise.

If the channel bandwidth were infinite, one could receive perfectly rectangular pulses. In practice the received pulse, with amplitude equal to A, has a rise time t_R different from

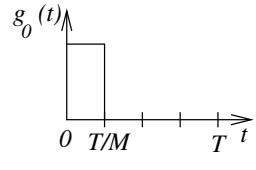


Figure 6.67. Fundamental pulse shape of PPM.

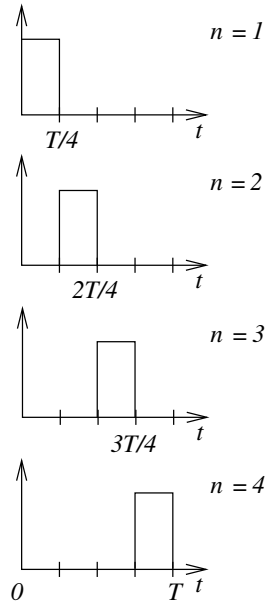


Figure 6.68. PPM waveforms for $M = 4$.

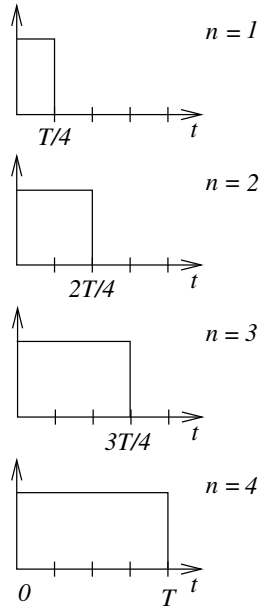


Figure 6.69. PDM waveforms for $M = 4$.

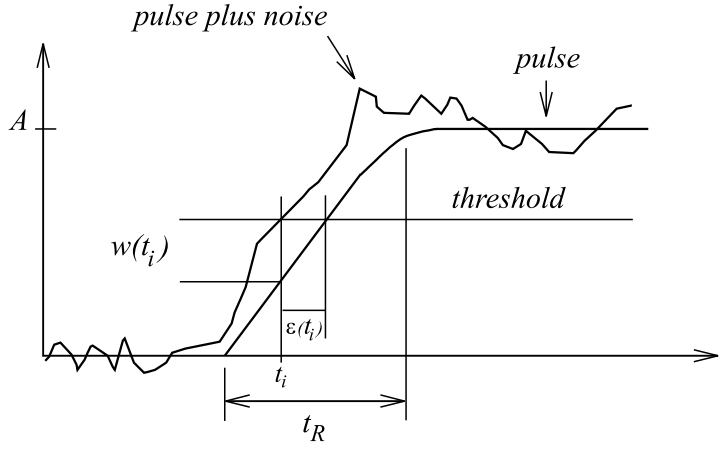


Figure 6.70. PPM and PDM demodulation in the presence of noise.

zero, as the channel has a finite bandwidth B, and noise disturbs the reception of the pulse, as illustrated in Figure 6.70. The detection of the front of a pulse is obtained by establishing the instant t_i in which the received signal, pulse plus noise, crosses a given threshold. The error $\varepsilon(t_i)$ is related to the noise $w(t_i)$, amplitude A, and rise time t_R of the received pulse:

$$\frac{\varepsilon(t_i)}{t_R} = \frac{w(t_i)}{A} \tag{6.381}$$

Assuming the noise stationary with PSD $N_0/2$ over the channel passband with bandwidth B, the mean-square error is given by

$$E[\varepsilon^2] = \left(\frac{t_R}{A}\right)^2 E[w^2] = \left(\frac{t_R}{A}\right)^2 N_0 B \tag{6.382}$$

We consider the following approximated expression that links the rise time to the bandwidth of the pulse

$$t_R \simeq \frac{1}{2B} \tag{6.383}$$

Substitution of the above result in (6.382) yields

$$E[\varepsilon^2] \simeq \frac{N_0}{4A^2 B} \tag{6.384}$$

On the other hand, assuming the average duration of the pulses is τ_0, the signal-to-noise ratio is given by (6.105) with $B_{min} = 1/(2T)$, that is

$$\Gamma = \frac{2E_{s_{Ch}}}{N_0} \tag{6.385}$$

where

$$E_{s_{Ch}} = \tau_0 A^2 \tag{6.386}$$

Finally, substitution of (6.385) in (6.384) yields

$$E[\varepsilon^2] = \frac{\tau_0}{2B} \frac{1}{\Gamma} \tag{6.387}$$

For a more in-depth analysis we refer to [20], where a trade-off between the channel bandwidth and the signal-to-noise ratio at the decision point (see (6.387)) is illustrated.

Appendix 6.D *Walsh codes*

We illustrate a procedure to obtain *orthogonal* binary sequences, with values $\{-1, 1\}$, of length 2^m.

We consider $2^m \times 2^m$ Hadamard matrices $\mathbf{A}_m$, with binary elements from the set $\{0, 1\}$. For the first orders, we have

$$\mathbf{A}_0 = [0] \tag{6.388}$$

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{6.389}$$

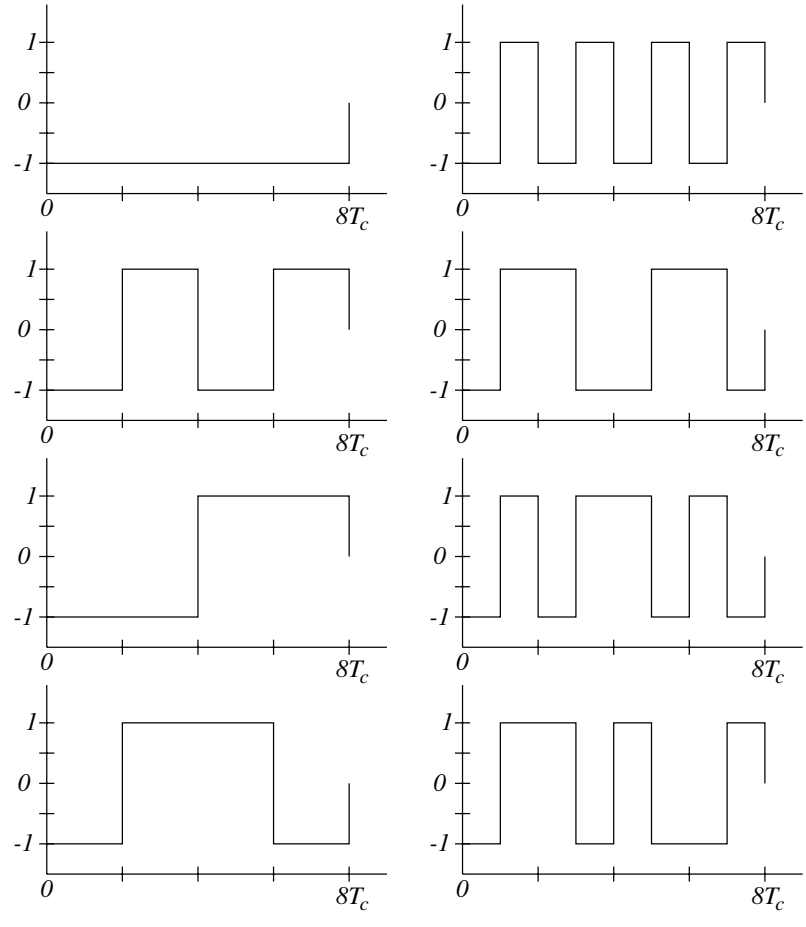


Figure 6.71. Eight orthogonal signals obtained from the Walsh code of length 8.

$$\mathbf{A}_2=\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \tag{6.390}$$

$$\mathbf{A}_3=\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{6.391}$$

In general the construction is recursive

$$\mathbf{A}_{m+1} = \begin{bmatrix} \mathbf{A}_m & \mathbf{A}_m \\ \mathbf{A}_m & \bar{\mathbf{A}}_m \end{bmatrix} \tag{6.392}$$

where $\bar{\mathbf{A}}_m$ denotes the matrix that is obtained by taking the 1's complement of the elements of $\mathbf{A}_m$.

A Walsh code of length 2^m is obtained by taking the rows (or columns) of the Hadamard matrix $\mathbf{A}_m$ and by mapping 0 into -1. From the construction of Hadamard matrices, it is easily seen that two words of a Walsh code are orthogonal.

Figure 6.71 shows the 8 signals obtained with the Walsh code of length 8: the signals are obtained by interpolating the Walsh code sequences by a filter having impulse response

$$\mathrm{w}_{T_c}(t) = \mathrm{rect}\,\frac{t - T_c/2}{T_c} \tag{6.393}$$

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 7

Transmission over dispersive channels

In this chapter we will reconsider amplitude modulation (PAM and QAM, see Chapter 6) for continuous transmission, taking into account the possibility that the transmission channel may distort the transmitted signal [1, 2]. We will also consider the effects of errors introduced by the digital transmission on a PCM encoded message (see Chapter 5) [3].

7.1 Baseband digital transmission (PAM systems)

Let us briefly examine the fundamental blocks of the baseband digital transmission system illustrated in Figure 7.1. The signals at various points of a ternary PAM system are shown in Figure 7.2.

Source. We assume that a source, not explicitly indicated in Figure 7.1, generates a message $\{b_\ell\}$ composed of a sequence of binary symbols $b_\ell \in \{0, 1\}$, that are emitted every T_b seconds:

$$\{b_\ell\} = \{\ldots, b_{-1}, b_0, b_1, b_2, \ldots\} \tag{7.1}$$

Usually $\{b_\ell\}$ is a sequence of i.i.d. symbols.

The system bit rate, which is associated with the message $\{b_\ell\}$, is equal to (see Section 6.2)

$$R_b = \frac{1}{T_b} \text{ (bit/s)} \tag{7.2}$$

Transmitter

Bit mapper. The bit mapper uses a one-to-one map to match a multilevel symbol to an input bit pattern. Let us consider, for example, symbols $\{a_k\}$ from a quaternary alphabet, $a_k \in \mathcal{A} = \{-3, -1, 1, 3\}$. To select the values of a_k we consider pairs of input bits and map them into quaternary symbols as indicated in Table 7.1. Note that bits are mapped into symbols without introducing redundancy, therefore we speak of uncoded transmission, or in other words the sequence of symbols $\{a_k\}$ is not obtained by applying channel coding.[1] This situation will be maintained throughout the chapter.

[1] We distinguish three types of coding: 1) source or entropy coding; 2) channel coding; and 3) line coding. Their objectives are respectively: 1) "compress" the digital message by lowering the bit rate without losing the original signal information (see Chapter 5); 2) increase the "reliability" of the transmission by inserting redundancy in the transmitted message, so that errors can be detected and/or corrected at the receiver (see Chapters 11 and 12); and 3) "shape" the spectrum of the transmitted signal (see Appendix 7.A).

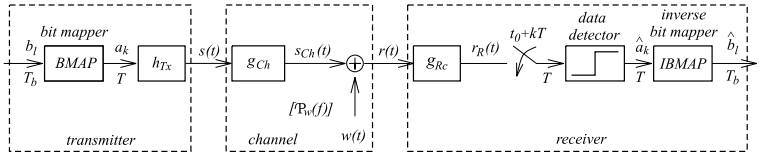


Figure 7.1. Block diagram of a baseband digital transmission system.

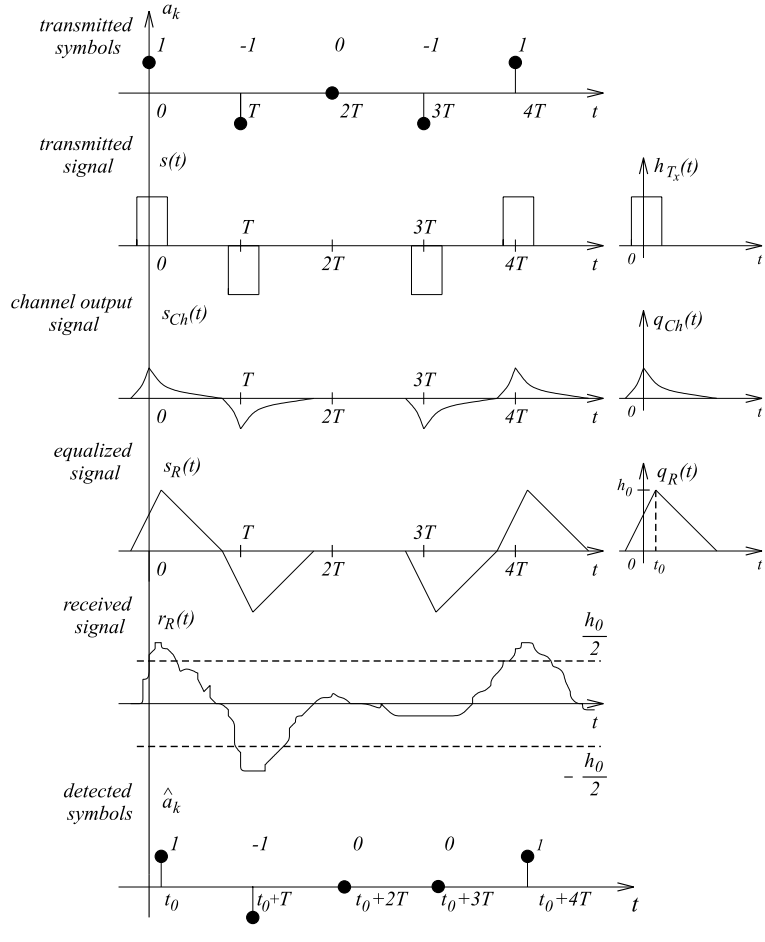


Figure 7.2. Signals at various points of a ternary PAM transmission system with alphabet $\mathcal{A} = \{-1, 0, 1\}$.

For uncoded quaternary transmission the symbol period or modulation interval T is given by $T = 2T_b$. $1/T$ is the modulation rate or symbol rate of the system and is measured in Baud: it indicates the number of symbols per second that are transmitted. In general, if the values of a_k belong to an alphabet $\mathcal{A}$ with M elements, then

$$\frac{1}{T} = \frac{1}{\log_2 M} \frac{1}{T_b} \text{ (Baud)} \tag{7.3}$$

Table 7.1 Example of quaternary bit map.

b_{2k}	b_{2k+1}	a_k
0	0	-3
1	0	-1
1	1	1
0	1	3

We note that in Section 6.3 we considered an alphabet whose elements were indices, that is $a_k \in \{1, 2, \ldots, M\}$. Now the values of a_k are associated with $\{\alpha_n\}$, $n = 1, \ldots, M$, that is $a_k \in \mathcal{A} = \{-(M-1), \ldots, -1, 1, \ldots, (M-1)\}$.

Modulator. For a PAM system, see (6.109), the modulator associates the symbol a_k with the amplitude of a given pulse h_{Tx}:

$$a_k \longrightarrow a_k h_{Tx}(t - kT) \tag{7.4}$$

Therefore the modulated signal $s(t)$ that is input to the transmission channel is given by

$$s(t) = \sum_{k=-\infty}^{+\infty} a_k h_{Tx}(t - kT) \tag{7.5}$$

Transmission channel

The transmission channel is assumed to be *linear* and *time invariant*, with impulse response g_{Ch}. Therefore the desired signal at the output of the transmission channel still has a PAM structure. From the relation

$$s_{Ch}(t) = g_{Ch} * s(t) \tag{7.6}$$

we define

$$q_{Ch}(t) = h_{Tx} * g_{Ch}(t) \tag{7.7}$$

then we have

$$s_{Ch}(t) = \sum_{k=-\infty}^{+\infty} a_k q_{Ch}(t - kT) \tag{7.8}$$

The transmission channel introduces an effective noise w. Therefore the signal at the input of the receive filter is given by:

$$r(t) = s_{Ch}(t) + w(t) \tag{7.9}$$

Receiver

The receiver consists of three functional blocks:

1. *Amplifier-equalizer filter*. This block is assumed *linear* and *time invariant* with impulse response g_{Rc}. Then the desired signal is given by:

$$s_R(t) = g_{Rc} * s_{Ch}(t) \tag{7.10}$$

   Let the *overall impulse response* of the system be

$$q_R(t) = q_{Ch} * g_{Rc}(t) = h_{Tx} * g_{Ch} * g_{Rc}(t) \tag{7.11}$$

   then

$$s_R(t) = \sum_{k=-\infty}^{+\infty} a_k q_R(t - kT) \tag{7.12}$$

   In the presence of noise,

$$r_R(t) = s_R(t) + w_R(t) \tag{7.13}$$

   where $w_R(t) = w * g_{Rc}(t)$.

2. *Sampler*. If the duration of $q_R(t)$ is confined to a modulation interval, then the various pulses do not overlap and, in the absence of noise, sampling at instants t_0+kT yields:[2]

$$r_k = r_R(t_0 + kT) = a_k q_R(t_0) = a_k h_0 \tag{7.14}$$

   where $h_0 = q_R(t_0)$ is the amplitude of the overall impulse response at the sampling instant t_0. The parameter t_0 is called *timing phase*, and its choice is fundamental for system performance.

3. *Threshold detector*. From the sequence $\{r_k\}$ we detect the transmitted sequence $\{a_k\}$. The simplest structure is the instantaneous non-linear threshold detector:

$$\hat{a}_k = Q[r_k] \tag{7.15}$$

   where $Q[r_k]$ is the quantizer characteristic with $r_k \in \Re$ and $\hat{a}_k \in \mathcal{A}$, alphabet of a_k. An example of quantizer characteristic for $\mathcal{A} = \{-1, 0, 1\}$ is given in Figure 7.3.

From the sequence $\{\hat{a}_k\}$, using an inverse bit mapper, the detected binary information message $\{\hat{b}_\ell\}$ is obtained.

[2] To simplify the notation, the sample index k, associated with the instant $t_0 + kT$, here appears as a subscript.

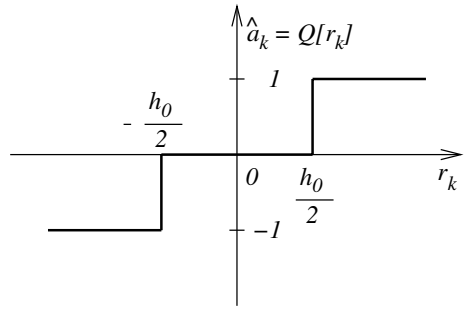


Figure 7.3. Characteristic of a threshold detector for ternary symbols with alphabet $\mathcal{A} = \{-1, 0, 1\}$, and amplitude h_0 of the overall impulse response.

We recall that the receiver structure described above was optimized in Chapter 6 for an ideal AWGN channel.

Power spectral density of a PAM signal

The PAM signals found in various points of the system, $s_R(t)$, $s_{Ch}(t)$, and $s(t)$, have the following structure:

$$s(t) = \sum_{k=-\infty}^{+\infty} a_k q(t - kT) \tag{7.16}$$

where $q(t)$ is the impulse response of a suitable filter. In other words, a PAM signal $s(t)$ may be regarded as a signal generated by an interpolator filter with impulse response $q(t)$, $t \in \Re$, as shown in Figure 7.4.

From the spectral analysis (see Example 1.9.9 on page 69) we know that s is a *cyclostationary* process with *average power spectral density* given by (see (1.398)):

$$\bar{\mathcal{P}}_s(f) = \left|\frac{1}{T}\mathcal{Q}(f)\right|^2 \mathcal{P}_a(f) \tag{7.17}$$

where $\mathcal{P}_a$ is the spectral density of the message and $\mathcal{Q}$ is the Fourier transform of q.

From Figure 7.4 it is important to verify that by filtering $s(t)$ we obtain a signal that is still PAM, with a pulse given by the convolution of the filter impulse responses. The spectral density of a filtered PAM signal is obtained by multiplying $\mathcal{P}_a(f)$ in (7.17) by the squared magnitude of the filter frequency response. As $\mathcal{P}_a(f)$ is periodic of period $1/T$, then the *bandwidth B of the transmitted signal* is equal to that of h_{Tx}.

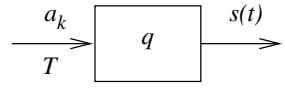


Figure 7.4. The PAM signal as output of an interpolator filter.

Occasionally the passband version of PAM that is obtained by DSB modulation is also proposed. Passband PAM, together with the associated parameters, is discussed in Appendix 7.C.

Example 7.1.1 (PSD of an i.i.d. symbol sequence)
Let $\{a_k\}$ be a sequence of i.i.d. symbols with values from the alphabet

$$\mathcal{A} = \{\alpha_1, \alpha_2, \ldots, \alpha_M\} \tag{7.18}$$

and $p(\alpha)$, $\alpha \in \mathcal{A}$, be the probability distribution of each symbol.

The mean value and the statistical power of the sequence are given by

$$\mathtt{m}_a = \sum_{\alpha \in \mathcal{A}} \alpha p(\alpha) \qquad \mathtt{M}_a = \sum_{\alpha \in \mathcal{A}} |\alpha|^2 p(\alpha) \tag{7.19}$$

Consequently $\sigma_a^2 = \mathtt{M}_a - |\mathtt{m}_a|^2$.

Following Example 1.7.1 on page 51, that describes the decomposition of the PSD of a message into ordinary and impulse functions, the decomposition of the PSD of s is given by

$$\bar{\mathcal{P}}_s^{(c)}(f) = \left|\frac{1}{T}\mathcal{Q}(f)\right|^2 \sigma_a^2 T = |\mathcal{Q}(f)|^2 \frac{\sigma_a^2}{T} \tag{7.20}$$

and

$$\bar{\mathcal{P}}_s^{(d)}(f) = \left|\frac{1}{T}\mathcal{Q}(f)\right|^2 |\mathtt{m}_a|^2 \sum_{\ell=-\infty}^{+\infty} \delta\left(f - \frac{\ell}{T}\right) \tag{7.21}$$

$$= \left|\frac{\mathtt{m}_a}{T}\right|^2 \sum_{\ell=-\infty}^{+\infty} \left|\mathcal{Q}\left(\frac{\ell}{T}\right)\right|^2 \delta\left(f - \frac{\ell}{T}\right) \tag{7.22}$$

We note that spectral lines occur in the PSD of s if $\mathtt{m}_a \neq 0$ and $\mathcal{Q}(f)$ is non-zero at frequencies multiple of $1/T$.

Typically the presence of spectral lines is not desirable for a transmission scheme. We obtain $\mathtt{m}_a = 0$ by choosing an alphabet with symmetric values with respect to zero, and by assigning equal probabilities to antipodal values.

In some applications, the spectrum of the transmitted signal is shaped by introducing correlation among transmitted symbols by a *line encoder*, see (7.17): in this case the sequence has memory. We refer the reader to Appendix 7.A for a description of the more common line codes.

7.2 Passband digital transmission (QAM systems)

We consider the QAM transmission system illustrated in Figure 7.5.

Transmitter

Bit mapper. The bit mapper uses a map to associate a complex-valued symbol a_k to an input bit pattern. In Figure 7.6 are given two examples of constellations and corresponding

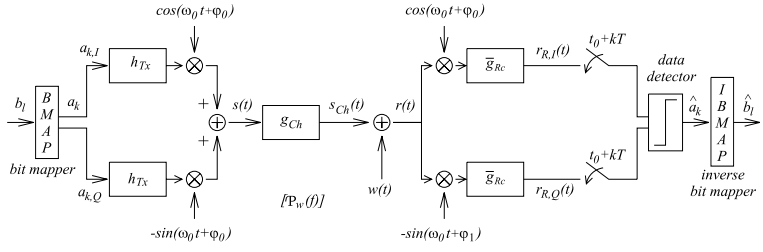
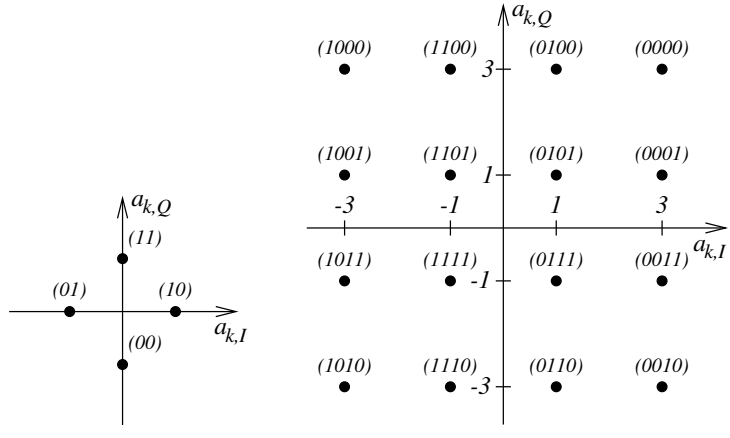


Figure 7.5. Block diagram of a passband digital transmission system.



(a) QPSK. (b) 16-QAM.

Figure 7.6. Two constellations and corresponding bit map.

binary representations, where

$$a_k = a_{k,I} + j a_{k,Q} \tag{7.23}$$

and $a_{k,I} = Re\,[a_k]$ and $a_{k,Q} = Im\,[a_k]$. 4-PSK (or QPSK) symbols are taken from an alphabet with four elements, each identified by two bits. Similarly each element in a 16-QAM constellation is uniquely identified by four bits.

Modulator. Typically the pulse h_{Tx} is real-valued, however, the baseband modulated signal is complex-valued:

$$\begin{aligned} s^{(bb)}(t) &= \sum_{k=-\infty}^{+\infty} a_k h_{Tx}(t - kT) \\ &= \sum_{k=-\infty}^{+\infty} a_{k,I} h_{Tx}(t - kT) + j \sum_{k=-\infty}^{+\infty} a_{k,Q} h_{Tx}(t - kT) \end{aligned} \tag{7.24}$$

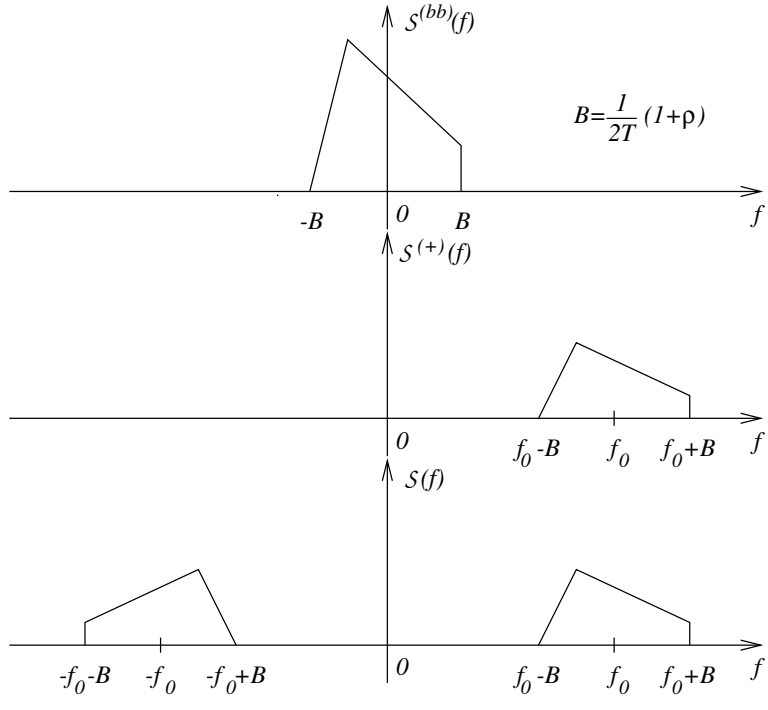


Figure 7.7. Fourier transforms of baseband signal and modulated signal.

Let f_0 ($\omega_0 = 2\pi f_0$) and φ_0 be, respectively, the carrier frequency (radian frequency) and phase. We define

$$s^{(+)}(t) = \tfrac{1}{2}s^{(bb)}(t)e^{j(\omega_0 t+\varphi_0)} \xleftrightarrow{\mathcal{F}} \mathcal{S}^{(+)}(f) = \tfrac{1}{2}\mathcal{S}^{(bb)}(f - f_0)e^{j\varphi_0} \tag{7.25}$$

then the real-valued transmitted signal is given by:

$$s(t) = 2Re\{s^{(+)}(t)\} \xleftrightarrow{\mathcal{F}} \mathcal{S}(f) = \mathcal{S}^{(+)}(f) + \mathcal{S}^{(+)*}(-f) \tag{7.26}$$

The transformation in the frequency domain from $s^{(bb)}$ to s is illustrated in Figure 7.7.

Power spectral density of a QAM signal

From the analysis leading to (1.395), $s^{(bb)}$ is a cyclostationary process of period T with an average PSD (see (1.398)):

$$\bar{\mathcal{P}}_{s^{(bb)}}(f) = \left|\frac{1}{T}\mathcal{H}_{Tx}(f)\right|^2 \mathcal{P}_a(f) \tag{7.27}$$

Moreover, starting from a relation similar to (1.304), we get that s is a cyclostationary random process with an average PSD given by[3]

$$\bar{\mathcal{P}}_s(f) = \tfrac{1}{4}[\bar{\mathcal{P}}_{s^{(bb)}}(f - f_0) + \bar{\mathcal{P}}_{s^{(bb)}}(-f - f_0)] \tag{7.28}$$

We note that the bandwidth B of the transmitted signal is equal to *twice* the bandwidth of h_{Tx}.

Three equivalent representations of the modulator

1. From (7.25) and (7.26), using (7.24), it turns out

$$\begin{aligned} s(t) &= Re[s^{(bb)}(t)e^{j(\omega_0 t+\varphi_0)}] \\ &= Re\left[e^{j(\omega_0 t+\varphi_0)} \sum_{k=-\infty}^{+\infty} a_k h_{Tx}(t - kT)\right] \end{aligned} \tag{7.29}$$

The block-diagram representation of (7.29) is shown in Figure 7.8. As $s^{(bb)}$ is in general a complex-valued signal, an implementation based on this representation requires a processor capable of complex arithmetic.

2. As

$$e^{j(\omega_0 t+\varphi_0)} = \cos(\omega_0 t + \varphi_0) + j\sin(\omega_0 t + \varphi_0) \tag{7.30}$$

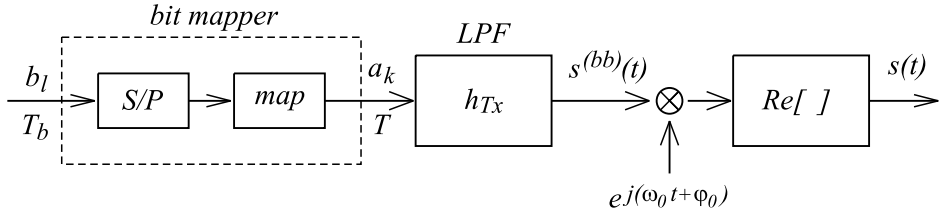


Figure 7.8. QAM transmitter: complex-valued representation.

---

[3] The result (7.28) needs clarification. We first consider the situation where the condition $r_{s^{(bb)}s^{(bb)*}}(t, t-\tau) = 0$ is satisfied, as for example in the case of QAM with i.i.d. circularly symmetric symbols (see (1.407)). From the equation (similar to (1.304)) that relates r_s to $r_{s^{(bb)}}$ and $r_{s^{(bb)}s^{(bb)*}}$, as the cross-correlations are zero, we find that the process s is cyclostationary in t of period T. Taking the average correlation in a period T, the results (7.27) and (7.28) follow.

We now consider the situation where $r_{s^{(bb)}s^{(bb)*}}(t, t-\tau) \neq 0$, and in particular the case where $r_{s^{(bb)}s^{(bb)*}}(t, t-\tau)$ is a periodic function in t of period T, as for example in the case of PAM-DSB (see Appendix 7.C). In this situation the cross-correlations, in the equation similar to (1.304), do not vanish. If a real value T_p exists, such that T_p is an integer multiple of both T and $1/f_0$, then s is cyclostationary in t of period equal to T_p. Taking the average correlation over the period T_p, and expanding $r_{s^{(bb)}s^{(bb)*}}(t, t-\tau)$ in Fourier series (in the variable t), for $1/T \ll f_0$ it happens that the autocorrelation terms approximate the same terms found in the previous case, and the cross-correlation terms become negligible.

(7.29) becomes:

$$s(t) = \cos(\omega_0 t + \varphi_0) \sum_{k=-\infty}^{+\infty} a_{k,I} h_{Tx}(t - kT) - \sin(\omega_0 t + \varphi_0) \sum_{k=-\infty}^{+\infty} a_{k,Q} h_{Tx}(t - kT) \tag{7.31}$$

The block-diagram representation of (7.31) is shown in Figure 7.5 (see also Figure 6.38). The implementation of a QAM transmitter based on (7.31) is discussed in Appendix 7.D.

3. Using the polar notation $a_k = |a_k| e^{j\theta_k}$, (7.29) takes the form:

$$\begin{aligned} s(t) &= Re\left[e^{j(\omega_0 t + \varphi_0)} \sum_{k=-\infty}^{+\infty} |a_k| e^{j\theta_k} h_{Tx}(t - kT) \right] \\ &= Re\left[\sum_{k=-\infty}^{+\infty} |a_k| e^{j(\omega_0 t + \varphi_0 + \theta_k)} h_{Tx}(t - kT) \right] \\ &= \sum_{k=-\infty}^{+\infty} |a_k| \cos(\omega_0 t + \varphi_0 + \theta_k) h_{Tx}(t - kT) \end{aligned} \tag{7.32}$$

If $|a_k|$ is a constant we obtain the PSK signal (6.127), where the information bits select only the value of the carrier phase.

Coherent receiver

In the *absence of noise* the general scheme of a coherent receiver is shown in Figure 7.9, which follows the scheme of Figure 6.40.

The received signal is given by:

$$s_{Ch}(t) = s * g_{Ch}(t) \xleftrightarrow{\mathcal{F}} \mathcal{S}_{Ch}(f) = \mathcal{S}(f)\mathcal{G}_{Ch}(f) \tag{7.33}$$

First, the received signal is translated to baseband by a frequency shift,

$$s_{M_0}(t) = s_{Ch}(t) e^{-j(\omega_0 t + \varphi_1)} \xleftrightarrow{\mathcal{F}} \mathcal{S}_{M_0}(f) = \mathcal{S}_{Ch}(f + f_0) e^{-j\varphi_1} \tag{7.34}$$

then it is filtered by a lowpass filter (LPF), $\overline{g}_{Rc}$,

$$s_R(t) = s_{M_0} * \overline{g}_{Rc}(t) \xleftrightarrow{\mathcal{F}} \mathcal{S}_R(f) = \mathcal{S}_{M_0}(f)\overline{\mathcal{G}}_{Rc}(f) \tag{7.35}$$

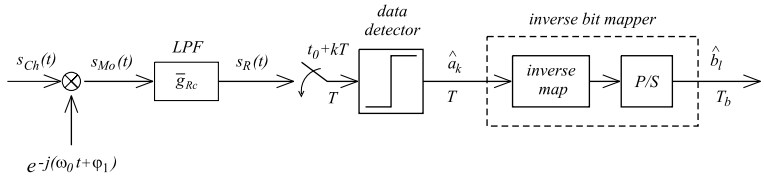


Figure 7.9. QAM receiver: complex-valued representation.

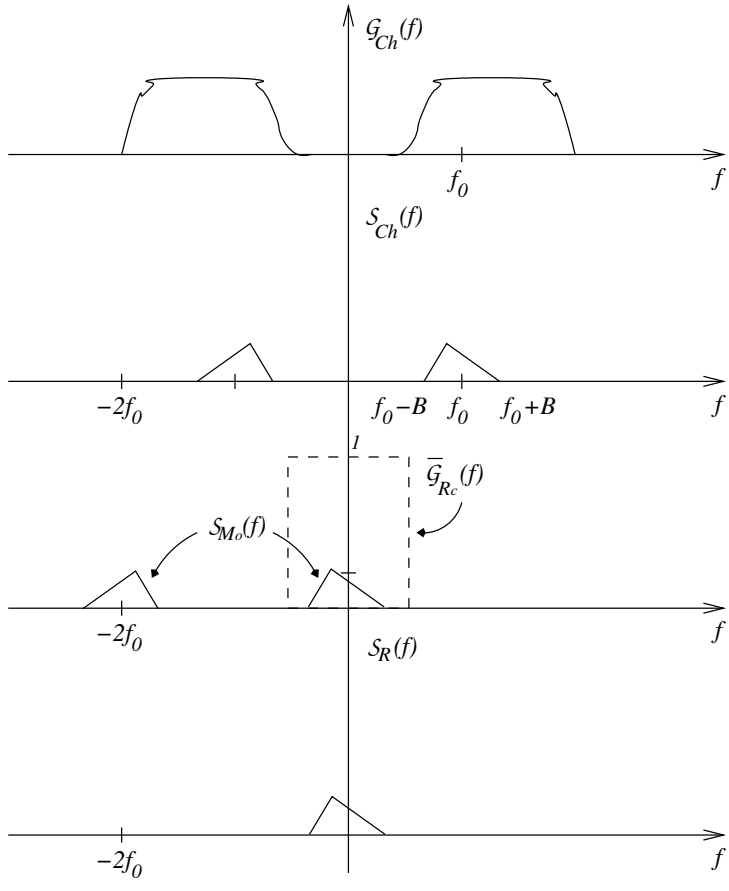


Figure 7.10. Frequency responses of the channel and of signals at various points of the receiver.

We note that, if $\overline{g}_{Rc}$ is a non-distorting ideal filter with unit gain, then $s_R(t) = (1/2)s_{Ch}^{(bb)}(t)$. In the particular case where $\overline{g}_{Rc}$ is a real-valued filter, then the receiver in Figure 7.9 is simplified into that of Figure 7.5. Figure 7.10 illustrates these transformations.

We note that in the above analysis, as the channel may introduce a phase offset, the receive carrier phase φ_1 may be different from the transmit carrier phase φ_0.

7.3 Baseband equivalent model of a QAM system

Recalling the relations of Figure 1.30, we illustrate the baseband equivalent scheme with reference to Figure 7.11:[4] by assuming that the transmit and receive carriers have the same

[4] We note that the term $e^{j\varphi_0}$ has been moved to the receiver; therefore the signals $s^{(bb)}$ and $r^{(bb)}$ of Figure 7.11 are defined apart from the term $e^{j\varphi_0}$. This is the same as assuming as reference carrier $e^{j(2\pi f_0 t+\varphi_0)}$.

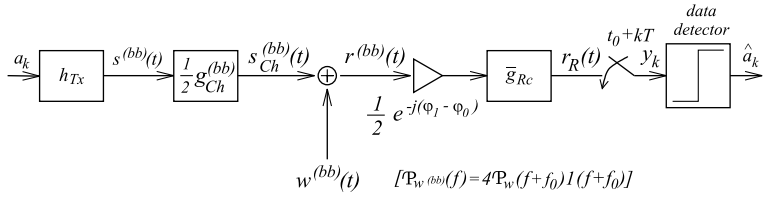


Figure 7.11. Baseband equivalent model of a QAM transmission system.

frequency, we can study QAM systems by the same method that we have developed for PAM systems.

7.3.1 Signal analysis

We refer to Section 1.7.4 for an analysis of passband signals; we recall here that if for $f > 0$ the spectral density of w, $\mathcal{P}_w(f)$, is an even function around the frequency f_0, then the real and imaginary parts of $w^{(bb)}(t) = w_I(t) + jw_Q(t)$ have a power spectral density that is given by

$$\mathcal{P}_{w_I}(f) = \mathcal{P}_{w_Q}(f) = \frac{1}{2}\mathcal{P}_{w^{(bb)}}(f) = \begin{cases} 2\mathcal{P}_w(f+f_0) & f \geq -f_0 \\ 0 & \text{elsewhere} \end{cases} \tag{7.36}$$

Moreover, $\mathcal{P}_{w_I w_Q}(f) = 0$, that is $w_I \perp w_Q$, hence

$$\sigma_{w_I}^2 = \sigma_{w_Q}^2 = \tfrac{1}{2}\sigma_{w^{(bb)}}^2 = \sigma_w^2 \tag{7.37}$$

To simplify the analysis, for the study of a QAM system we will adopt the PAM model of Figure 7.1, assuming that all signals and filters are in general complex. We note that the factor $(1/2)e^{-j(\varphi_1-\varphi_0)}$ appears in Figure 7.11. We will include the factor $e^{-j(\varphi_1-\varphi_0)}/\sqrt{2}$ in the impulse response of the transmission channel g_{Ch}, and the factor $1/\sqrt{2}$ in the impulse response g_{Rc}. Consequently the additive noise has a spectral density equal to $(1/2)\mathcal{P}_{w^{(bb)}}(f) = 2\mathcal{P}_w(f+f_0)$ for $f \geq -f_0$. Therefore the scheme of Figure 7.1 holds also for QAM in the presence of additive noise: the only difference is that in the case of a QAM system the noise is complex-valued with orthogonal in-phase and quadrature components, each having spectral density $\mathcal{P}_w(f+f_0)$ for $f \geq -f_0$.

Hence the scheme of Figure 7.12 is a reference scheme for both PAM and QAM, where

$$\mathcal{G}_C(f) = \begin{cases} \mathcal{G}_{Ch}(f) & \text{for PAM} \\ \dfrac{e^{-j(\varphi_1-\varphi_0)}}{\sqrt{2}}\mathcal{G}_{Ch}(f+f_0)1(f+f_0) & \text{for QAM} \end{cases} \tag{7.38}$$

We note that for QAM we have

$$g_C(t) = \frac{e^{-j(\varphi_1-\varphi_0)}}{2\sqrt{2}} g_{Ch}^{(bb)}(t) \tag{7.39}$$

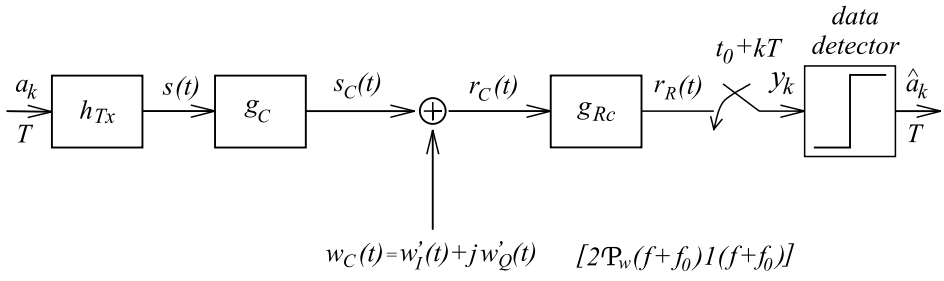


Figure 7.12. Baseband equivalent model of PAM and QAM transmission systems.

With reference to the scheme of Figure 7.9, the relation between the impulse responses of the receive filters is given by

$$g_{Rc}(t) = \frac{1}{\sqrt{2}}\overline{g}_{Rc}(t) \tag{7.40}$$

In the following, to simplify the notation, the filter $\overline{g}_{Rc}$ will be indicated in many passband schemes simply as g_{Rc}.

We summarize the definitions of the various signals in QAM systems.

1. *Sequence of input symbols,* $\{a_k\}$, sequence of symbols with values from a complex-valued alphabet $\mathcal{A}$. In PAM systems, the symbols of the sequence $\{a_k\}$ assume real values.

2. *Modulated signal*,[5]

$$s(t) = \sum_{k=-\infty}^{+\infty} a_k h_{Tx}(t - kT) \tag{7.41}$$

3. *Signal at the channel output*,

$$s_C(t) = \sum_{k=-\infty}^{+\infty} a_k q_C(t - kT) \qquad q_C(t) = h_{Tx} * g_C(t) \tag{7.42}$$

4. *Circularly-symmetric, complex-valued, additive Gaussian noise,* $w_C(t) = w'_I(t) + jw'_Q(t)$, with spectral density $\mathcal{P}_{w_C}$. In the case of white noise it is:[6]

$$\mathcal{P}_{w'_I}(f) = \mathcal{P}_{w'_Q}(f) = \frac{N_0}{2}\ (\mathrm{V}^2/\mathrm{Hz}) \tag{7.43}$$

and

$$\mathcal{P}_{w_C}(f) = N_0\ (\mathrm{V}^2/\mathrm{Hz}) \tag{7.44}$$

In the model of PAM systems, only the component w'_I is considered.

---

5 We point out that for QAM $s(t)$ is in fact $s^{(bb)}(t)$.

6 In fact (7.43) should include the condition $f > -f_0$. Because the bandwidth of g_{Rc} is smaller than f_0, this condition can be omitted.

5. *Received or observed signal*,

$$r_C(t) = s_C(t) + w_C(t) \tag{7.45}$$

6. *Signal at the output of the complex-valued amplifier-equalizer filter* g_{Rc},

$$r_R(t) = s_R(t) + w_R(t) \tag{7.46}$$

where

$$s_R(t) = \sum_{k=-\infty}^{+\infty} a_k q_R(t - kT) \tag{7.47}$$

with

$$q_R(t) = q_C * g_{Rc}(t) \quad \text{and} \quad w_R(t) = w_C * g_{Rc}(t) \tag{7.48}$$

In PAM systems, g_{Rc} is a real-valued filter.

7. *Signal at the decision point at instant* $t_0 + kT$,

$$y_k = r_R(t_0 + kT) \tag{7.49}$$

8. *Sequence of detected symbols*,

$$\{\hat{a}_k\} \tag{7.50}$$

Signal-to-noise ratio

The performance of a system is expressed as a function of the signal-to-noise ratio Γ defined in (6.105), that we recall here.

In general, with reference to the schemes of Figures 7.1 and 7.5, for a channel output signal, s_{Ch}, having minimum bandwidth B_{min}, and assuming the noise w white with $\mathcal{P}_w(f) = N_0/2$, we have

$$\Gamma = \frac{E[s_{Ch}^2(t)]}{(N_0/2)2B_{min}} = \frac{\mathtt{M}_{s_{Ch}}}{N_0 B_{min}} = \frac{E_{s_{Ch}}}{N_0(B_{min}T)} \tag{7.51}$$

We express now $E_{s_{Ch}}$ in the cases of PAM and QAM systems.

PAM systems. For an i.i.d. input symbol sequence, using (1.399) we have

$$E_{s_{Ch}} = \mathtt{M}_a E_{q_{Ch}} \tag{7.52}$$

and, for $B_{min} = 1/(2T)$, we obtain

$$\Gamma = \frac{\mathtt{M}_a E_{q_{Ch}}}{N_0/2} \tag{7.53}$$

where $\mathtt{M}_a$ is the statistical power of the data and $E_{q_{Ch}}$ is the energy of the pulse $q_{Ch} = h_{Tx} * g_{Ch}$. Because for PAM, observing (7.38), we get $q_{Ch} = q_C$, then (7.53) can be expressed as

$$\Gamma = \frac{\mathtt{M}_a E_{q_C}}{N_0/2} \tag{7.54}$$

QAM systems. From (1.295), using (7.38), we obtain

$$E[s_{Ch}^2(t)] = \tfrac{1}{2}E[|s_{Ch}^{(bb)}(t)|^2] = E[|s_C(t)|^2] \tag{7.55}$$

Hence, as $B_{min} = 1/T$, (7.51) becomes

$$\Gamma = \frac{\mathtt{M}_a E_{q_C}}{N_0} \tag{7.56}$$

We note that (7.56), expressed as[7]

$$\Gamma = \frac{\mathtt{M}_a E_{q_C}/2}{N_0/2} \tag{7.57}$$

represents the ratio between the energy per component of s_C, given by $TE[(Re[s_C(t)])^2]$ and $TE[(Im[s_C(t)])^2]$, and the PSD of the noise components, equal to $N_0/2$. Then (7.56), observing also (7.38), coincides with (7.53) of PAM systems.

7.3.2 Characterization of system elements

We consider some characteristics of the signals in the scheme of Figure 7.12.

Transmitter

The choice of the transmit pulse is quite important because it determines the bandwidth of the system (see (7.17) and (7.28)). Two choices are shown in Figure 7.13, where

1. $h_{Tx}(t) = \mathtt{w}_T(t) = \text{rect}\left(\frac{t-\frac{T}{2}}{T}\right)$, with wide spectrum;
2. $h_{Tx}(t)$ with longer duration and smaller bandwidth.

Transmission channel

The transmission channel is modelled as a time invariant linear system. Therefore it is represented by a filter having impulse response g_{Ch}. As described in Chapter 4, the majority of channels are characterized by frequency responses having a null at DC. Therefore the shape of the frequency response $\mathcal{G}_{Ch}(f)$ is as represented in Figure 7.14, where the passband goes from f_1 to f_2. For transmission over cables, f_1 may be of the order of a few hundred Hertz, whereas for radio links, f_1 may be in the range of MHz or GHz. Consequently, PAM (possibly using a line code) as well as QAM transmission systems may be considered over cables; for transmission over radio, instead, a PAM signal needs to be translated in frequency (PAM-DSB or PAM-SSB), or a QAM system may be used, assuming as carrier frequency f_0 the center frequency of the passband (f_1, f_2). In any case, the channel is bandlimited with a finite bandwidth $f_2 - f_1$.

[7] The term $\mathtt{M}_a E_{q_C}/2$ represents the energy of both $Re[s_C(t)]$ and $Im[s_C(t)]$, assuming that $s_C(t)$ is circularly symmetric (see (1.407)).

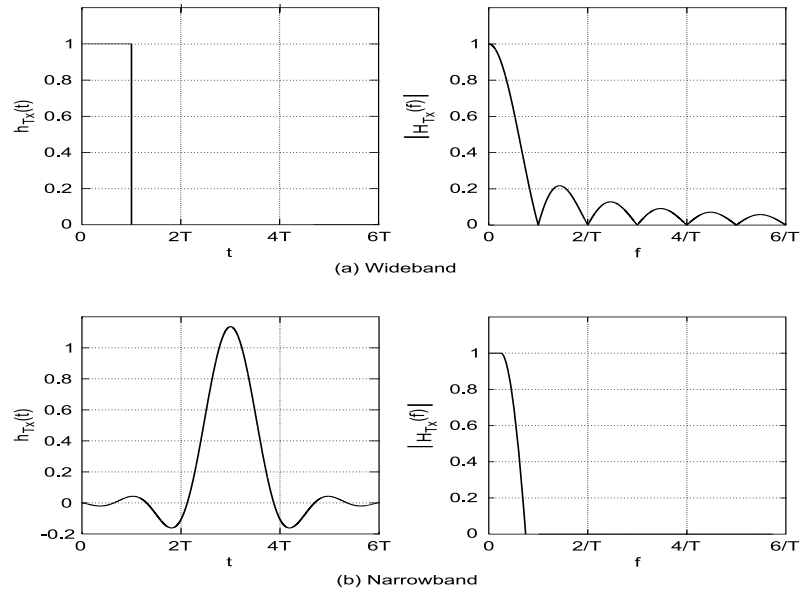


Figure 7.13. Two examples of transmit pulse h_{Tx}.

With reference to the general model of Figure 7.12, we adopt the polar notation for $\mathcal{G}_C$:

$$\mathcal{G}_C(f) = |\mathcal{G}_C(f)| e^{j \arg \mathcal{G}_C(f)} \tag{7.58}$$

Let B be the bandwidth of $s(t)$. According to (1.144), a channel presents *ideal characteristics*, known as *Heaviside conditions for the absence of distortion*, if the following two properties are satisfied:

1. the magnitude response is a constant for $|f| < B$,

$$|\mathcal{G}_C(f)| = \mathcal{G}_0 \qquad \text{for } |f| < B \tag{7.59}$$

2. the phase response is proportional to f for $|f| < B$,

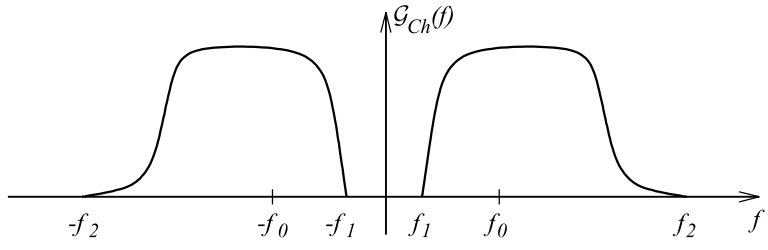
$$\arg \mathcal{G}_C(f) = -2\pi f t_0 \qquad \text{for } |f| < B \tag{7.60}$$

Under these conditions, s is reproduced at the output of the channel without distortion, that is:

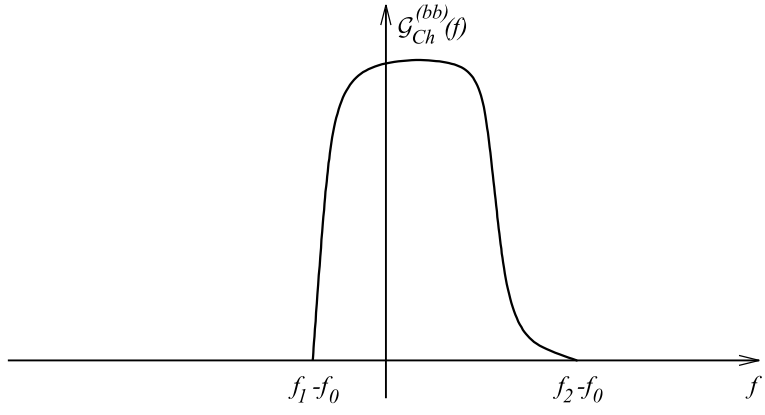
$$s_C(t) = \mathcal{G}_0 s(t - t_0) \tag{7.61}$$

In practice, channels introduce both "amplitude distortion" and "phase distortion". An example of frequency response of a radio channel is given in Figure 4.32: the overall effect is that the signal $s_C(t)$ may be very different from $s(t)$.

In short, for channels encountered in practice conditions (7.59) and (7.60) are too stringent; for PAM and QAM transmission systems we will refer instead to the Nyquist criterion (7.79).



(a) Channel



(b) Baseband equivalent channel

Figure 7.14. Frequency response of the transmission channel.

Receiver

We return to the receiver structure of Figure 7.12, consisting of a filter g_{Rc} followed by a sampler with sampling rate $1/T$, and a data detector.

In general, if the frequency response of the receive filter $\mathcal{G}_{R_c}(f)$ contains a factor $C(e^{j2\pi fT})$, periodic of period $1/T$, such that the following factorization holds:

$$\mathcal{G}_{Rc}(f) = \mathcal{G}_M(f)C(e^{j2\pi fT}) \tag{7.62}$$

where $\mathcal{G}_M(f)$ is a generic function, then the filter-sampler block before the data detector of Figure 7.12 can be represented as in Figure 7.15, where the sampler is followed by a discrete-time filter. It is easy to prove that in the two systems the relation between $r_C(t)$ and y_k is the same.

Ideally, in the system of Figure 7.15 y_k should be equal to a_k. In practice, as illustrated in Figure 7.16, linear distortion and additive noise, the only disturbances considered here, may determine a significant deviation of y_k from the desired symbol a_k.

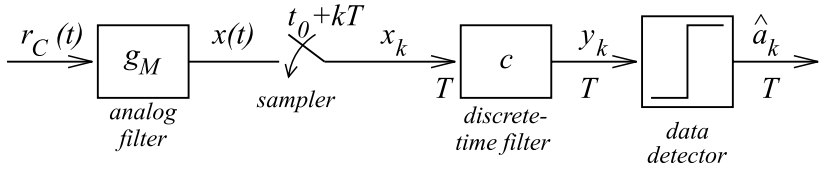
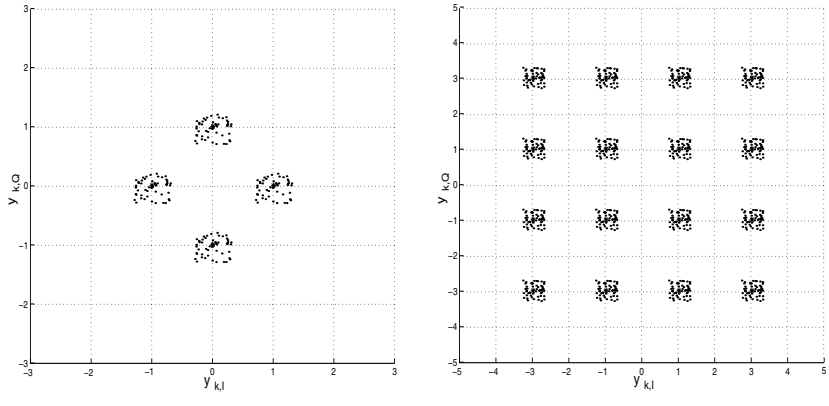


Figure 7.15. Receiver structure with analog and discrete-time filters.



(a) QPSK. (b) 16-QAM.

Figure 7.16. Values of $y_k = y_{k,I} + jy_{k,Q}$, in the presence of noise and linear distortion.

The last element in the receiver is the data detector. One of the simplest data detectors is the *threshold detector*, that associates with each value of y_k a possible value of a_k in the constellation. Using the rule of deciding for the symbol closest to the sample y_k, the decision regions for a QPSK system and a 16-QAM system are illustrated in Figure 7.17.

7.3.3 Intersymbol interference

Discrete-time equivalent system

From (7.46) we define

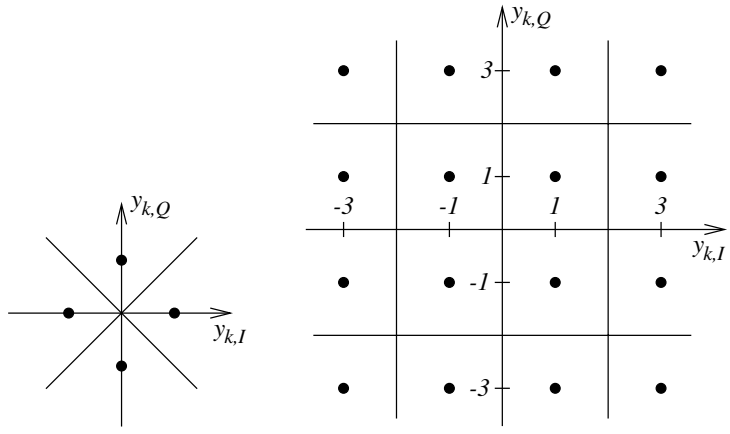
$$s_{R,k} = s_R(t_0 + kT) = \sum_{i=-\infty}^{+\infty} a_i q_R(t_0 + (k-i)T) \tag{7.63}$$

and

$$w_{R,k} = w_R(t_0 + kT) \tag{7.64}$$

Then, from (7.49), at the decision point the generic sample is expressed as

$$y_k = s_{R,k} + w_{R.k} \tag{7.65}$$



(a) QPSK. (b) 16-QAM.

Figure 7.17. Decision regions for a QPSK system and a 16-QAM system.

Introducing the version of q_R, time-shifted by t_0, as

$$h(t) = q_R(t_0 + t) \tag{7.66}$$

and defining

$$h_i = h(iT) = q_R(t_0 + iT) \tag{7.67}$$

it follows that

$$s_{R,k} = \sum_{i=-\infty}^{+\infty} a_i h_{k-i} = a_k h_0 + \mathtt{i}_k \tag{7.68}$$

where

$$\mathtt{i}_k = \sum_{i=-\infty,\, i\neq k}^{+\infty} a_i h_{k-i} = \cdots + h_{-1}a_{k+1} + h_1 a_{k-1} + h_2 a_{k-2} + \cdots \tag{7.69}$$

represents the intersymbol interference (ISI). The coefficients $\{h_i\}_{i\neq 0}$ are called *interferers*. Moreover (7.65) becomes

$$y_k = a_k h_0 + \mathtt{i}_k + w_{R,k} \tag{7.70}$$

We observe that, even in the absence of noise, the detection of a_k from y_k by a threshold detector takes place in the presence of the term $\mathtt{i}_k$, which behaves as a disturbance with respect to the desired term $a_k h_0$.

For the analysis, it is often convenient to approximate $\mathtt{i}_k$ as noise with a Gaussian distribution: the more numerous and similar in amplitude are the interferers, the more valid

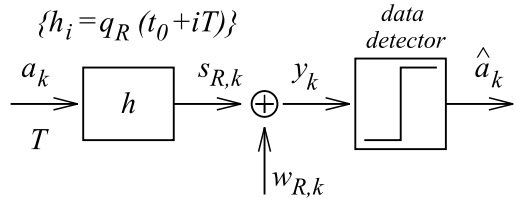


Figure 7.18. Discrete-time equivalent scheme, with period T, of a QAM system.

is this approximation. In the case of i.i.d. symbols, the first two moments of $\mathtt{i}_k$ are easily determined.

$$\textit{Mean value of } \mathtt{i}_k: \qquad \mathtt{m}_\mathtt{i} = \mathtt{m}_a \sum_{i=-\infty,\, i\neq 0}^{+\infty} h_i \tag{7.71}$$

$$\textit{Variance of } \mathtt{i}_k: \qquad \sigma_\mathtt{i}^2 = \sigma_a^2 \sum_{i=-\infty,\, i\neq 0}^{+\infty} |h_i|^2 \tag{7.72}$$

From (7.65), with $\{s_{R,k}\}$ given by (7.68), we derive the discrete-time equivalent scheme, with period T (see Figure 7.18), that relates the signal at the decision point to the data transmitted over a discrete-time channel with impulse response given by the sequence $\{h_i\}$, called *overall discrete-time equivalent impulse response* of the system.

Concerning the additive noise $\{w_{R,k}\}$,[8] being

$$\mathcal{P}_{w_R}(f) = \mathcal{P}_{w_C}(f)|\mathcal{G}_{Rc}(f)|^2 \tag{7.73}$$

the PSD of $\{w_{R,k}\}$ is given by

$$\mathcal{P}_{w_{R,k}}(f) = \sum_{\ell=-\infty}^{+\infty} \mathcal{P}_{w_R}\left(f - \ell\frac{1}{T}\right) \tag{7.74}$$

In any case, the variance of $w_{R,k}$ is equal to that of w_R and is given by

$$\sigma_{w_{R,k}}^2 = \sigma_{w_R}^2 = \int_{-\infty}^{+\infty} \mathcal{P}_{w_C}(f)|\mathcal{G}_{Rc}(f)|^2\, df \tag{7.75}$$

In particular, the *variance per dimension* of the noise is given by

$$\text{PAM} \qquad \sigma_I^2 = E[w_{R,k}^2] = \sigma_{w_R}^2 \tag{7.76}$$

$$\text{QAM} \qquad \sigma_I^2 = E[(Re[w_{R,k}])^2] = E[(Im[w_{R,k}])^2] = \tfrac{1}{2}\sigma_{w_R}^2 \tag{7.77}$$

In the case of PAM (QAM) transmission over a channel with white noise, where $\mathcal{P}_{w_C}(f) = N_0/2$ (N_0), (7.75) yields a variance per dimension equal to

$$\sigma_I^2 = \frac{N_0}{2} E_{g_{Rc}} \tag{7.78}$$

[8] See Observation 1.6 on page 62.

where $E_{g_{Rc}}$ is the energy of the receive filter. We observe that (7.78) holds for PAM as well as for QAM.

Nyquist pulses

The problem we wish to address consists in finding the conditions on the various filters of the system, so that, *in the absence of noise*, y_k is a replica of a_k. The solution is the *Nyquist criterion* for the absence of distortion in digital transmission.

From (7.68), to obtain $y_k = a_k$ it must be:

Nyquist criterion in the time domain

$$\begin{cases} h_0 = 1 \\ h_i = 0 \qquad i \neq 0 \end{cases} \tag{7.79}$$

and ISI vanishes. A pulse $h(t)$ that satisfies the conditions (7.79) is said to be a *Nyquist pulse with modulation interval* T.

The conditions (7.79) have their equivalent in the frequency domain. They can be derived using the Fourier transform of the sequence $\{h_i\}$ (1.90),

$$\sum_{i=-\infty}^{+\infty} h_i e^{-j2\pi f i T} = \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \mathcal{H}\left(f - \frac{\ell}{T}\right) \tag{7.80}$$

where $\mathcal{H}(f)$ is the Fourier transform of $h(t)$. From the conditions (7.79) the left-hand side of (7.80) is equal to 1, hence the condition for the absence of ISI is formulated in the frequency domain for the generic pulse h as:

Nyquist criterion in the frequency domain

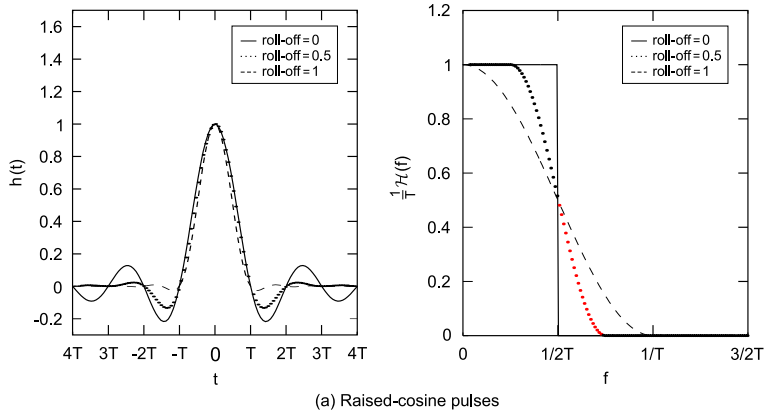
$$\sum_{\ell=-\infty}^{+\infty} \mathcal{H}\left(f - \frac{\ell}{T}\right) = T \tag{7.81}$$

From (7.81) we deduce an important fact: the Nyquist pulse with minimum bandwidth is given by:

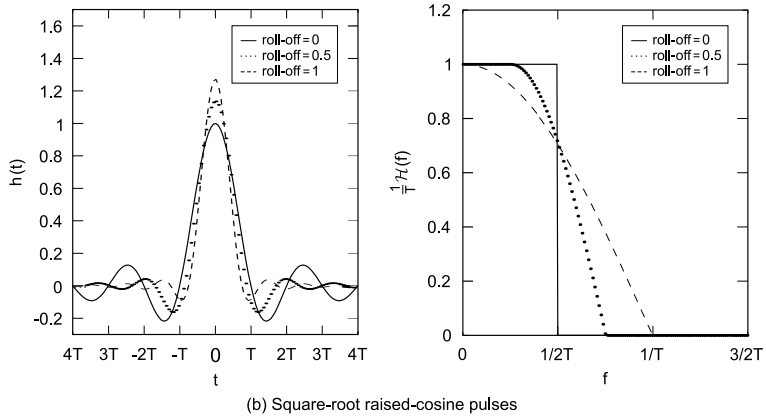
$$h(t) = h_0 \operatorname{sinc} \frac{t}{T} \xleftrightarrow{\mathcal{F}} \mathcal{H}(f) = T h_0 \operatorname{rect} \frac{f}{1/T} \tag{7.82}$$

Definition 7.1
The frequency $1/(2T)$, which coincides with half of the modulation frequency, is called *Nyquist frequency*.

A family of Nyquist pulses widely used in telecommunications is composed of the *raised cosine* pulses whose time and frequency plots, for three values of the parameter ρ, are illustrated in Figure 7.19a.



(a) Raised-cosine pulses



(b) Square-root raised-cosine pulses

Figure 7.19. Time and frequency plots of *raised cosine* and *square root raised cosine* pulses for three values of the roll-off factor ρ.

We define

$$\mathrm{rcos}(x,\rho) = \begin{cases} 1 & 0 \le |x| \le \dfrac{1-\rho}{2} \\ \cos^2\left(\dfrac{\pi}{2}\dfrac{|x| - \dfrac{1-\rho}{2}}{\rho}\right) & \dfrac{1-\rho}{2} < |x| \le \dfrac{1+\rho}{2} \\ 0 & |x| > \dfrac{1+\rho}{2} \end{cases} \tag{7.83}$$

then

$$\mathcal{H}(f) = T\,\mathrm{rcos}\left(\frac{f}{1/T}, \rho\right) \tag{7.84}$$

with inverse Fourier transform

$$\begin{aligned} h(t) &= \operatorname{sinc}\left(\frac{t}{T}\right)\frac{\pi}{4}\left[\operatorname{sinc}\left(\rho\frac{t}{T}+\frac{1}{2}\right)+\operatorname{sinc}\left(\rho\frac{t}{T}-\frac{1}{2}\right)\right] \\ &= \operatorname{sinc}\left(\frac{t}{T}\right)\cos\left(\pi\rho\frac{t}{T}\right)\frac{1}{1-\left(2\rho\frac{t}{T}\right)^2} \end{aligned} \tag{7.85}$$

It is easily proven that, from (7.83), the area of $\mathcal{H}(f)$ in (7.84) is equal to one, that is

$$h(0) = \int_{-\infty}^{+\infty} T\,\operatorname{rcos}\left(\frac{f}{1/T},\rho\right)df = 1 \tag{7.86}$$

and the energy is

$$E_h = \int_{-\infty}^{+\infty} T^2\,\operatorname{rcos}^2\left(\frac{f}{1/T},\rho\right)df = T\left(1-\frac{\rho}{4}\right) = \mathcal{H}(0)h(0)\left(1-\frac{\rho}{4}\right) \tag{7.87}$$

We note that, from (7.84), the bandwidth of the baseband equivalent system is equal to $(1+\rho)/(2T)$. Consequently, for a QAM system the required bandwidth is $(1+\rho)/T$.

Later we will also refer to *square root raised cosine* pulses, with frequency response given by

$$\mathcal{H}(f) = T\sqrt{\operatorname{rcos}\left(\frac{f}{1/T},\rho\right)} \tag{7.88}$$

and inverse Fourier transform

$$\begin{aligned} h(t) &= (1-\rho)\operatorname{sinc}\left[(1-\rho)\frac{t}{T}\right]+\rho\cos\left[\pi\left(\frac{t}{T}+\frac{1}{4}\right)\right]\operatorname{sinc}\left(\rho\frac{t}{T}+\frac{1}{4}\right) \\ &\quad+\rho\cos\left[\pi\left(\frac{t}{T}-\frac{1}{4}\right)\right]\operatorname{sinc}\left(\rho\frac{t}{T}-\frac{1}{4}\right) \\ &= \frac{\sin\left[\pi\,(1-\rho)\,\frac{t}{T}\right]+4\rho\frac{t}{T}\cos\left[\pi(1+\rho)\frac{t}{T}\right]}{\pi\left[1-\left(4\rho\frac{t}{T}\right)^2\right]\frac{t}{T}} \end{aligned} \tag{7.89}$$

In this case

$$h(0) = \int_{-\infty}^{+\infty} T\sqrt{\operatorname{rcos}\left(\frac{f}{1/T},\rho\right)}\,df = 1-\rho\left(1-\frac{4}{\pi}\right) \tag{7.90}$$

and the pulse energy is given by

$$E_h = \int_{-\infty}^{+\infty} T^2\,\operatorname{rcos}\left(\frac{f}{1/T},\rho\right)df = T \tag{7.91}$$

We note that $\mathcal{H}(f)$ in (7.88) is not the frequency response of a Nyquist pulse. Plots of $h(t)$ and $\mathcal{H}(f)$, given respectively by (7.89) and (7.88), for various values of ρ are shown in Figure 7.19b.

The parameter ρ, called *excess bandwidth parameter* or *roll-off* factor, is in the range between 0 and 1. We note that ρ determines how fast the pulse decays in time.

Observation 7.1
From the Nyquist conditions we deduce that:

1. a data sequence can be transmitted with modulation rate $1/T$ without errors if $\mathcal{H}(f)$ satisfies the Nyquist criterion and there is no noise;
2. the channel, with frequency response $\mathcal{G}_C$, must have a bandwidth equal to at least $1/(2T)$, otherwise intersymbol interference cannot be avoided.

Eye diagram

From (7.68) we observe that if the samples $\{h_i\}$, for $i \neq 0$, are not sufficiently small with respect to h_0, the ISI may result a dominant disturbance with respect to noise and impair the performance of the system. On the other hand, from (7.66) and (7.67) the discrete-time impulse response $\{h_i\}$ depends on the choice of the *timing phase* t_0 (see Chapter 14) and on the pulse shape q_R.

In the absence of noise, at the decision point the sample y_0, as a function of t_0, is given by

$$\begin{aligned} y_0 = y(t_0) &= \sum_{i=-\infty}^{+\infty} a_i q_R(t_0 - iT) \\ &= a_0 q_R(t_0) + \mathtt{i}_0(t_0) \end{aligned} \tag{7.92}$$

where

$$\begin{aligned} \mathtt{i}_0(t_0) &= \sum_{i=-\infty,\, i\neq 0}^{+\infty} a_i q_R(t_0 - iT) \\ &= \cdots + a_{-1} q_R(t_0 + T) + a_1 q_R(t_0 - T) + a_2 q_R(t_0 - 2T) + \cdots \end{aligned} \tag{7.93}$$

is the ISI.

We now illustrate, through an example, a graphic method to represent the effect of the choice of t_0 for a given pulse q_R. We consider a PAM transmission system where $y(t_0)$ is real: for a QAM system, both $Re[y(t_0)]$ and $Im[y(t_0)]$ need to be represented. We consider quaternary transmission with

$$a_k = \alpha_n \in \mathcal{A} = \{-3, -1, 1, 3\} \tag{7.94}$$

and pulse q_R as shown in Figure 7.20.

In the absence of ISI, $\mathtt{i}_0(t_0) = 0$ and $y_0 = a_0 q_R(t_0)$. In relation to each possible value α_n of a_0, the pattern of y_0 as a function of t_0 is shown in Figure 7.21: it is seen that the possible values of y_0, for $\alpha_n \in \mathcal{A}$, are further apart, therefore they offer a greater

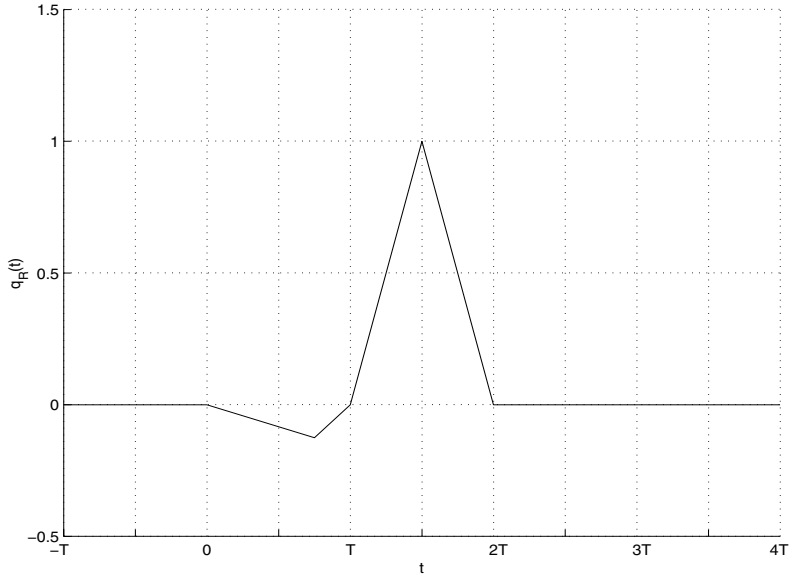


Figure 7.20. Pulse shape for the computation of the eye diagram.

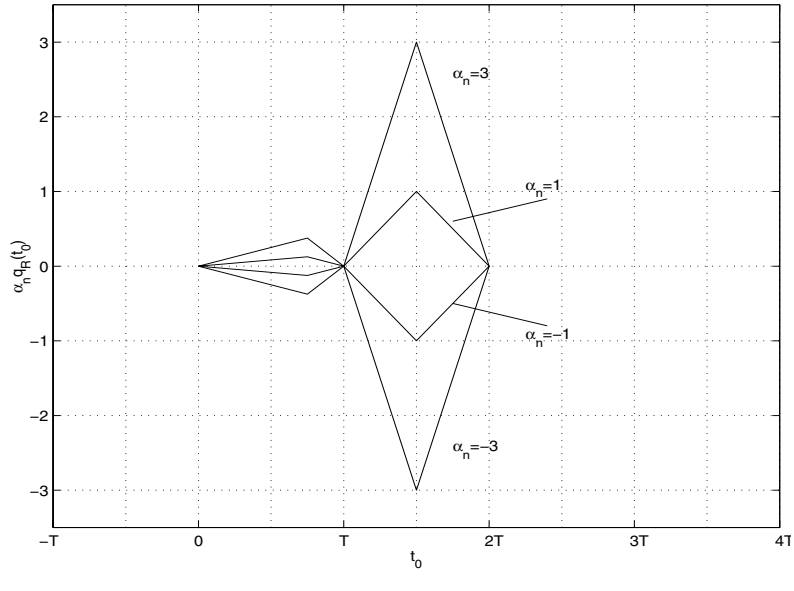


Figure 7.21. Desired component $\alpha_n q_R(t_0)$ as a function of t_0, $\alpha_n \in \{-3, -1, 1, 3\}$.

margin against noise in relation to the peak of q_R, which in this example occurs at instant $t_0 = 1.5T$. In fact, for a given t_0 and for a given message $\ldots, a_{-1}, a_1, a_2, \ldots$, it may result in $\mathtt{i}_0(t_0) \neq 0$, and this value is added to the desired sample $a_0 q_R(t_0)$.

The range of variations of $y_0(t_0)$ around the desired sample $\alpha_n q_R(t_0)$ is determined by the values

$$\mathtt{i}_{max}(t_0;\alpha_n) = \max_{\{a_k\},\, a_0=\alpha_n} \mathtt{i}_0(t_0) \tag{7.95}$$

$$\mathtt{i}_{min}(t_0;\alpha_n) = \min_{\{a_k\},\, a_0=\alpha_n} \mathtt{i}_0(t_0) \tag{7.96}$$

The *eye diagram* is characterized by the $2M$ profiles

$$\alpha_n q_R(t_0) + \begin{cases} \mathtt{i}_{max}(t_0;\alpha_n) \\ \mathtt{i}_{min}(t_0;\alpha_n) \end{cases} \qquad \alpha_n \in \mathcal{A} \tag{7.97}$$

If the symbols $\{a_k\}$ are statistically independent with balanced values, that is both α_n and $-\alpha_n$ belong to $\mathcal{A}$, defining

$$\alpha_{max} = \max_n \alpha_n \tag{7.98}$$

and

$$\mathtt{i}_{abs}(t_0) = \alpha_{max} \sum_{i=-\infty,\, i\neq 0}^{+\infty} |q_R(t_0 - iT)| \tag{7.99}$$

we have that

$$\mathtt{i}_{max}(t_0) = \mathtt{i}_{abs}(t_0) \tag{7.100}$$

$$\mathtt{i}_{min}(t_0) = -\mathtt{i}_{abs}(t_0) \tag{7.101}$$

We note that both functions do not depend on $a_0 = \alpha_n$.

For the considered pulse, the eye diagram is given in Figure 7.22. We observe that as a result of the presence of ISI, the values of y_0 may be very close to each other, and therefore reduce considerably the margin against noise. We also note that, in general, the timing phase that offers the largest margin against noise is not necessarily found in relation to the peak of q_R. In this example, however, the choice $t_0 = 1.5T$ guarantees the largest margin against noise.

In the general case, where there exists correlation between the symbols of the sequence $\{a_k\}$, it is easy to show that $\mathtt{i}_{max}(t_0;\alpha_n) \leq \mathtt{i}_{abs}(t_0)$ and $\mathtt{i}_{min}(t_0;\alpha_n) \geq -\mathtt{i}_{abs}(t_0)$. Consequently the eye may be wider as compared to the case of i.i.d. symbols.

For quaternary transmission, we show in Figure 7.23 the eye diagram obtained with a raised cosine pulse q_R, for two values of the roll-off factor.

In general, the $M - 1$ "pupils" of the eye diagram have a shape as illustrated in Figure 7.24, where two parameters are identified: the height a and the width b. The height a is an indicator of the *noise immunity* of the system. The width b indicates the *immunity* with respect to *deviations from the optimum timing phase*. For example a raised cosine pulse with $\rho = 1$ offers greater immunity against errors in the choice of t_0 as compared to the case $\rho = 0.125$. The price we pay is a larger bandwidth of the transmission channel.

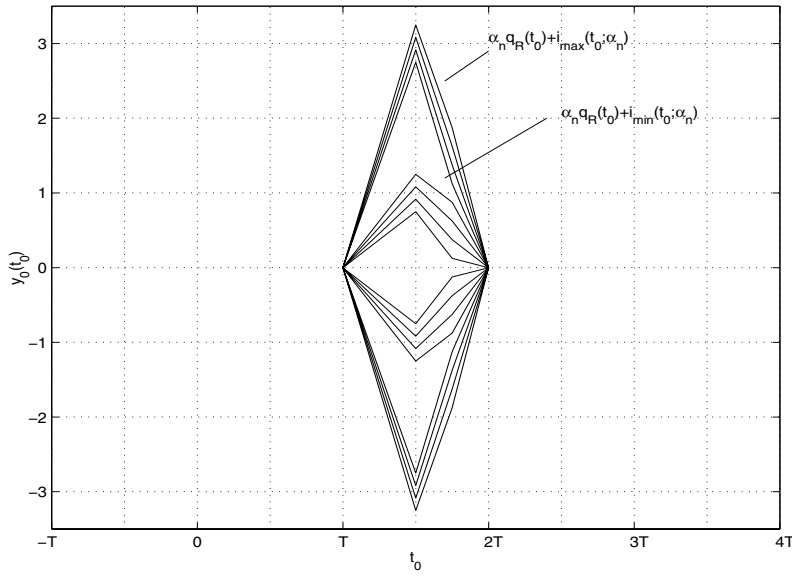


Figure 7.22. Eye diagram for quaternary transmission and pulse q_R of Figure 7.20.

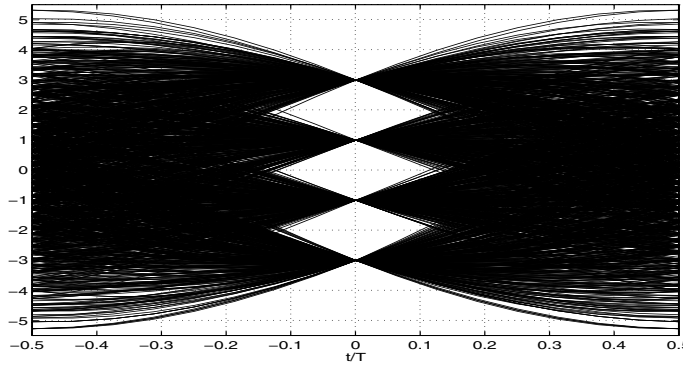
We now illustrate an alternative method to obtain the eye diagram. A long random sequence of symbols $\{a_k\}$ is transmitted over the channel, and the portions of the curve $y(t) = s_R(t)$ relative to the various intervals $[t_1, t_1 + T), [t_1 + T, t_1 + 2T), [t_1 + 2T, t_1 + 3T), \ldots]$, are mapped on the same interval, for example, on $[t_1, t_1 + T)$. Typically, we select t_1 so that the center of the eye falls in the center of the interval $[t_1, t_1 + T)$. Then the contours of the obtained eye diagram correspond to the different profiles (7.97). If the contours of the eye do not appear, it means that for all values of t_0 the worst case ISI is larger than the desired component and the *eye is shut*. We note that, if the pulse $q_R(t)$, $t \in \Re$, has a duration equal to t_h, and define $N_h = \lceil t_h/T \rceil$, we must omit plotting the values of $y(t)$ for the first and last $N_h - 1$ modulation intervals, as they would be affected by the transient behavior of the system. Moreover, for transmission with i.i.d. symbols, at every instant $t \in \Re$ the number of symbols $\{a_k\}$ that contribute to $y(t)$ is at most equal to N_h. To plot the eye diagram we need in principle to generate all the M-ary symbol sequences of length N_h: in this manner we will reproduce the values of $y(t)$ in correspondence of the various profiles.

7.3.4 Performance analysis

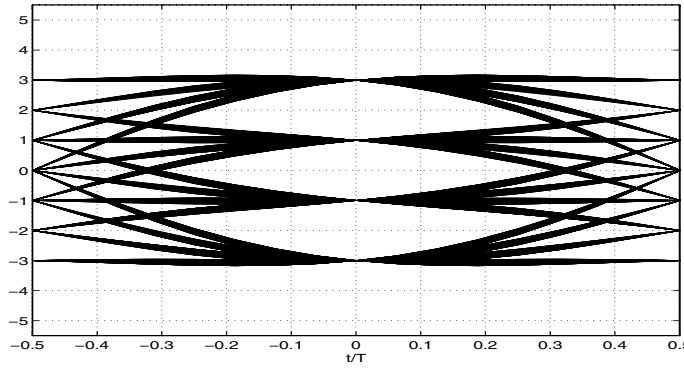
Symbol error probability in the absence of ISI

If the Nyquist conditions (7.79) are verified, from (7.65) the samples of the received signal at the decision point are given by

$$y_k = a_k + w_{R,k} \qquad a_k \in \mathcal{A} \tag{7.102}$$



(a)



(b)

Figure 7.23. Eye diagram for quaternary transmission and raised cosine pulse q_R with roll-off factor: (a) $\rho = 0.125$ and (b) $\rho = 1$.

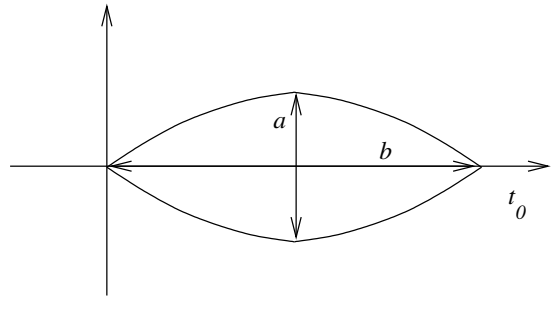


Figure 7.24. Height *a* and width *b* of the ''pupil'' of an eye diagram.

For a *memoryless decision rule* on y_k, i.e. regarding y_k as an isolated sample, and still considering the ML detection criterion described in Section 6.1, we have the following correspondences:

$$r_{R,k} = y_k = [y_{k,I}, y_{k,Q}] \quad \longleftrightarrow \quad \mathbf{r} = [r_1, r_2]^T \tag{7.103}$$

$$a_k = [a_{k,I}, a_{k,Q}] \quad \longleftrightarrow \quad \mathbf{s} = [s_1, s_2]^T \tag{7.104}$$

$$w_{R,k} = [Re[w_{R,k}], Im[w_{R,k}]] \longleftrightarrow \mathbf{w} = [w_1, w_2]^T \tag{7.105}$$

If $w_{R,k}$ has a circularly symmetric Gaussian probability density function, as is the case if equation (7.43) holds, and the values assumed by a_k are equally likely, then, given the observation y_k, the detection criterion leads to choosing the value of $\alpha_n \in \mathcal{A}$ that is closest to y_k.[9] Moreover, the error probability depends on the distance d_m between adjacent signals, in this specific case between adjacent values $\alpha_n \in \mathcal{A}$, and on the *variance per dimension* of the noise $w_{R,k}$, σ_I^2.

Hence, defining

$$\gamma = \left(\frac{d_m}{2\sigma_I}\right)^2 \tag{7.106}$$

and using (6.122) and (6.196), we have

$$M\text{-PAM} \qquad P_e = 2\left(1 - \frac{1}{M}\right) Q(\sqrt{\gamma}) \tag{7.107}$$

$$M\text{-QAM} \qquad P_e \simeq 4\left(1 - \frac{1}{\sqrt{M}}\right) Q(\sqrt{\gamma}) \tag{7.108}$$

We note that, for the purpose of computing P_e, only the variance of the noise $w_{R,k}$ is needed and not its PSD. We also note that (7.106) coincides with (6.57).

With reference to Table 7.1 and to Figure 7.6, we consider $d_m = 2h_0 = 2$. Now, for a channel with white noise, σ_I^2 is given by (7.78), hence from (7.106) it follows that

$$\gamma = \frac{2}{N_0 E_{g_{Rc}}} \tag{7.109}$$

Apparently, the above equation could lead to choosing a filter g_{Rc} with very low energy, so that $\gamma \gg 1$. However, here g_{Rc} is not arbitrary, but it must be chosen such that the condition (7.79) for the absence of ISI is satisfied. We will see in Chapter 8 a criterion to design the filter g_{Rc}.

The general case of computation of P_e in the presence of ISI and non-Gaussian noise is given in Appendix 7.B.

Matched filter receiver

Assuming absence of ISI, (7.107) and (7.108) imply that the best performance, that is the minimum value of P_e, is obtained when the ratio γ is maximum.

[9] We observe that this memoryless decision criterion is optimum only if the noise samples $\{w_{R,k}\}$ are statistically independent.

Assuming that the pulse q_C that determines the signal at the channel output is *given*, the solution (see Section 1.10 on page 73) is provided by the receive filter g_{Rc} matched to q_C: hence the name matched filter (MF). In particular, with reference to the scheme of Figure 7.12 and for white noise w_C, we have

$$\mathcal{G}_{Rc}(f) = K\, Q_C^*(f)\, e^{-j2\pi f t_0} \tag{7.110}$$

where K is a constant. In this case, from the condition

$$h_0 = \mathcal{F}^{-1}[\mathcal{G}_{Rc}(f) Q_C(f)]\,|_{t=t_0} = K\, \mathrm{r}_{q_C}(0) = 1 \tag{7.111}$$

we obtain

$$K = \frac{1}{E_{q_C}} \tag{7.112}$$

Substitution of (7.112) in (7.110) yields $E_{g_{Rc}} = 1/E_{q_C}$. Therefore (7.109) assumes the form

$$\gamma = \gamma_{MF} = \frac{2E_{q_C}}{N_0} \tag{7.113}$$

The matched filter receiver is of interest also for another reason. Using (7.56) it is possible to determine the relation between the signal-to-noise ratios γ at the decision point and Γ at the receiver input: for a QAM system it turns out

$$\gamma_{MF} = \frac{\Gamma}{\frac{1}{2}\mathrm{M}_a} \tag{7.114}$$

where $\mathrm{M}_a/2$ is the statistical power per dimension of the symbol sequence.

We stress the point that, for a certain modulation system with a given pulse q_C and a given Γ, it is not possible by varying the filter g_{Rc} to obtain a higher γ at the decision point than (7.114), and consequently a better P_e. The equation (7.114) is often used as an upper bound of the system performance. However, we note that we have ignored the possible presence of ISI at the decision point that the choice of (7.110) might imply. We further observe that the matched filter receiver corresponds to the optimum receiver developed in Chapter 6. The only difference is that in that case the energy of the matched filter is equal to one.

In Appendix 7.E we describe a Monte Carlo method for simulations of a discrete-time QAM system.

7.4 Carrierless AM/PM (CAP) modulation

The carrierless AM/PM (CAP) modulation is a passband modulation technique that is closely related to QAM (see Section 7.2). The scheme of a QAM system is repeated for convenience in Figure 7.25. In CAP systems, the carrier is omitted by using passband filters and exploiting the periodicity of the PSD of the sequence $\{a_k\}$.

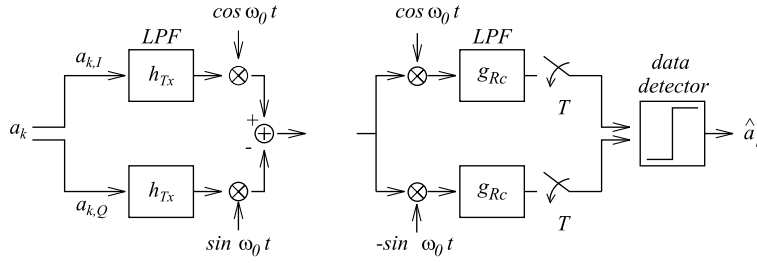


Figure 7.25. QAM implementation using baseband filters.

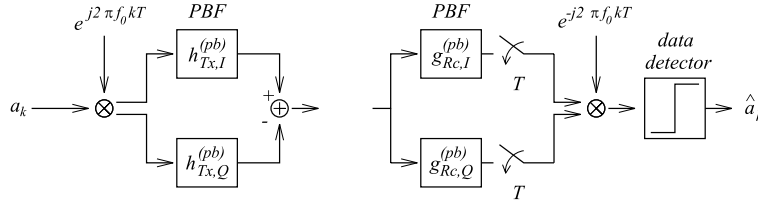


Figure 7.26. QAM implementation using passband filters.

Using passband filters, the QAM scheme of Figure 7.25 is modified into the scheme of Figure 7.26, where the impulse responses of the transmit filters are given by

$$h_{Tx,I}^{(pb)}(t) = h_{Tx}(t)\cos(2\pi f_0 t) \tag{7.115}$$

$$h_{Tx,Q}^{(pb)}(t) = h_{Tx}(t)\sin(2\pi f_0 t) \tag{7.116}$$

and the impulse responses of the receive filters are given by

$$g_{Rc,I}^{(pb)}(t) = g_{Rc}(t)\cos(2\pi f_0 t) \tag{7.117}$$

$$g_{Rc,Q}^{(pb)}(t) = g_{Rc}(t)\sin(2\pi f_0 t) \tag{7.118}$$

Applying Theorem 1.1 we observe that, if f_0 is larger than the bandwidth of h_{Tx}, the hypothesis always verified in practice, then the pulses $h_{Tx,I}^{(pb)}$ and $h_{Tx,Q}^{(pb)}$ are related by the Hilbert transform (1.163); the same relation exists between the pulses $g_{Rc,I}^{(pb)}$ and $g_{Rc,Q}^{(pb)}$. Consequently the two pulses (7.115) and (7.116) are orthogonal. Note that this property holds through the transmission channel.

From (7.29), where for simplicity we set $\varphi_0 = 0$, in a QAM system the transmitted signal can be expressed as

$$\begin{aligned} s_{QAM}(t) &= Re\left[\sum_{k=-\infty}^{\infty} a_k\, h_{Tx}(t-kT)\, e^{j2\pi f_0 t}\right] \\ &= \sum_{k=-\infty}^{\infty} \tilde{a}_{k,I}\, h_{Tx,I}^{(pb)}(t-kT) - \tilde{a}_{k,Q}\, h_{Tx,Q}^{(pb)}(t-kT) \end{aligned} \tag{7.119}$$

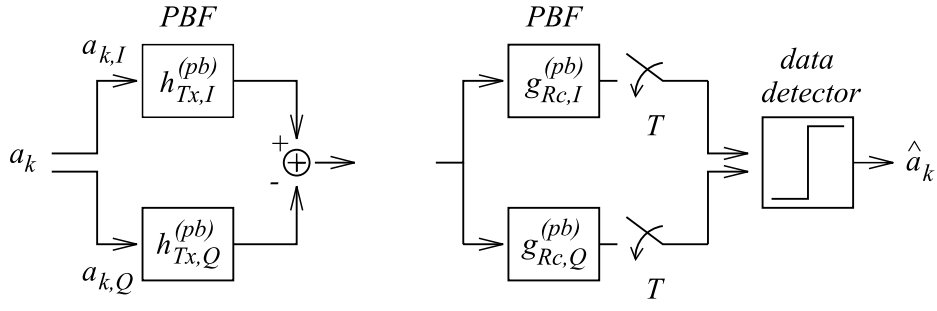


Figure 7.27. Modulator and demodulator for a CAP system.

where $\tilde{a}_{k,I} = Re[a_k e^{j2\pi f_0 kT}]$ and $\tilde{a}_{k,Q} = Im[a_k e^{j2\pi f_0 kT}]$. Therefore in the scheme of Figure 7.26, the input to the modulation filters at instant k is given by $\tilde{a}_k = a_k e^{j2\pi f_0 kT}$, which is equal to the symbol a_k with an additional deterministic phase that must be removed at the receiver.

CAP modulation is obtained by leaving out the additional phase, as shown in Figure 7.27. If we define

$$\tilde{h}_{Tx}(t) = h_{Tx}(t)e^{j2\pi f_0 t} = h^{(pb)}_{Tx,I}(t) + jh^{(pb)}_{Tx,Q}(t) \tag{7.120}$$

the transmitted signal is then given by

$$\begin{aligned} s_{CAP}(t) &= Re\left[\sum_{k=-\infty}^{\infty} a_k \tilde{h}_{Tx}(t-kT)\right] \\ &= \sum_{k=-\infty}^{\infty} a_{k,I} h^{(pb)}_{Tx,I}(t-kT) - a_{k,Q} h^{(pb)}_{Tx,Q}(t-kT) \end{aligned} \tag{7.121}$$

Because the pulses $h^{(pb)}_{Tx,I}(t)$ and $h^{(pb)}_{Tx,Q}(t)$, filtered by the transmission channel, are still related by the Hilbert transform, the receiver uses a passband matched filter of the *phase-splitter* type, implemented by two real-valued filters with impulse responses given by (7.117) and (7.118) (see Figure 7.27). In general, if the transfer function of the transmission medium is unknown *a priori*, the receive filters are adaptive (see Chapter 8).

We note that CAP modulation is equivalent to QAM, with the difference that in a QAM system the input symbols $\{a_k\}$ are substituted by the rotated symbols $\{\tilde{a}_k\}$. If f_0 is an integer multiple of $1/T$, then there is no difference between CAP and QAM. QAM is usually selected if $f_0 \gg 1/(2T)$. In the case where f_0 is not much larger than $1/(2T)$, as usually occurs in data transmission systems over metallic cables, CAP modulation may be preferred to QAM because it does not need carrier recovery. On the other hand, for transmission channels that introduce frequency offset, an acquisition mechanism of the carrier must be introduced and typically QAM is adopted.

7.5 Regenerative PCM repeaters

This section is divided into two parts: the first considers a PCM encoded signal (see Chapter 5) and evaluates the effects of digital channel errors on the reproduced analog signal, the second compares the performance of an analog transmission system with that of a digital system for the transmission of analog signals represented by the PCM method.

7.5.1 PCM signals over a binary channel

With reference to Figure 7.1 for PAM and to Figure 7.5 for QAM, the transformation that maps input bits $\{b_\ell\}$ in output bits $\{\hat{b}_\ell\}$ is called *binary channel* and is represented in Figure 7.28 (see also Figure 6.21 on page 457). A binary channel is typically characterized by the *bit rate* $R_b = 1/T_b$ (bit/s) and by the bit error probability.

The simplest model considers errors to be i.i.d., therefore the various distributions are obtainable starting from:

$$P_{bit} = P[\hat{b}_\ell \neq b_\ell] \tag{7.122}$$

Correspondingly we give in Figure 7.29 the statistical model associated with a *memoryless binary symmetric channel*. In the following it is useful to evaluate the error probability of words $\mathbf{c}$ composed of $\mathtt{b}$ bits:

$$P_{e,\mathbf{c}} = 1 - (1 - P_{bit})^{\mathtt{b}} \tag{7.123}$$

assuming errors are i.i.d.. On the other hand, if $P_{bit} \ll 1$ it follows that $(1 - P_{bit})^{\mathtt{b}} \simeq 1 - \mathtt{b}P_{bit}$ and

$$P_{e,\mathbf{c}} \simeq \mathtt{b}P_{bit} \tag{7.124}$$

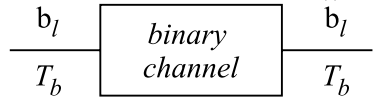


Figure 7.28. Binary channel associated with digital transmission.

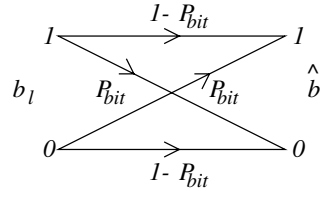


Figure 7.29. Memoryless binary symmetric channel.

Linear PCM coding of waveforms

As seen in Chapter 5, PCM is essentially performed by an analog-to-digital converter, which represents the information contained in the instantaneous samples of an analog signal by words of $\mathtt{b}$ bits. Figure 7.30 gives the composite scheme where an input analog signal is converted by an ADC into a binary sequence $\{b_\ell\}$, which in turn is transmitted over a binary channel. The signal $\tilde{s}(t)$ is then reconstructed from the received bits by a DAC.

The operations that transform $s(t)$ into $\{b_\ell\}$ are summarized as 1) sampling; 2) quantization; and 3) coding. We assume that each word is composed of $\mathtt{b}$ bits: hence there are $L = 2^{\mathtt{b}}$ possible words corresponding to L quantizer levels. The quantizer is assumed uniform in the range $[-\tau_{sat}, \tau_{sat}]$, with a quantization step-size Δ given by (5.25).

The inverse bit mapper of the ADC performs the following function:

$$s_q(kT_c) = Q_i \longrightarrow c(k) = [c_{\mathtt{b}-1}(k), \ldots, c_0(k)] \tag{7.125}$$

where, to simplify the analysis, we assume the rule

$$\begin{cases} i = \displaystyle\sum_{j=0}^{\mathtt{b}-1} c_j(k)2^j \\ Q_i = -\tau_{sat} + (i + \frac{1}{2})\Delta \end{cases} \tag{7.126}$$

In (7.125) $c(k)$ is the word of $\mathtt{b}$ bits transmitted over the binary channel, with components $c_j(k) \in \{0, 1\}$, $j = 0, 1, \ldots, \mathtt{b} - 1$.

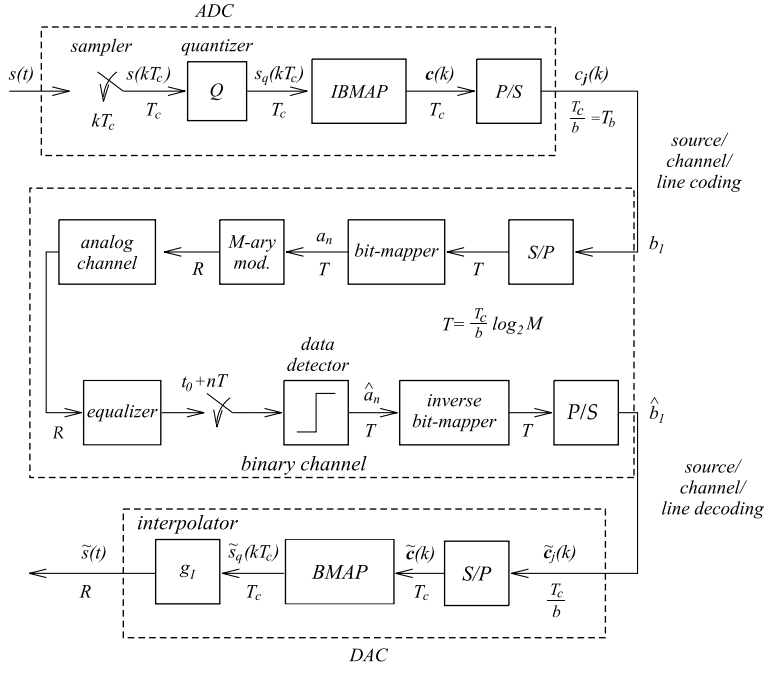


Figure 7.30. Composite transmission scheme of an analog signal via a binary channel.

We assume the binary channel symmetric and memoryless. Hence, if we express the generic detected bit at the output of the binary channel as $\tilde{c}_j(k) \in \{0, 1\}$, the error probability is given by:

$$P[\tilde{c}_j(k) \neq c_j(k)] = P_{bit} \tag{7.127}$$

as illustrated in Figure 7.29.

If we denote with $\tilde{c}(k) = [\tilde{c}_{\mathtt{b}-1}(k), \ldots, \tilde{c}_0(k)]$ the received word, the bit mapper of the DAC performs the inverse operation of (7.125):

$$\tilde{c}(k) \longrightarrow \tilde{s}_q(kT_c) = Q_{\tilde{\imath}} \tag{7.128}$$

where

$$\begin{cases} \tilde{\imath} = \displaystyle\sum_{j=0}^{\mathtt{b}-1} \tilde{c}_j(k) 2^j \\ Q_{\tilde{\imath}} = -\tau_{sat} + (\tilde{\imath} + \frac{1}{2})\Delta \end{cases} \tag{7.129}$$

Given the one-to-one map between words and quantizer levels, using (7.124) it follows that

$$P[\tilde{s}_q(kT_c) \neq s_q(kT_c)] = P_{e,\mathbf{c}} \simeq \mathtt{b} P_{bit} \tag{7.130}$$

Overall system performance

In Figure 7.30 the reconstructed analog signal $\tilde{s}$ is different from the transmitted signal s for two reasons:

1. the presence of the quantization noise in the ADC;
2. the errors on the detection of the binary sequence at the output of the binary channel.

The quantizer introduces an error e_q such that

$$s_q(kT_c) = s(kT_c) + e_q(kT_c) \tag{7.131}$$

and the binary channel reconstructs s_q with a certain error e_{Ch},

$$\tilde{s}_q(kT_c) = s_q(kT_c) + e_{Ch}(kT_c) \tag{7.132}$$

Therefore the overall relation is

$$\tilde{s}_q(kT_c) = s(kT_c) + e_q(kT_c) + e_{Ch}(kT_c) \tag{7.133}$$

where the two error terms are assumed uncorrelated, as they are to be ascribed to different phenomena. In particular, assuming the quantization noise uniform, from (5.41) it follows

$$\mathtt{M}_{e_q} = \frac{\Delta^2}{12} = \frac{\tau_{sat}^2}{3 \cdot 2^{2\mathtt{b}}} \tag{7.134}$$

The computation of $\mathtt{M}_{e_{Ch}}$ is somewhat more difficult. First, from (7.126) and (7.132) we have

$$e_{Ch}(kT_c) = \Delta \sum_{j=0}^{\mathtt{b}-1} (\tilde{c}_j(k) - c_j(k)) 2^j \tag{7.135}$$

Let the error on the j-th transmitted bit be

$$\varepsilon_j(k) = \tilde{c}_j(k) - c_j(k) \tag{7.136}$$

then (7.135) becomes

$$e_{Ch}(kT_c) = \Delta \sum_{j=0}^{\mathtt{b}-1} \varepsilon_j(k) 2^j \tag{7.137}$$

We note that $\varepsilon_j(k) \in \{-1, 0, 1\}$, with probabilities given by

$$\begin{cases} P[\varepsilon_j = 1] = \frac{1}{2} P_{bit} \\ P[\varepsilon_j = -1] = \frac{1}{2} P_{bit} \\ P[\varepsilon_j = 0] = 1 - P_{bit} \end{cases} \tag{7.138}$$

Then, observing (7.138), we get

$$E[\varepsilon_j(k)] = 0 \tag{7.139}$$

and

$$\begin{aligned} E[\varepsilon_j^2(k)] &= 1 P[\varepsilon_j(k) \neq 0] + 0 P[\varepsilon_j(k) = 0] \\ &= P[\tilde{c}_j(k) \neq c_j(k)] = P_{bit} \end{aligned} \tag{7.140}$$

For a memoryless binary channel

$$E[\varepsilon_{j_1}(k) \varepsilon_{j_2}(k)] = \begin{cases} E[\varepsilon_{j_1}^2] = P_{bit} & \text{for } j_1 = j_2 \\ 0 & \text{for } j_1 \neq j_2 \end{cases} \tag{7.141}$$

hence from (7.137)

$$E[e_{Ch}^2(kT_c)] = \Delta^2 P_{bit} \sum_{j=0}^{\mathtt{b}-1} 2^{2j} = \Delta^2 P_{bit} \frac{2^{2\mathtt{b}} - 1}{3} \tag{7.142}$$

We note that, recalling footnote 3 on page 338, the statistical power of the output signal of an interpolator filter in a DAC is equal to the statistical power of the input samples. Consequently, from (7.133) the output signal-to-noise ratio is given by

$$\begin{aligned} \Lambda_{PCM} &= \frac{E[s^2(t)]}{E[|\tilde{s}(t) - s(t)|^2]} \\ &= \frac{E[s^2(kT_c)]}{E[|\tilde{s}_q(kT_c) - s(kT_c)|^2]} \\ &= \frac{\mathtt{M}_s}{\mathtt{M}_{e_q} + \mathtt{M}_{e_{Ch}}} \end{aligned} \tag{7.143}$$

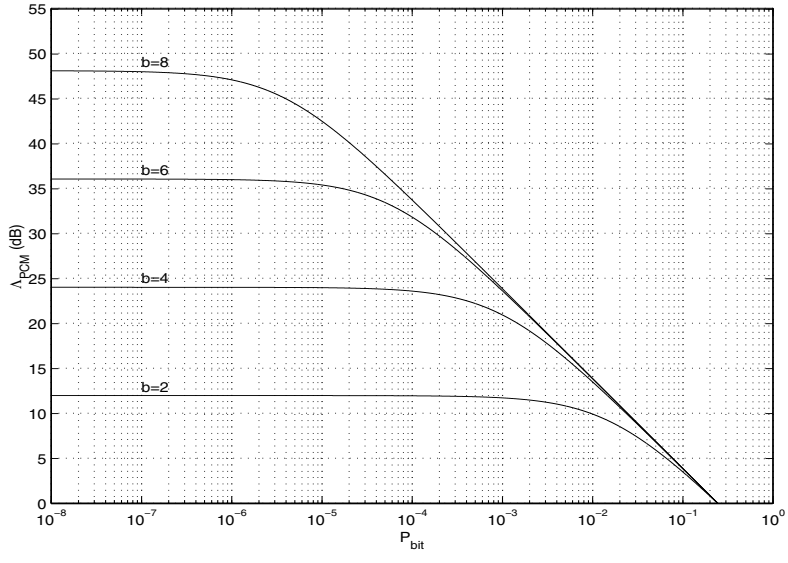


Figure 7.31. Signal-to-noise ratio of a PCM system as a function of P_{bit}.

Using (7.134) and (7.142), and for a signal-to-quantization noise ratio $\Lambda_q = \mathtt{M}_s/(\Delta^2/12)$ (see (5.33)), we get

$$\Lambda_{PCM} = \frac{\Lambda_q}{1 + 4P_{bit}(2^{2\mathtt{b}} - 1)} \tag{7.144}$$

We note that usually P_{bit} is such that $P_{bit}2^{2\mathtt{b}} \ll 1$: thus it results $\Lambda_{PCM} \simeq \Lambda_q$, that is the output error is mainly due to the quantization error.

In particular, for a signal $s \in \mathcal{U}(-\tau_{sat}, \tau_{sat}]$ whereby $\Lambda_q = 2^{2\mathtt{b}}$, equation (7.144) is represented in Figure 7.31 for various values of $\mathtt{b}$. For $P_{bit} < 1/(4 \cdot 2^{2\mathtt{b}})$ the output signal is corrupted mainly by the quantization noise, whereas for $P_{bit} > 1/(4 \cdot 2^{2\mathtt{b}})$ the output is affected mainly by errors introduced by the binary channel. For example for $P_{bit} = 10^{-4}$, going from $\mathtt{b} = 6$ to $\mathtt{b} = 8$ bits per sample yields an increment of Λ_{PCM} of only 2 dB.

We observe that in the general case of non-uniform quantization there are no simple expressions similar to (7.142) and (7.144); however, the above observations remain valid.

7.5.2 Regenerative repeaters

The signal sent over a transmission line is attenuated and corrupted by noise. To cover long distances it is therefore necessary to place repeaters along the transmission line to restore the signal.

Analog transmission

The only solution possible in an analog transmission system is to place *analog repeaters* consisting of amplifiers with suitable filters to restore the level of the signal and eliminate the noise outside the passband of the desired signal. The cascade of amplifiers along a transmission line, however, deteriorates the signal-to-noise ratio.

We consider the simplified scheme of Example 4.2.2 on page 271 with

- $s(t)$, transmitted signal with bandwidth B and available power $\mathtt{P}_s$;
- $s_{Ch}(t)$, desired signal at the output of transmission section i, with available power $\mathtt{P}_{s_{Ch}}$;
- $w(t)$, effective noise at the input of repeater i;
- $r(t) = s_{Ch}(t) + w(t)$, overall signal at the amplifier input of repeater i;
- $\tilde{s}(t)$, signal at the output of a system with N repeaters.

We note that, if $\mathtt{a}_c$ is the attenuation of the generic section i, then

$$\mathtt{P}_{s_{Ch}} = \frac{1}{\mathtt{a}_c}\mathtt{P}_s \tag{7.145}$$

In this example both the transmission channel and the various amplifiers do not introduce distortion; the only disturbance in $\tilde{s}(t)$ is due to additive noise introduced by the various devices.

For a source at noise temperature $\mathtt{T}_0$, if $\mathtt{F}_A$ is the noise figure of a single amplifier, the signal-to-noise ratio at the amplifier output of a single section is given by (4.92):

$$\Lambda = \frac{\mathtt{P}_{s_{Ch}}}{\mathtt{k}\mathtt{T}_0\mathtt{F}_A B} \tag{7.146}$$

Analogously for N analog repeater sections, as the overall noise figure is equal to $\mathtt{F} = N\mathtt{F}_{sr}$ (see (4.77)), the overall signal-to-noise ratio, expressed as

$$\Lambda_a = \frac{E[s^2(t)]}{E[|\tilde{s}(t) - s(t)|^2]} \tag{7.147}$$

is given by

$$\Lambda_a = \frac{\mathtt{P}_s}{\mathtt{k}\mathtt{T}_0\mathtt{F}B} = \frac{\Lambda}{N} \tag{7.148}$$

Obviously in the derivation of (7.148) it is assumed that (4.83) holds, as a statistical power ratio is equated with an effective power ratio.

Hence, in a system with analog repeaters, the noise builds up repeater after repeater and the overall signal-to-noise ratio worsens as the number of repeaters increases. Moreover, it must be remembered that in practical systems, possible distortion experienced by the desired signal through the various transmission channels and amplifiers also accumulates, contributing to an increase of the disturbance in $\tilde{s}(t)$.

Digital transmission

In a digital transmission system, as an alternative to the simple amplification of the received signal $r(t)$, we can resort to the *regeneration* of the signal. With reference to the scheme of Figure 7.32, given the signal $r(t)$, the digital message $\{\hat{b}_\ell\}$ is first reconstructed, and then re-transmitted by a modulator.

Modeling each *regenerative repeater* by a memoryless binary symmetric channel (see Definition 6.1 on page 457) with error probability P_{bit}, and ignoring the probability that a bit undergoes more errors along the various repeaters, the bit error probability at the output of N regenerative repeaters is equal to[10]

$$P_{bit,N} \simeq 1 - (1 - P_{bit})^N \simeq N P_{bit} \tag{7.149}$$

assuming $P_{bit} \ll 1$, and errors of the different repeaters statistically independent.

To obtain an expression of P_{bit}, it is necessary to specify the type of modulator. Let us consider an M-PAM system; then from (6.125) we get

$$P_{bit} = \frac{2(M-1)}{M \log_2 M} Q\left(\sqrt{\frac{3}{M^2-1}\Gamma}\right) \tag{7.150}$$

where from (6.108)[11]

$$\Gamma = \frac{\mathrm{P}_{s_{Ch}}}{\mathrm{k}\mathrm{T}_0\mathrm{F}_A B_{min}} \tag{7.151}$$

It is interesting to compare the bit error probability at the output of N repeaters in the two cases.

$$\textit{Analog repeaters}\colon\ P_{bit,N} = \frac{2(M-1)}{M \log_2 M} Q\left(\sqrt{\frac{3}{M^2-1}\frac{\Gamma}{N}}\right) \tag{7.152}$$

$$\textit{Regenerative repeaters}\colon\ P_{bit,N} = \frac{2(M-1)}{M \log_2 M} N Q\left(\sqrt{\frac{3}{M^2-1}\Gamma}\right) \tag{7.153}$$

Note that in (7.152) we used (7.148).

Even if a regenerative repeater is much more complex than an analog repeater, for a given overall P_{bit}, regeneration allows a significant saving in the power of the transmitted signal.

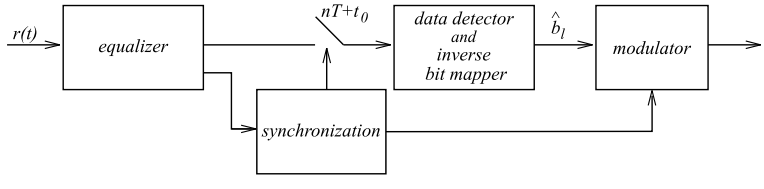


Figure 7.32. Basic scheme of digital regeneration.

[10] We note that a more accurate study shows that the errors have a Bernoulli distribution [4].

[11] To simplify the notation, we have indicated with the same symbol s_{Ch} the desired signal at the amplifier input for both analog transmission and digital transmission. Note, however, that in the first case s_{Ch} depends linearly on s, whereas in the second it represents the modulated signal that does not depend linearly on s.

Comparison between analog and digital transmission

We now compare the analog transmission of a signal $s(t)$ with the digital transmission, which includes PCM coding of $s(t)$ and modulation of the message.

For PCM coding of $s(t)$ the bit rate of the message is given by

$$R_b = \mathrm{b}\, 2\mathrm{B} \tag{7.154}$$

Consequently, for an M-PAM modulator, the modulation interval T is equal to $\log_2 M/R_b$, and the minimum bandwidth of the transmission channel is equal to

$$B_{min} = \frac{1}{2T} = \frac{\mathrm{b}}{\log_2 M} B \tag{7.155}$$

We note that the digital transmission of an analog signal may require a considerable expansion of the required bandwidth, if M is small. Obviously, using a more efficient digital representation of waveforms, for example by CELP, and/or a modulator with higher spectral efficiency, for example, by resorting to multilevel transmission, B_{min} may result very close to B or even smaller.

Using (7.155) in (7.151), from (7.146) we have

$$\Gamma = \frac{\log_2 M}{\mathrm{b}} \Lambda \tag{7.156}$$

The comparison between the two systems is based on the overall signal-to-noise ratio for the same transmitted power and transmission channel characteristics. To simplify the notation, initially we will consider a 2-PAM as modulator.

Substituting the value of Γ given by (7.156) for $M = 2$ in (7.152) and (7.153), and recalling (7.144), valid for a uniform quantizer with $\Lambda_q = 2^{2\mathrm{b}}$, that is assuming a uniform signal, see (5.44), we get

$$\Lambda_{PCM} = \begin{cases} \dfrac{2^{2\mathrm{b}}}{1 + 4(2^{2\mathrm{b}} - 1) Q\left(\sqrt{\dfrac{\Lambda}{\mathrm{b}N}}\right)} & N \text{ analog repeaters} \\ \dfrac{2^{2\mathrm{b}}}{1 + 4(2^{2\mathrm{b}} - 1) N Q\left(\sqrt{\dfrac{\Lambda}{\mathrm{b}}}\right)} & N \text{ regenerative repeaters} \end{cases} \tag{7.157}$$

Or else, using (7.148), we get

$$\Lambda_{PCM} = \begin{cases} \dfrac{2^{2\mathrm{b}}}{1 + 4(2^{2\mathrm{b}} - 1) Q\left(\sqrt{\dfrac{\Lambda_a}{\mathrm{b}}}\right)} & N \text{ analog repeaters} \\ \dfrac{2^{2\mathrm{b}}}{1 + 4(2^{2\mathrm{b}} - 1) N Q\left(\sqrt{\dfrac{\Lambda_a N}{\mathrm{b}}}\right)} & N \text{ regenerative repeaters} \end{cases} \tag{7.158}$$

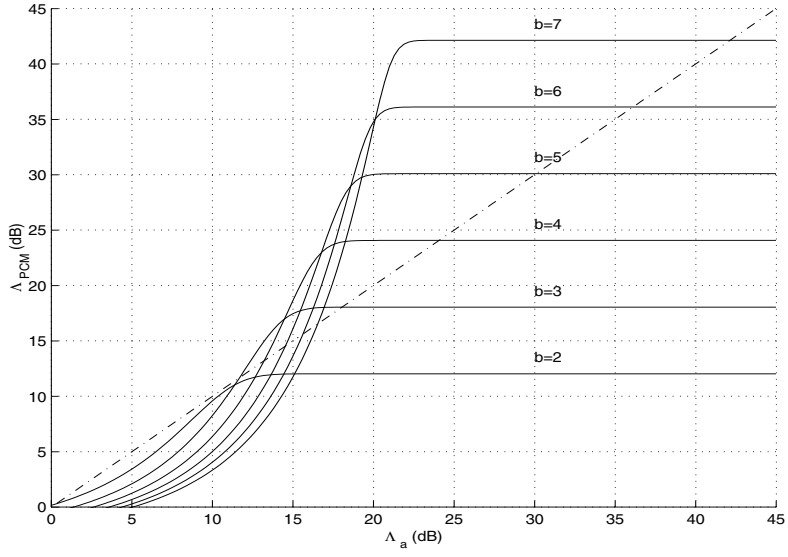


Figure 7.33. Λ_{PCM} as a function of Λ_a for analog repeaters and 2-PAM. The parameter $\mathtt{b}$ denotes the number of bits for linear PCM representation.

In the case of *analog repeaters*, the plot of Λ_{PCM} as a function of Λ_a is given in Figure 7.33. We note that Λ_{PCM} is typically higher than Λ_a, as long as a sufficiently large number of bits and Λ_a larger than 17 dB are considered. However, the PCM system is penalized by the increment of the bandwidth of the transmission channel.

Using *regenerative repeaters*, for example $N = 20$ in Figure 7.34, Λ_{PCM} is always much higher than Λ_a, assuming an adequate number of bits for PCM coding is used.

We note the threshold effect of P_{bit} as a function of Γ in a digital transmission system: if the ratio Γ is higher than a certain threshold, then P_{bit} is very small. Consequently, the quantization error becomes predominant at the receiver.

While the previous graphs relate Λ_{PCM} directly to Λ_a, in practice it is interesting to determine the minimum value of Λ (or Γ) so that Λ_{PCM} and Λ_a reach a certain value, say, of the order of 20–40 dB, depending on the applications.

We illustrate in Figure 7.35 these relations by varying the number N of repeaters and using a PCM encoder with $\mathtt{b} = 7$. We show also a comparison for the same required bandwidth, which implies a modulator with $M = 2^{\mathtt{b}}$ levels. In this case, with respect to 2-PAM, for the same P_{bit} the modulator requires an increment of about $6(\mathtt{b} - 1)$ dB in terms of Γ; therefore, from (7.156), the increment in terms of Λ is equal to $6(\mathtt{b} - 1) - 10\log_{10}(\mathtt{b}-1)$. The curve of Λ_{PCM} as a function of Λ for 128-PAM, plotted in Figure 7.35, is shifted to the right by about 28 dB with respect to 2-PAM.

Therefore also for the same bandwidth, digital transmission is more efficient than analog transmission if the number of repeaters is large.

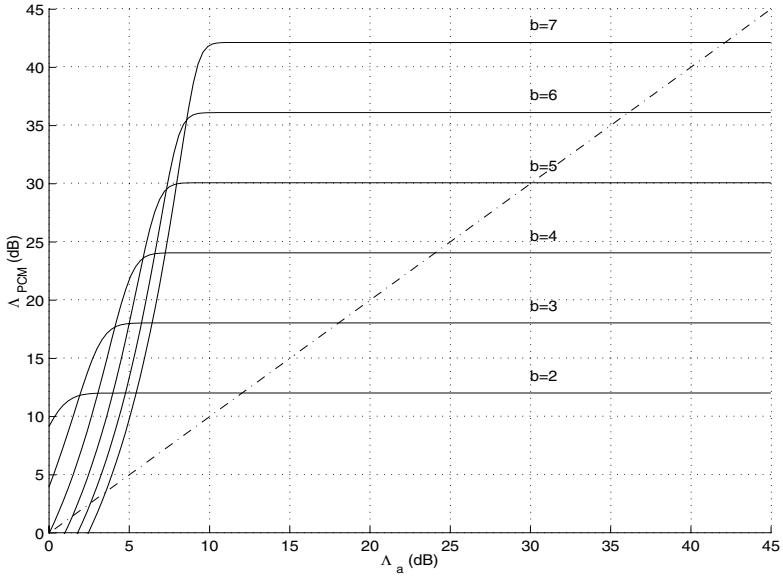


Figure 7.34. Λ_{PCM} as a function of Λ_a for 2-PAM transmission and $N = 20$ regenerative repeaters. The parameter b is the number of bits for linear PCM representation.

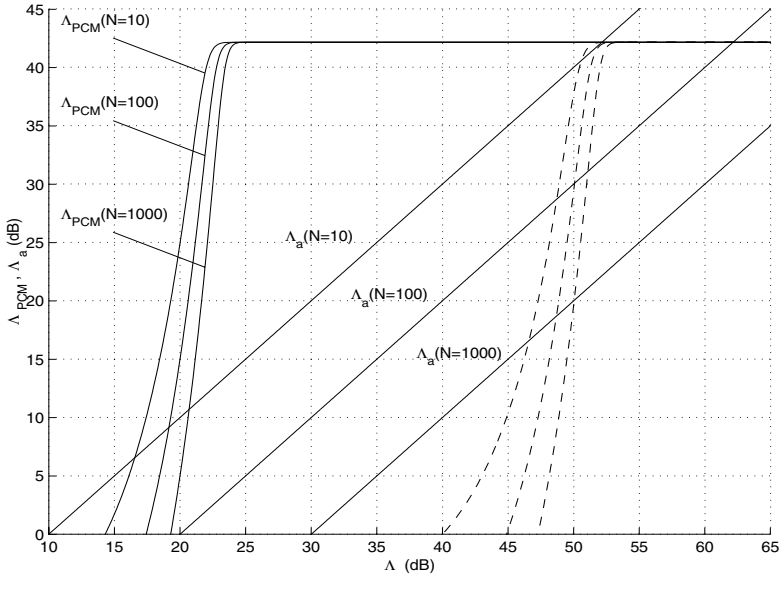


Figure 7.35. Λ_a for analog transmission obtained by varying the number N of analog repeaters, and Λ_{PCM} for digital transmission with 2-PAM and $\mathrm{b} = 7$, obtained by varying the number N of regenerative repeaters, as a function of Λ (signal-to-noise ratio of each repeater section). The dashed line represents Λ_{PCM} for 128-PAM and $\mathrm{b} = 7$.

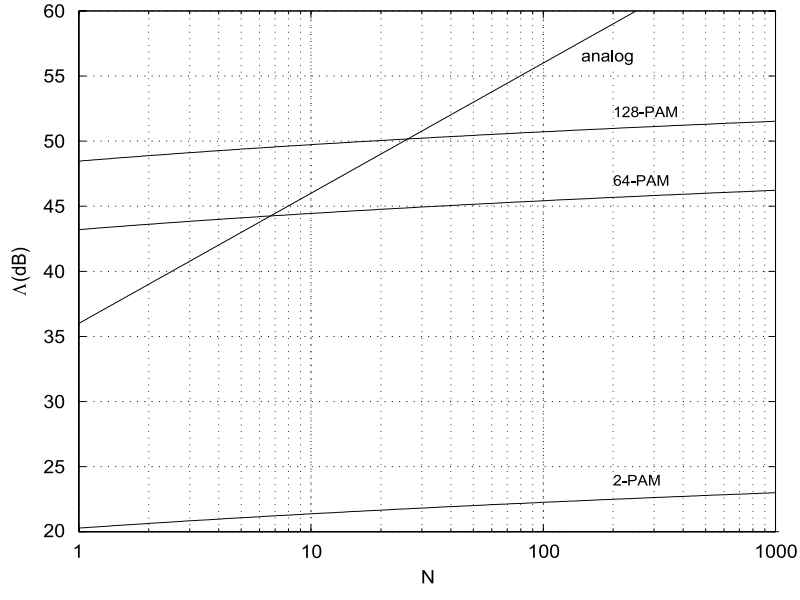


Figure 7.36. Minimum value of Λ as a function of the number N of regenerative repeaters required to guarantee an overall signal-to-noise ratio of 36 dB, for analog transmission and digital transmission with three different modulators. The number of bits for PCM coding is $\mathtt{b} = 7$.

Finally, for a given objective,

$$\Lambda_{PCM} = \Lambda_a = 36\,\text{dB} \tag{7.159}$$

we illustrate in Figure 7.36 the minimum value of Λ as a function of the number of regenerative repeaters, for three different modulators.

Bibliography

[1] L. W. Couch, *Digital and analog communication systems*. Upper Saddle River, NJ: Prentice-Hall, 1997.

[2] J. G. Proakis and M. Salehi, *Communication system engineering*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[3] M. S. Roden, *Analog and digital communication systems*. Upper Saddle River, NJ: Prentice-Hall, 1996.

[4] A. Papoulis, *Probability, random variables and stochastic processes*. New York: McGraw-Hill, 3rd ed., 1991.

[5] S. Benedetto and E. Biglieri, *Principles of digital transmission with wireless applications*. New York: Kluwer Academic Publishers, 1999.

[6] P. Kabal and P. Pasupathy, "Partial-response signaling", *IEEE Trans. on Communications*, vol. 23, pp. 921–934, Sept. 1975.

[7] D. L. Duttweiler, J. E. Mazo, and D. G. Messerschmitt, "An upper bound on the error probability in decision-feedback equalization", *IEEE Trans. on Information Theory*, vol. 20, pp. 490–497, July 1974.

[8] G. Birkoff and S. MacLane, *A survey of modern algebra*. New York: Macmillan Publishing Company, 3rd ed., 1965.

[9] D. G. Messerschmitt and E. A. Lee, *Digital communication*. Boston, MA: Kluwer Academic Publishers, 2nd ed., 1994.

[10] B. R. Saltzberg, "Intersymbol interference error bounds with application to ideal bandlimited signaling", *IEEE Trans. on Information Theory*, vol. 9, pp. 563–568, July 1968.

[11] R. Gitlin, J. Hayes, and S. Weinstein, *Data communication principles*. New York: Plenum Press, 1992.

[12] M. C. Jeruchim, P. Balaban, and K. S. Shanmugan, *Simulation of communication systems*. New York: Plenum Press, 1992.

Appendix 7.A *Line codes for PAM systems*

The functions of line codes are:

1. to shape the spectrum of the transmitted signals, and match it to the characteristics of the channel (see (7.17)); this task may be performed also by the transmit filter;
2. to facilitate synchronization at the receiver, especially in case the information message contains long sequences of ones or zeros;
3. to improve system performance in terms of P_e.

This appendix is divided in two parts: in the first, several representations of binary symbols are listed; in the second, partial response systems are introduced. For in-depth study and analysis of spectral properties of line codes we refer to the bibliography, in particular [1, 5].

7.A.1 Line codes

With reference to Figure 7.37, the binary sequence $\{b_\ell\}$, $b_\ell \in \{0, 1\}$, could be directly generated by a source, or be the output of a channel encoder. The sequence $\{a_k\}$ is produced by a line encoder. The channel input is a PAM signal $s(t)$, obtained by modulating a rectangular pulse h_{Tx}.

Non-return-to-zero (NRZ) format

The main feature of the NRZ family is that NRZ signals are antipodal signals: therefore NRZ line codes are characterized by the lowest error probability, for transmission over AWGN channels in the absence of ISI. Four formats are illustrated in Figure 7.38.

1. NRZ level (NRZ-L) or, simply, NRZ:
   "1" and "0" are represented by two different levels.
2. NRZ mark (NRZ-M):
   "1" is represented by a level transition, "0" by no level transition.
3. NRZ space (NRZ-S):
   "1" is represented by no level transition, "0" by a level transition.
4. Dicode NRZ:
   A change of polarity in the sequence $\{b_\ell\}$, "1-0" or "0-1", is represented by a level transition; every other case is represented by the zero level.

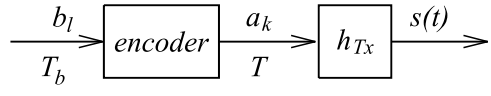


Figure 7.37. PAM transmitter with line encoder.

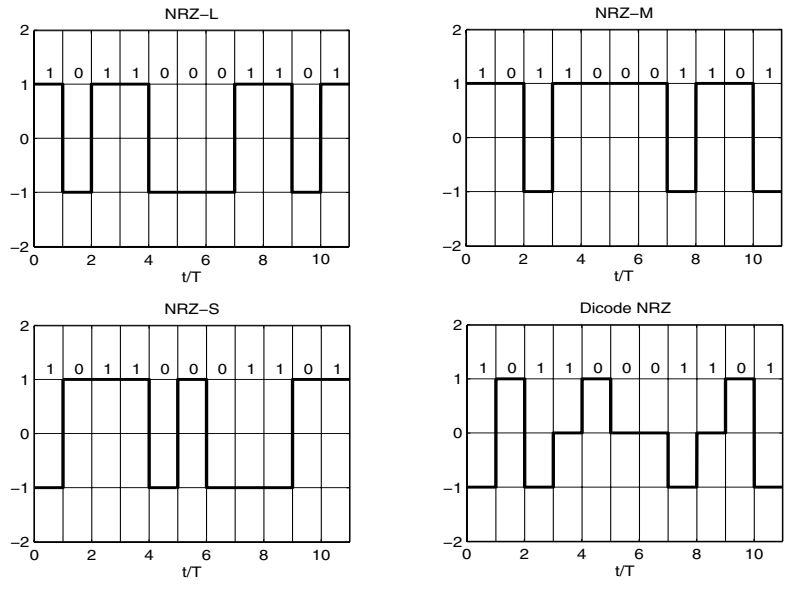


Figure 7.38. NRZ line codes.

Return-to-zero (RZ) format

1. Unipolar RZ:
   "1" is represented by a pulse having duration equal to half a bit interval, "0" by a zero pulse; we observe that the signal does not have zero mean. This property is usually not desirable, as, for example, for transmission over coaxial cables.

2. Polar RZ:
   "1" and "0" are represented by opposite pulses with duration equal to half a bit interval.

3. Bipolar RZ or *alternate mark inversion* (AMI):
   Bits equal to "1" are represented by rectangular pulses having duration equal to half a bit interval, sequentially alternating in sign, bits equal to "0" by the zero level.

4. Dicode RZ:
   A change of polarity in the sequence $\{b_\ell\}$, "1-0" or "0-1", is represented by a level transition, using a pulse having duration equal to half a bit interval; every other case is represented by the zero level.

RZ line codes are illustrated in Figure 7.39.

Biphase (B-ϕ) format

1. Biphase level (B-ϕ-L) or Manchester NRZ:
   "1" is represented by a transition from high level to low level, "0" by a transition from low level to high level. Long sequences of ones or zeros in the sequence $\{b_\ell\}$

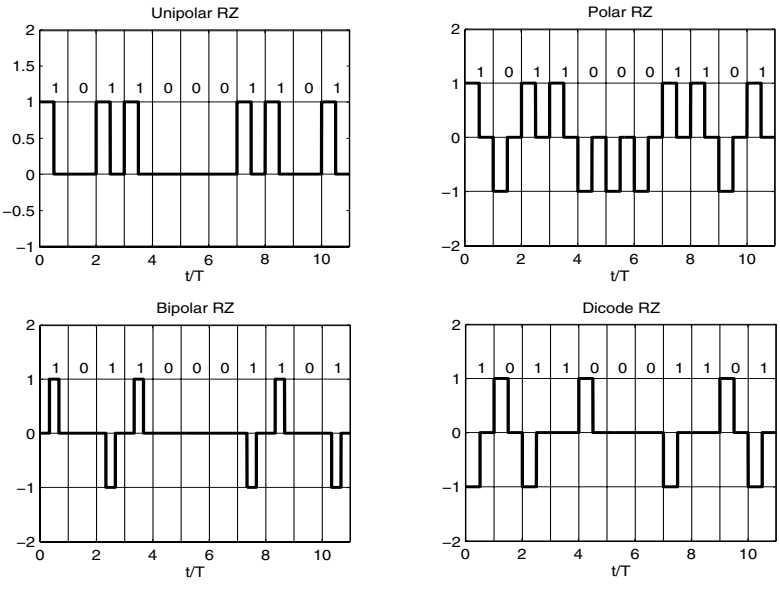


Figure 7.39. RZ line codes.

do not create synchronization problems. It is easy to see, however, that this line code leads to a doubling of the transmission bandwidth.

2. Biphase mark (B-ϕ-M) or Manchester 1:
   A transition occurs at the beginning of every bit interval; "1" is represented by a second transition within the bit interval, "0" is represented by a constant level.

3. Biphase space (B-ϕ-S):
   A transition occurs at the beginning of every bit interval; "0" is represented by a second transition within the bit interval, "1" is represented by a constant level.

Biphase line codes are illustrated in Figure 7.40.

Delay modulation or Miller code

"1" is represented by a transition at midpoint of the bit interval, "0" is represented by a constant level; if "0" is followed by another "0", a transition occurs at the end of the bit interval. This code shapes the spectrum similar to the Manchester code, but requires a lower bandwidth. The delay modulation line code is illustrated in Figure 7.40.

Block line codes

The input sequence $\{b_\ell\}$ is divided into blocks of K bits. Each block of K bits is then mapped into a block of N symbols belonging to an alphabet of cardinality M, with

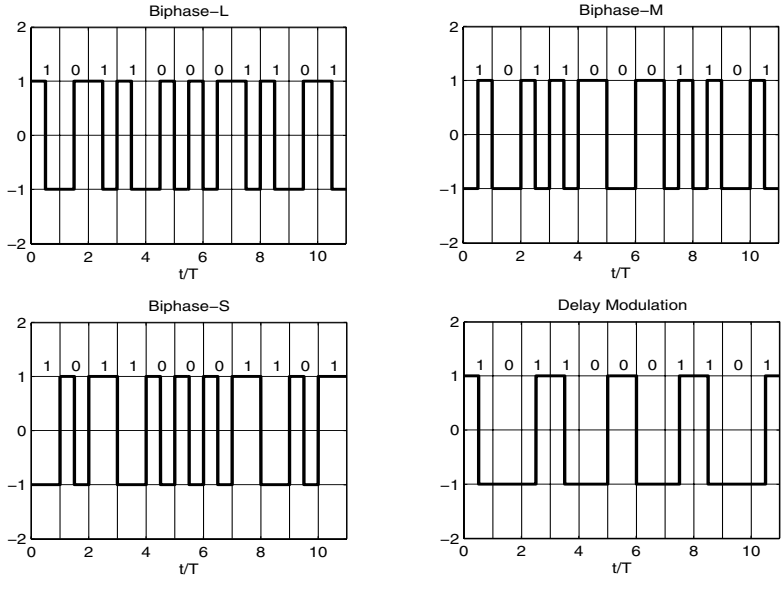


Figure 7.40. B-ϕ and delay modulation line codes.

the constraint

$$2^K \leq M^N \tag{7.160}$$

The KBNT codes are an example of block line codes where the output symbol alphabet is ternary $\{-1, 0, 1\}$.

Alternate mark inversion (AMI)

We consider a *differential binary* encoder, that is

$$a_k = b_k - b_{k-1} \qquad \text{with} \quad b_k \in \{0, 1\} \tag{7.161}$$

At the decoder the bits of the information sequence may be recovered by

$$\hat{b}_k = \hat{a}_k + \hat{b}_{k-1} \tag{7.162}$$

Note that $a_k \in \{-1, 0, 1\}$; in particular

$$a_k = \begin{cases} \pm 1 & \text{if } b_k \neq b_{k-1} \\ 0 & \text{if } b_k = b_{k-1} \end{cases} \tag{7.163}$$

From (7.161), the relation between the PSDs of the sequences $\{a_k\}$ and $\{b_k\}$ is given by

$$\begin{aligned} \mathcal{P}_a(f) &= \mathcal{P}_b(f)\,|1 - e^{-j2\pi fT}|^2 \\ &= \mathcal{P}_b(f)\,4\sin^2(\pi fT) \end{aligned}$$

Therefore $\mathcal{P}_a(f)$ exhibits zeros at frequencies that are integer multiples of $1/T$, in particular at $f = 0$. Moreover, from (7.161) we have $\mathtt{m}_a = 0$, independently of the distribution of $\{b_k\}$.

If the power of the transmitted signals is constrained, a disadvantage of the encoding method (7.161) is a reduced noise immunity with respect to antipodal transmission, that is for $a_k \in \{-1, 1\}$, because a detector at the receiver must now decide among three levels. Moreover, long sequences of information bits $\{\hat{b}_k\}$ that are all equal to 1 or 0 generate sequences of symbols $\{a_k\}$ that are all equal: this is not desirable for synchronization.

In any case, the biggest problem is the error propagation at the decoder, which, observing (7.162), given that an error occurs in $\{\hat{a}_k\}$, generates a sequence of bits $\{\hat{b}_k\}$ that are in error until another error occurs in $\{\hat{a}_k\}$. This problem can be solved by precoding: from the sequence of bits $\{b_k\}$ we first generate the sequence of bits $\{c_k\}$, with $c_k \in \{0, 1\}$, by

$$c_k = b_k \oplus c_{k-1} \tag{7.164}$$

where $\oplus$ denotes the modulo 2 sum. Next,

$$a_k = c_k - c_{k-1} \tag{7.165}$$

with $a_k \in \{-1, 0, 1\}$. Hence, it results in

$$a_k = \begin{cases} \pm 1 & \text{if } b_k = 1 \\ 0 & \text{if } b_k = 0 \end{cases} \tag{7.166}$$

In other words, a bit $b_k = 0$ is mapped into the symbol $a_k = 0$, and a bit $b_k = 1$ is mapped alternately in $a_k = +1$ or $a_k = -1$. Consequently, from (7.166) decoding may be performed simply by taking the magnitude of the detected symbol:

$$\hat{b}_k = |\hat{a}_k| \tag{7.167}$$

It is easy to prove that for a message $\{b_k\}$ with statistically independent symbols, and $p = P[b_k = 1]$, we have

$$P_a\left(e^{j2\pi fT}\right) = 2p(1-p)\,\frac{\sin^2(\pi fT)}{p^2 + (1-2p)\sin^2 \pi fT} \tag{7.168}$$

The plot of $P_a\left(e^{j2\pi fT}\right)$ is shown in Figure 7.41 for different values of p. Note that the PSD presents a zero at $f = 0$. Also in this case $\mathtt{m}_a = 0$.

We observe that the AMI line code is a particular case of the partial response system named *dicode* [6].

7.A.2 Partial response systems

From Section 7.1, we recall in Figure 7.42 the block diagram of a baseband transmission system, where the symbols $\{a_k\}$ belong to the following alphabet[12] of cardinality M:

$$a_k \in \mathcal{A} = \{-(M-1), -(M-3), \ldots, (M-3), (M-1)\} \tag{7.169}$$

and $w(t)$ is an additive white Gaussian noise.

[12] In the present analysis only M-PAM systems are considered; for M-QAM systems the results can be extended to the signals on the I and Q branches.

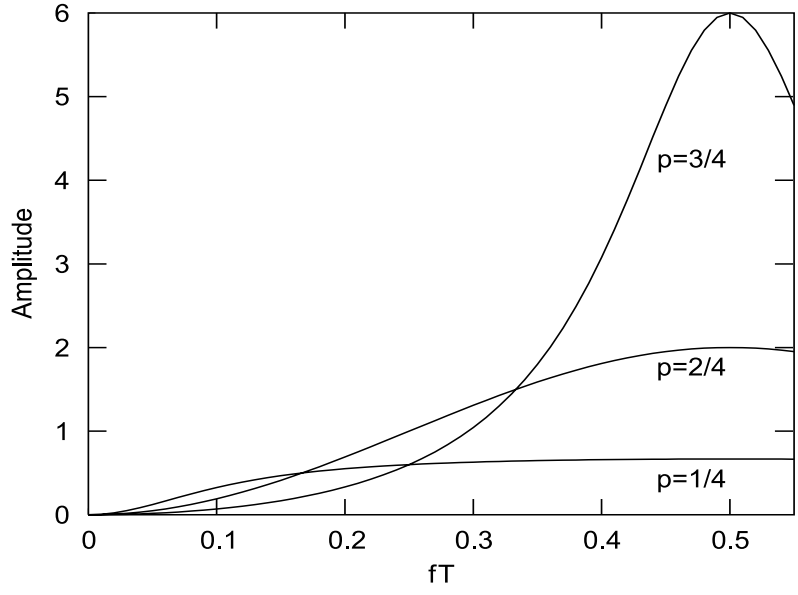


Figure 7.41. Power spectral density $P_a(e^{j2\pi fT})$ of an AMI encoded message.

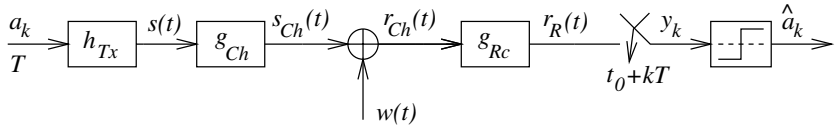


Figure 7.42. Block diagram of a baseband transmission system.

We assume that the transmission channel is ideal: the overall system can then be represented as an interpolator filter having impulse response

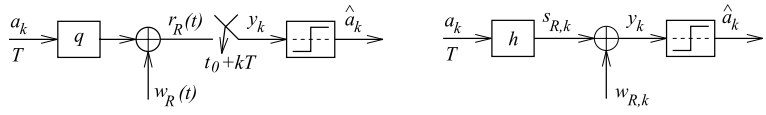
$$q(t) = h_{Tx} * g_{Rc}(t) \tag{7.170}$$

A noise signal $w_R(t)$, obtained by filtering $w(t)$ by the receive filter, is added to the desired signal. Sampling the received signal at instants t_0+kT yields the sequence $\{y_k\}$, as illustrated in Figure 7.43a. The discrete-time equivalent of the system is shown in Figure 7.43b, where $\{h_i = q(t_0+iT)\}$, and $w_{R,k} = w_R(t_0+kT)$. We assume that $\{h_i\}$ is equal to zero for $i < 0$ and $i \geq N$.

The *partial response (PR) polynomial* of the system is defined as

$$l(D) = \sum_{i=0}^{N-1} \mathtt{l}_i \, D^i \tag{7.171}$$

where the coefficients $\{\mathtt{l}_i\}$ are equal to the samples $\{h_i\}$, and D is the unit delay operator.



(a) Analog equivalent (b) Digital equivalent

Figure 7.43. Equivalent schemes to the system of Figure 7.42.

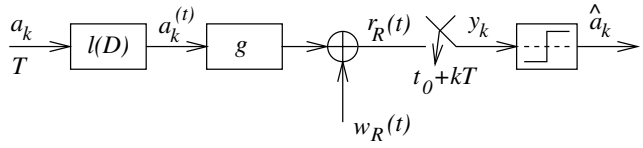


Figure 7.44. PR version of the system of Figure 7.42.

A PR system is illustrated in Figure 7.44, where $l(D)$ is defined in (7.171), and g is an analog filter satisfying the Nyquist criterion for the absence of ISI,

$$\sum_{m=-\infty}^{+\infty} \mathcal{G}\left(f - \frac{m}{T}\right) = T \tag{7.172}$$

The symbols at the output of the filter $l(D)$ in Figure 7.44 are given by

$$a_k^{(t)} = \sum_{i=0}^{N-1} \mathtt{l}_i \, a_{k-i} \tag{7.173}$$

Note that the overall scheme of Figure 7.44 is equivalent to that of Figure 7.43a with

$$q(t) = \sum_{i=0}^{N-1} \mathtt{l}_i \, g(t - iT) \tag{7.174}$$

Also, observing (7.172), the equivalent discrete-time model is obtained for $h_i = \mathtt{l}_i$.

In other words, from (7.174) the system of Figure 7.42 is decomposed into two parts:

- a filter with frequency response $l(e^{-j2\pi fT})$, periodic of period $1/T$, that forces the system to have an overall discrete-time impulse response equal to $\{h_i\}$;
- an analog filter g that does not modify the overall filter $h(D)$ and limits the system bandwidth.

As it will be clear from the analysis, the decomposition of Figure 7.44, on one hand, allows simplification of the study of the properties of the filter $h(D)$, and, on the other, to design an efficient receiver.

The scheme of Figure 7.44 suggests two possible ways to implement the system of Figure 7.42:

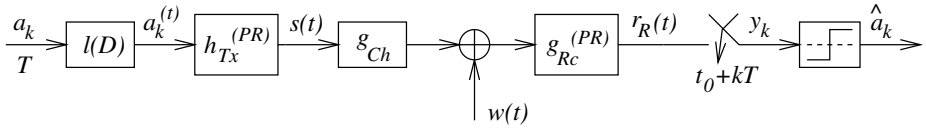


Figure 7.45. Implementation of a PR system using a digital filter.

1. *Analog*: the system is implemented in analog form; therefore the transmit filter h_{Tx} and the receive filter g_{Rc} must satisfy the relation

$$\mathcal{H}_{Tx}(f)\,\mathcal{G}_{Rc}(f) = \mathcal{Q}(f) = l(e^{-j2\pi fT})\,\mathcal{G}(f) \tag{7.175}$$

2. *Digital*: the filter $l(D)$ is implemented as a component of the transmitter by a digital filter; then the transmit filter $h_{Tx}^{(PR)}$ and receive filter $g_{Rc}^{(PR)}$ must satisfy the relation

$$\mathcal{H}_{Tx}^{(PR)}(f)\,\mathcal{G}_{Rc}^{(PR)}(f) = \mathcal{G}(f) \tag{7.176}$$

The implementation of a PR system using a digital filter is shown in Figure 7.45. Note from (7.172) that in both relations (7.175) and (7.176) g is a Nyquist filter.

The choice of the PR polynomial

Several considerations lead to the selection of the polynomial $l(D)$.

a) *System bandwidth.* With the aim of maximizing the transmission bit rate, many PR systems are designed for *minimum bandwidth*, i.e. from (7.175) it must be

$$l(e^{-j2\pi fT})\,\mathcal{G}(f) = 0 \qquad |f| > \frac{1}{2T} \tag{7.177}$$

Substitution of (7.177) into (7.172) yields the following conditions on the filter g:

$$\mathcal{G}(f) = \begin{cases} T & |f| \le \dfrac{1}{2T} \\ 0 & \text{elsewhere} \end{cases} \qquad \xleftrightarrow{\mathcal{F}^{-1}} \qquad g(t) = \operatorname{sinc}\left(\frac{t}{T}\right) \tag{7.178}$$

Correspondingly, observing (7.174) the filter q assumes the expression

$$q(t) = \sum_{i=0}^{N-1} \mathtt{l}_i \operatorname{sinc}\left(\frac{t - iT}{T}\right) \tag{7.179}$$

b) *Spectral zeros at* $f = 1/(2T)$. From the theory of signals, it is known that if $\mathcal{Q}(f)$ and its first $(n-1)$ derivatives are continuous and the n-th derivative is discontinuous, then $|q(t)|$ asymptotically decays as $1/|t|^{n+1}$. The continuity of $\mathcal{Q}(f)$ and of its derivatives helps to reduce the portion of energy contained in the tails of $q(t)$.

It is easily proven that in a *minimum bandwidth system*, the $(n-1)$-th derivative of $\mathcal{Q}(f)$ is continuous if and only if $l(D)$ has $(1+D)^n$ as a factor. On the other hand, if $l(D)$ has a zero of multiplicity greater than one in $D = -1$, then the transition band of $\mathcal{G}(f)$ around $f = 1/(2T)$ can be widened, thus simplifying the design of the analog filters.

c) *Spectral zeros at* $f = 0$. A transmitted signal with attenuated spectral components at low frequencies is desirable in many cases, e.g. for the implementation of SSB modulators (see Example 1.7.4 on page 58), or for transmission over channels with frequency responses that exhibit a spectral null at the frequency $f = 0$. Note that a zero of $l(D)$ in $D = 1$ corresponds to a zero of $l(e^{-j2\pi fT})$ at $f = 0$.

d) *Number of output levels.* From (7.173), the symbols at the output of the filter $l(D)$ have an alphabet $\mathcal{A}^{(t)}$ of cardinality $M^{(t)}$. If we indicate with $n_{\mathtt{l}}$ the number of coefficients of $l(D)$ different from zero, then the following inequality for $M^{(t)}$ holds

$$n_{\mathtt{l}}(M-1)+1 \leq M^{(t)} \leq M^{n_{\mathtt{l}}} \tag{7.180}$$

In particular, if the coefficients $\{\mathtt{l}_i\}$ are all equal, then $M^{(t)} = n_{\mathtt{l}}(M-1)+1$.

We note that, if $l(D)$ contains more than one factor $(1 \pm D)$, then $n_{\mathtt{l}}$ increases and, observing (7.180), also the number of output levels increases. If the power of the transmitted signal is constrained, detection of the sequence $\{a_k^{(t)}\}$ by a threshold detector will cause a loss in system performance.

e) *Some examples of minimum bandwidth systems.* In the case of *minimum bandwidth* systems, it is possible to evaluate the expression of $\mathcal{Q}(f)$ and $q(t)$ once the polynomial $l(D)$ has been selected.

As the coefficients $\{\mathtt{l}_i\}$ are generally symmetric or antisymmetric around $i = (N-1)/2$, it is convenient to consider the time-shifted pulse

$$\begin{aligned} \tilde{q}(t) &= q\left(t - \frac{(N-1)T}{2}\right) \\ \tilde{\mathcal{Q}}(f) &= e^{j\pi f(N-1)T}\,\mathcal{Q}(f) \end{aligned} \tag{7.181}$$

In Table 7.2 the more common polynomials $l(D)$ are described, as well as the corresponding expressions of $\tilde{\mathcal{Q}}(f)$ and $\tilde{q}(t)$, and the cardinality $M^{(t)}$ of the output alphabet $\mathcal{A}^{(t)}$. In the next three examples, polynomials $l(D)$ that are often found in practical applications of PR systems are considered.

Example 7.A.1 (Dicode filter)
The dicode filter introduces a zero at frequency $f = 0$ and has the following expression

$$l(D) = 1 - D \tag{7.182}$$

The frequency response, obtained by setting $D = e^{-j2\pi fT}$, is given by

$$l(e^{-j2\pi fT}) = 2j\, e^{-j\pi fT} \sin(\pi fT) \tag{7.183}$$

Example 7.A.2 (Duobinary filter)
The duobinary filter introduces a zero at frequency $f = 1/(2T)$ and has the following expression

$$l(D) = 1 + D \tag{7.184}$$

Table 7.2 Properties of several minimum bandwidth systems.

$l(D)$	$\tilde{\mathcal{Q}}(f)$ for $\lvert f\rvert \le 1/(2T)$	$\tilde{q}(t)$	$M^{(t)}$
$1+D$	$2T\cos(\pi fT)$	$\frac{4T^2}{\pi}\frac{\cos(\pi t/T)}{T^2-4t^2}$	$2M-1$
$1-D$	$j2T\sin(\pi fT)$	$\frac{8Tt}{\pi}\frac{\cos(\pi t/T)}{4t^2-T^2}$	$2M-1$
$1-D^2$	$j2T\sin(2\pi fT)$	$\frac{2T^2}{\pi}\frac{\sin(\pi t/T)}{t^2-T^2}$	$2M-1$
$1+2D+D^2$	$4T\cos^2(\pi fT)$	$\frac{2T^3}{\pi t}\frac{\sin(\pi t/T)}{T^2-t^2}$	$4M-3$
$1+D-D^2-D^3$	$j4T\cos(\pi fT)\sin(2\pi fT)$	$-\frac{64T^3t}{\pi}\frac{\cos(\pi t/T)}{(4t^2-9T^2)(4t^2-T^2)}$	$4M-3$
$1-D-D^2+D^3$	$-4T\sin(\pi fT)\sin(2\pi fT)$	$\frac{16T^2}{\pi}\frac{\cos(\pi t/T)(4t^2-3T^2)}{(4t^2-9T^2)(4t^2-T^2)}$	$4M-3$
$1-2D^2+D^4$	$-4T\sin^2(2\pi fT)$	$\frac{8T^3}{\pi t}\frac{\sin(\pi t/T)}{t^2-4T^2}$	$4M-3$
$2+D-D^2$	$T+T\cos(2\pi fT)+j3T\sin(2\pi fT)$	$\frac{T^2}{\pi t}\sin(\pi t/T)\left(\frac{3t-T}{t^2-T^2}\right)$	$4M-3$
$2-D^2-D^4$	$-T+T\cos(4\pi fT)+j3T\sin(4\pi fT)$	$\frac{2T^2}{\pi t}\sin(\pi t/T)\left(\frac{2T-3t}{t^2-4T^2}\right)$	$4M-3$

The frequency response is given by

$$l(e^{-j2\pi fT}) = 2e^{-j\pi fT}\cos(\pi fT) \tag{7.185}$$

Observing (7.179) we have

$$q(t) = \operatorname{sinc}\left(\frac{t}{T}\right) + \operatorname{sinc}\left(\frac{t-T}{T}\right) \tag{7.186}$$

The plot of the impulse response of a duobinary filter is shown in Figure 7.46 with a continuous line. We notice that the tails of the two sinc functions cancel each other, in line with what was stated at point b) regarding the aymptotical decay of the pulse of a PR system with a zero in $D=-1$.

Example 7.A.3 (Modified duobinary filter)
The modified duobinary filter combines the characteristics of duobinary and dicode filters, and has the following expression

$$l(D) = (1-D)(1+D) = 1-D^2 \tag{7.187}$$

The frequency response becomes

$$l(e^{-j2\pi fT}) = 1-e^{-j4\pi fT} = 2j\,e^{-j2\pi fT}\sin(2\pi fT) \tag{7.188}$$

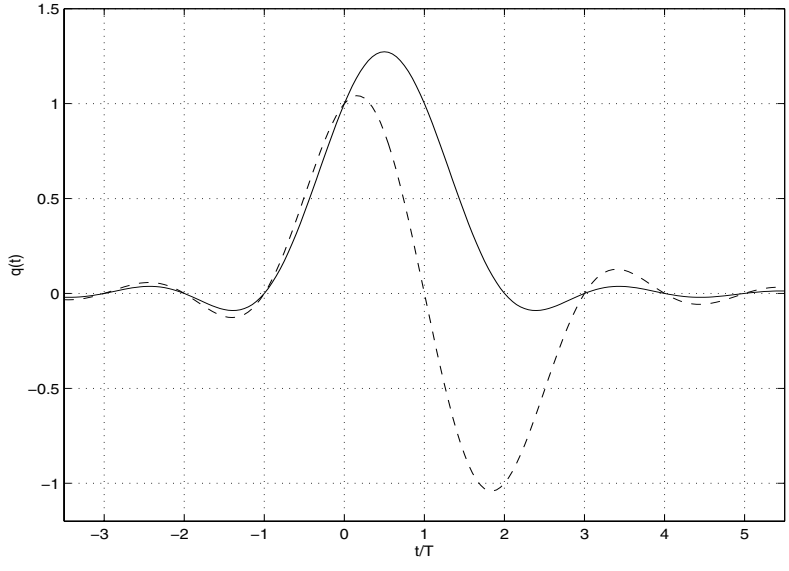


Figure 7.46. Plot of $q(t)$ for duobinary (–) and modified duobinary (- -) filters.

Using (7.179) it results in

$$q(t) = \text{sinc}\left(\frac{t}{T}\right) - \text{sinc}\left(\frac{t-2T}{T}\right) \tag{7.189}$$

The plot of the impulse response of a modified duobinary filter is shown in Figure 7.46 with a dashed line.

f) *Transmitted signal spectrum.* With reference to the PR system of Figure 7.45, the spectrum of the transmitted signal is given by (see (7.17))

$$\mathcal{P}_s(f) = \left|\frac{1}{T}\, l(e^{-j2\pi fT})\, \mathcal{H}_{Tx}^{(PR)}(f)\right|^2 \mathcal{P}_a(f) \tag{7.190}$$

For a minimum bandwidth system, with $\mathcal{H}_{Tx}^{(PR)}(f)$ given by (7.178), (7.190) simplifies into

$$\mathcal{P}_s(f) = \begin{cases} |l(e^{-j2\pi fT})|^2\, \mathcal{P}_a(f) & |f| \le \dfrac{1}{2T} \\ 0 & |f| > \dfrac{1}{2T} \end{cases} \tag{7.191}$$

In Figure 7.47 the PSD of a minimum bandwidth PR system is compared with that of a PAM system. The spectrum of the sequence of symbols $\{a_k\}$ is assumed white. For the PR system, a modified duobinary filter is considered, so that the spectrum is obtained as the

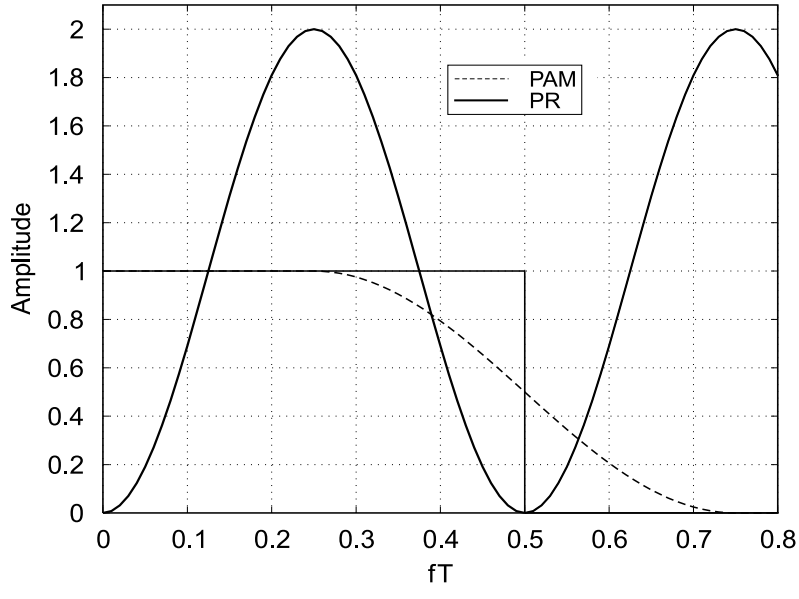


Figure 7.47. PSD of a modified duobinary PR system and of a PAM system.

product of the functions $|l(-e^{j2\pi fT})|^2 = |2\sin(2\pi fT)|^2$ and $|\mathcal{H}_{Tx}^{(PR)}(f)|^2 = T^2\,\mathrm{rect}(fT)$, plotted with continuous lines. For the PAM system, the transmit filter h_{Tx} is a square root raised cosine with roll-off factor $\rho = 0.5$, and the spectrum is plotted with a dashed line.

Symbol detection and error probability

We consider the discrete-time equivalent scheme of Figure 7.43b; the signal $s_{R,k}$ can be expressed as a function of symbols $\{a_k\}$ and coefficients $\{\mathtt{l}_i\}$ of the filter $l(D)$ in the following form

$$s_{R,k} = a_k^{(t)} = \mathtt{l}_0\, a_k + \sum_{i=1}^{N-1} \mathtt{l}_i\, a_{k-i} \tag{7.192}$$

The term $\mathtt{l}_0\, a_k$ is the desired part of the signal $s_{R,k}$, whereas the summation represents the ISI term that is often designated as "controlled ISI", as it is deliberately introduced.

The receiver detects the symbols $\{a_k\}$ using the sequence of samples $\{y_k = a_k^{(t)} + w_{R,k}\}$. We discuss four possible solutions.[13]

1. *LE-ZF*. A zero-forcing linear equalizer (LE-ZF) having D transform equal to $1/l(D)$ is used. At the equalizer output, at instant k the symbol a_k plus a noise term is

[13] For a first reading it is suggested that only solution 3 is considered. The study of the other solutions should be postponed until the equalization methods of Chapter 8 are examined.

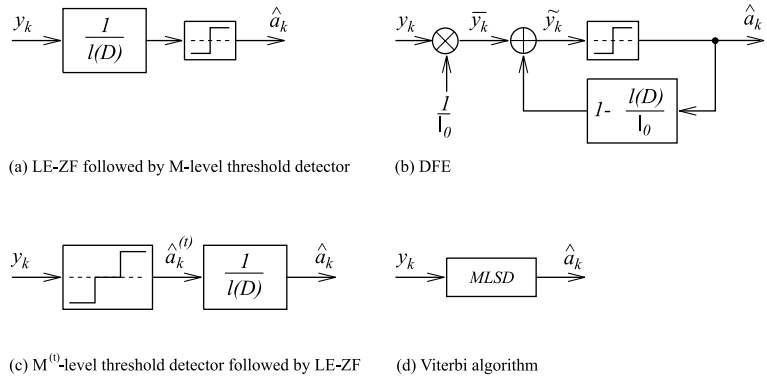


Figure 7.48. Four possible solutions to the detection problem in the presence of *controlled* ISI.

obtained; the detected symbols $\{\hat{a}_k\}$ are obtained by an M-level threshold detector, as illustrated in Figure 7.48a. We note, however, that the amplification of noise by the filter $1/l(D)$ is infinite at frequencies f such that $l(e^{-j2\pi fT}) = 0$.

2. *DFE.* A second solution resorts to a decision-feedback equalizer (DFE), as shown in Figure 7.48b. An M-level threshold detector is also employed by the DFE, but there is no noise amplification as the ISI is removed by the feedback filter, having D transform equal to $1 - l(D)/l_0$.
   We observe that at the decision point the signal $\bar{y}_k$ has the expression

$$\tilde{y}_k = \frac{1}{l_0}\left(a_k^{(t)} + w_{R,k} - \sum_{i=1}^{N-1} l_i\, \hat{a}_{k-i}\right) \tag{7.193}$$

   If we indicate with $e_k = a_k - \hat{a}_k$ a detection error, then substituting (7.192) in (7.193), we obtain

$$\tilde{y}_k = a_k + \frac{1}{l_0}\left(w_{R,k} + \sum_{i=1}^{N-1} l_i\, e_{k-i}\right) \tag{7.194}$$

   The equation (7.194) shows that a wrong decision negatively influence successive decisions: this phenomenon is known as *error propagation*.

3. *Threshold detector with $M^{(t)}$ levels.* This solution, shown in Figure 7.48c, exploits the $M^{(t)}$-ary nature of the symbols $a_k^{(t)}$, and makes use of a threshold detector with $M^{(t)}$ levels followed by a LE-ZF. This structure does not lead to noise amplification as solution 1, because the noise is eliminated by the threshold detector; however, there is still the problem of error propagation.

4. *Viterbi algorithm.* This solution, shown in Figure 7.48d, corresponds to maximum-likelihood sequence detection (MLSD) of $\{a_k\}$. It yields the best performance.

Solution 2 using the DFE is often adopted in practice: in fact it avoids noise amplification and is simpler to implement than the Viterbi algorithm. However, the problem of *error propagation* remains.

In this case, using (7.194) the error probability can be written as

$$P_e = \left(1 - \frac{1}{M}\right) P\left[\left| w_{R,k} + \sum_{i=1}^{N-1} \mathtt{l}_i \, e_{k-i} \right| > \mathtt{l}_0\right] \tag{7.195}$$

A *lower bound* $P_{e,L}$ can be computed for P_e by assuming the error propagation is absent, or setting $\{e_k\} = 0$, $\forall k$, in (7.195). If we denote by σ_{w_R} the standard deviation of the noise $w_{R,k}$, we obtain

$$P_{e,L} = 2\left(1 - \frac{1}{M}\right) Q\left(\frac{\mathtt{l}_0}{\sigma_{w_R}}\right) \tag{7.196}$$

Assuming $w_{R,k}$ white noise, an *upper bound* $P_{e,U}$ is given in [7] in terms of $P_{e,L}$:

$$P_{e,U} = \frac{M^{N-1} \, P_{e,L}}{(M/(M-1)) \, P_{e,L}(M^{N-1} - 1) + 1} \tag{7.197}$$

From (7.197) we observe that the effect of the error propagation is that of increasing the error probability by a factor M^{N-1} with respect to $P_{e,L}$.

A solution to the problem of error propagation is represented by *precoding*, which will be investigated in depth in Chapter 13.

Precoding

We make use here of the following two simplifications:

1. the coefficients $\{\mathtt{l}_i\}$ are integer numbers;
2. the symbols $\{a_k\}$ belong to the alphabet $\mathcal{A} = \{0, 1, \ldots, M-1\}$; this choice is made because arithmetic modulo M is employed.

We define the sequence of precoded symbols $\{\bar{a}_k^{(p)}\}$ as:

$$\bar{a}_k^{(p)} \, \mathtt{l}_0 = \left(a_k - \sum_{i=1}^{N-1} \mathtt{l}_i \, \bar{a}_{k-i}^{(p)}\right) \bmod M \tag{7.198}$$

We note that (7.198) has only one solution if and only if $\mathtt{l}_0$ and M are relatively prime [8]. In case $\mathtt{l}_0 = \cdots = \mathtt{l}_{j-1} = 0 \bmod M$, and $\mathtt{l}_j$ and M are relatively prime, (7.198) becomes

$$\bar{a}_{k-j}^{(p)} \, \mathtt{l}_j = \left(a_k - \sum_{i=j+1}^{N-1} \mathtt{l}_i \, \bar{a}_{k-i}^{(p)}\right) \bmod M \tag{7.199}$$

For example, if $l(D) = 2 + D - D^2$ and $M = 2$, (7.198) is not applicable as $\mathtt{l}_0 \bmod M = 0$. Therefore (7.199) is used.

Applying the PR filter to $\{\bar{a}_k^{(p)}\}$ we obtain the sequence

$$a_k^{(t)} = \sum_{i=0}^{N-1} l_i \, \bar{a}_{k-i}^{(p)} \tag{7.200}$$

From the comparison between (7.198) and (7.200), or in general (7.199), we have the fundamental relation

$$a_k^{(t)} \bmod M = a_k \tag{7.201}$$

Equation (7.201) shows that, as in the absence of noise we have $y_k = a_k^{(t)}$, the symbol a_k can be detected by considering the received signal y_k modulo M; this operation is *memoryless*, therefore the detection of $\hat{a}_k$ is independent of the previous detections $\{\hat{a}_{k-i}\}$, $i = 1, \ldots, N-1$. Therefore the problem of error propagation is solved. Moreover, the desired signal is not affected by ISI.

If the instantaneous transformation

$$a_k^{(p)} = 2\bar{a}_k^{(p)} - (M-1) \tag{7.202}$$

is applied to the symbols $\{\bar{a}_k^{(p)}\}$, then we obtain a sequence of symbols that belong to the alphabet $\mathcal{A}^{(p)}$ in (7.169). The sequence $\{a_k^{(p)}\}$ is then input to the filter $l(D)$. *Precoding* consists of the operation (7.198) followed by the transformation (7.202).

However, we note that (7.201) is no longer valid. From (7.202), (7.200), and (7.198), we obtain the new *decoding* operation, given by

$$a_k = \left(\frac{a_k^{(t)}}{2} + \mathcal{K} \right) \bmod M \tag{7.203}$$

where

$$\mathcal{K} = (M-1) \sum_{i=0}^{N-1} l_i \tag{7.204}$$

A PR system with precoding is illustrated in Figure 7.49. The receiver is constituted by a threshold detector with $M^{(t)}$ levels that provides the symbols $\{\hat{a}_k^{(t)}\}$, followed by a block that realizes (7.203) and yields the detected data $\{\hat{a}_k\}$.

Error probability with precoding

To evaluate the error probability of a system with precoding, the statistics of the symbols $\{a_k^{(t)}\}$ must be known; it is easy to prove that if the symbols $\{a_k\}$ are i.i.d., the symbols $\{a_k^{(t)}\}$ are also i.i.d.

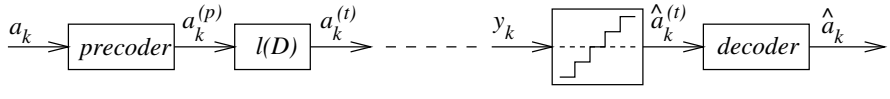


Figure 7.49. PR system with *precoding*.

If we assume that the cardinality of the set $\mathcal{A}^{(t)}$ is maximum, i.e. $M^{(t)} = M^{n_1}$, then the output levels are equally spaced and the symbols $a_k^{(t)}$ result equally likely with probability

$$P[a_k^{(t)} = \alpha] = \frac{1}{M^{n_1}} \qquad \alpha \in \mathcal{A}^{(t)} \tag{7.205}$$

In general, however, the symbols $\{a_k^{(t)}\}$ are not equiprobable, because several output levels are redundant, as can be deduced from the following example.

Example 7.A.4 (Dicode filter)
We assume $M = 2$, therefore $a_k = \{0, 1\}$; the precoding law (7.198) is simply an *exclusive or* and

$$\bar{a}_k^{(p)} = a_k \oplus \bar{a}_{k-1}^{(p)} \tag{7.206}$$

The symbols $\{a_k^{(p)}\}$ are obtained from (7.202),

$$a_k^{(p)} = 2\bar{a}_k^{(p)} - 1 \tag{7.207}$$

they are antipodal as $a_k^{(p)} = \{-1, +1\}$. Finally, the symbols at the output of the filter $l(D)$ are given by

$$a_k^{(t)} = a_k^{(p)} - a_{k-1}^{(p)} = 2\left(\bar{a}_k^{(p)} - \bar{a}_{k-1}^{(p)}\right) \tag{7.208}$$

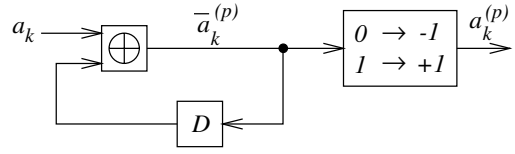
The values of $\bar{a}_{k-1}^{(p)}$, a_k, $\bar{a}_k^{(p)}$ and $a_k^{(t)}$ are given in Table 7.3. We observe that both output levels ± 2 correspond to the symbol $a_k = 1$ and therefore are redundant; the three levels are not equally likely. The symbol probabilities are given by

$$\begin{aligned} P[a_k^{(t)} = \pm 2] &= \tfrac{1}{4} \\ P[a_k^{(t)} = 0] &= \tfrac{1}{2} \end{aligned} \tag{7.209}$$

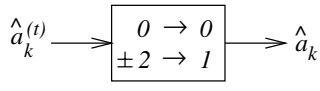
Figure 7.50a shows the precoder that realizes equations (7.206) and (7.207). The *decoder*, realized as a map that associates the symbol $\hat{a}_k = 1$ to ± 2, and the symbol $\hat{a}_k = 0$ to 0, is illustrated in Figure 7.50b.

Table 7.3 Precoding for the dicode filter.

$\bar{a}_{k-1}^{(p)}$	a_k	$\bar{a}_k^{(p)}$	$a_k^{(t)}$
0	0	0	0
0	1	1	+2
1	0	1	0
1	1	0	−2



(a) precoder



(b) decoder

Figure 7.50. Precoder and decoder for a dicode filter $I(D)$ with $M = 2$.

Alternative interpretation of PR systems

Up to now we have considered a general transmission system, and looked for an efficient design method. We now assume that the system is given, i.e. that the transmit filter as well as the receive filter are assigned. The scheme of Figure 7.44 can be regarded as a tool for the optimization of a given system where $l(D)$ includes the characteristics of the transmit and receive filters: as a result, the symbols $\{a_k^{(t)}\}$ no longer are the *transmitted* symbols, but are to be interpreted as the symbols that are *ideally received*. In the light of these considerations, the assumption of an ideal channel can also be removed. In this case the filter $l(D)$ will also include the ISI introduced by the channel.

We observe that the *precoding/decoding* technique is an alternative equalization method to the DFE that presents the advantage of eliminating error propagation, which can considerably deteriorate system performance.

In the following two examples [9], additive white Gaussian noise $w_{R,k} = \tilde{w}_k$ is assumed, and various systems are studied for the same signal-to-noise ratio at the receiver.

Example 7.A.5 (Ideal channel g)
a) *Antipodal signals.* We transmit a sequence of symbols from a binary alphabet, $a_k \in \{-1, 1\}$. The received signal is

$$y_k = a_k + \tilde{w}_{A,k} \tag{7.210}$$

where the variance of the noise is given by $\sigma^2_{\tilde{w}_A} = \sigma_I^2$.

At the receiver, using a threshold detector with threshold set to zero, we obtain

$$P_{bit} = Q\left(\frac{1}{\sigma_I}\right) \tag{7.211}$$

b) *Duobinary signal with precoding.* The transmitted signal is now given by $a_k^{(t)} = a_k^{(p)} + a_{k-1}^{(p)} \in \{-2, 0, 2\}$, where $a_k^{(p)} \in \{-1, 1\}$ is given by (7.202) and (7.198).

The received signal is given by

$$y_k = a_k^{(t)} + \tilde{w}_{B,k} \tag{7.212}$$

where the variance of the noise is $\sigma_{\tilde{w}_B}^2 = 2\sigma_I^2$, as $\sigma_{a_k^{(t)}}^2 = 2$.

At the receiver, using a threshold detector with thresholds set at ± 1, we have the following conditional error probabilities:

$$P[\mathrm{E} \mid a_k^{(t)} = 0] = 2Q\left(\frac{1}{\sigma_{\tilde{w}_B}}\right)$$

$$P[\mathrm{E} \mid a_k^{(t)} = -2] = P[\mathrm{E} \mid a_k^{(t)} = 2] = Q\left(\frac{1}{\sigma_{\tilde{w}_B}}\right)$$

Consequently, at the detector output we have

$$\begin{aligned} P_{bit} &= P[\hat{a}_k \neq a_k] \\ &= P[\mathrm{E} \mid a_k^{(t)} = 0]\tfrac{1}{2} + P[\mathrm{E} \mid a_k^{(t)} = \pm 2]\tfrac{1}{2} \\ &= 2Q\left(\frac{1}{\sqrt{2}\,\sigma_I}\right) \end{aligned}$$

We observe a worsening of about 3 dB in terms of the signal-to-noise ratio with respect to case a).

c) *Duobinary signal.* The transmitted signal is $a_k^{(t)} = a_k + a_{k-1}$. The received signal is given by

$$y_k = a_k + a_{k-1} + \tilde{w}_{C,k} \tag{7.213}$$

where $\sigma_{\tilde{w}_C}^2 = 2\sigma_I^2$. We consider using a receiver that applies MLSD to recover the data; from Example 8.12.1 on page 687 it results in

$$P_{bit} = K\, Q\left(\frac{\sqrt{8}}{2\sigma_{\tilde{w}_C}}\right) = K\, Q\left(\frac{1}{\sigma_I}\right) \tag{7.214}$$

where K is a constant.

We note that the PR system employing MLSD at the receiver achieves a performance similar to that of a system transmitting antipodal signals, as MLSD exploits the correlation between symbols of the sequence $\{a_k^{(t)}\}$.

Example 7.A.6 (Equivalent channel g of the type $1 + D$)
In this example it is the channel itself that forms a *duobinary* signal.

d) *Antipodal signals.* Transmitting $a_k \in \{-1, 1\}$, the received signal is given by

$$y_k = a_k + a_{k-1} + \tilde{w}_{D,k} \tag{7.215}$$

where $\sigma^2_{\tilde{w}_D} = 2\sigma_I^2$.

An attempt at pre-equalizing the signal at the transmitter by inserting a filter $l(D) = 1/(1 + D) = 1 - D + D^2 + \cdots$ would yield symbols $a_k^{(t)}$ with unlimited amplitude; therefore such a configuration cannot be used. Equalization at the receiver using the scheme of Figure 7.48a would require a filter of the type $1/(1+D)$, which would lead to unlimited noise enhancement.

Therefore we resort to the scheme of Figure 7.48c, where the threshold detector has thresholds set at ± 1. To avoid error propagation, we precode the message and transmit the sequence $\{a_k^{(p)}\}$ instead of $\{a_k\}$. At the receiver we have

$$y_k = a_k^{(p)} + a_{k-1}^{(p)} + \tilde{w}_{D,k} \tag{7.216}$$

We are therefore in the same conditions as in case b), and

$$P_{bit} = 2Q\left(\frac{1}{\sqrt{2}\,\sigma_I}\right) \tag{7.217}$$

e) *MLSD receiver.* To detect the sequence of information bits from the received signal (7.215), MLSD can be adopted. P_{bit} is in this case given by (7.214).

Appendix 7.B *Computation of P_e for some cases of interest*

7.B.1 P_e in the absence of ISI

In the absence of ISI, the signal at the decision point is the type (7.102)

$$y_k = h_0 a_k + w_{R,k} \qquad a_k \in \mathcal{A} \tag{7.218}$$

where $w_{R,k}$ is the sample of an additive noise signal. Assuming $\{w_{R,k}\}$ stationary with probability density function $p_w(\xi)$, from (7.218) for $a_k = \alpha_n \in \mathcal{A}$ we have

$$p_{y_k|a_k}(\rho \mid \alpha_n) = p_w(\rho - h_0\alpha_n) \tag{7.219}$$

Therefore the MAP criterion (6.26) becomes

$$\rho \in R_m \qquad \hat{a}_k = \alpha_m \qquad \text{if} \quad \alpha_m = \arg\max_{\alpha_n} p_n\, p_w(\rho - h_0\alpha_n) \tag{7.220}$$

We consider now the application of the MAP criterion to an M-PAM system, where

$$\alpha_n = 2n - 1 - M \qquad n = 1, \ldots, M \tag{7.221}$$

The decision regions $\{R_n\}$, $n = 1, \ldots, M$, are formed by intervals, or, in general, by the union of intervals, whose boundary points are called *decision thresholds* $\{\tau_i\}$, $i = 1, \ldots, M-1$.

Example 7.B.1 (Determination of the optimum decision threholds)
We consider a 4-PAM system with the following symbol probabilities:

$$\{p_1, p_2, p_3, p_4\} = \left\{\frac{3}{20}, \frac{3}{20}, \frac{1}{2}, \frac{1}{5}\right\} \tag{7.222}$$

The noise is assumed to have an exponential probability density function

$$p_w(\xi) = \frac{\beta}{2} e^{-|\xi|\beta} \tag{7.223}$$

where β is a constant; the variance of the noise is given by $\sigma_w^2 = 2/\beta^2$. The curves

$$p_n\, p_w(\rho - h_0\alpha_n) \qquad n = 1, \ldots, 4 \tag{7.224}$$

are illustrated in Figure 7.51. We note that, for the choice in (7.222) of the symbols probabilities, the decision thresholds, also shown in Figure 7.51, are obtained from the intersections between curves in (7.224) relative to two adjacent symbols; therefore they are given by the solutions of the $M-1$ equations

$$p_i\, p_w(\tau_i - h_0\alpha_i) = p_{i+1}\, p_w(\tau_i - h_0\alpha_{i+1}) \qquad i = 1, \ldots, M-1 \tag{7.225}$$

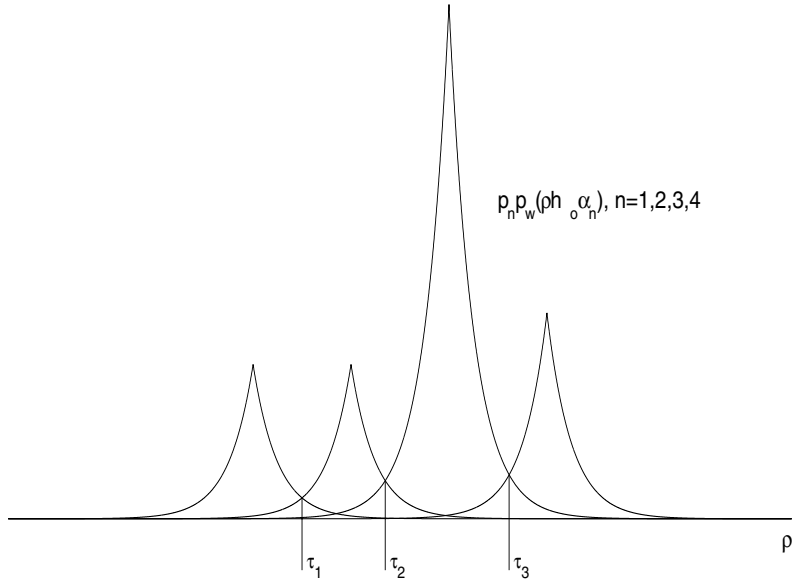


Figure 7.51. Optimum thresholds for a 4-PAM system with non-equally likely symbols.

We point out that, if the probability that the symbol ℓ is sent is very small, $p_\ell \ll 1$, the measure of the corresponding decision interval could be equal to zero, and consequently this symbol would never be detected. In this case the decision thresholds will be fewer than $M - 1$.

Example 7.B.2 (Computation of P_e for a 4-PAM system)
We indicate with $F_w(x)$ the probability distribution of $w_{R,k}$:

$$F_w(x) = \int_{-\infty}^{x} p_w(\xi)\, d\xi \tag{7.226}$$

For a M-PAM system with thresholds τ_1, τ_2, and τ_3, the probability of correct decision is given by (6.18):

$$\begin{aligned} P[\mathrm{C}] &= \sum_{n=1}^{4} p_n \int_{R_n} p_w(\rho - h_0\alpha_n)\, d\rho \\ &= p_1 \int_{-\infty}^{\tau_1} p_w(\rho - h_0\alpha_1)\, d\rho + p_2 \int_{\tau_1}^{\tau_2} p_w(\rho - h_0\alpha_2)\, d\rho \\ &\quad + p_3 \int_{\tau_2}^{\tau_3} p_w(\rho - h_0\alpha_3)\, d\rho + p_4 \int_{\tau_3}^{+\infty} p_w(\rho - h_0\alpha_4)\, d\rho \\ &= p_1[F_w(\tau_1 - h_0\alpha_1)] + p_2[F_w(\tau_2 - h_0\alpha_2) - F_w(\tau_1 - h_0\alpha_2)] \\ &\quad + p_3[F_w(\tau_3 - h_0\alpha_3) - F_w(\tau_2 - h_0\alpha_3)] + p_4[1 - F_w(\tau_3 - h_0\alpha_4)] \end{aligned} \tag{7.227}$$

We note that, if F_w is a continuous function, optimum thresholds can be obtained by equating to zero the derivative of the expression in (7.227) with respect to τ_1, τ_2, and τ_3.

In the case of equally likely symbols and equidistant thresholds, i.e.

$$\tau_i = h_0(2i - M) \qquad i = 1, \ldots, M-1 \tag{7.228}$$

equation (7.227) yields

$$P[\mathtt{C}] = 1 - 2\left(1 - \frac{1}{M}\right) F_w(-h_0) \tag{7.229}$$

We note that (7.229) is in agreement with (6.122) obtained for Gaussian noise.

7.B.2 P_e in the presence of ISI

We consider M-PAM transmission in the presence of ISI. We assume the symbols in (7.221) are equally likely and the decision thresholds are of the type given by (7.228). With reference to (7.65), the received signal at the decision point assumes the following expression:

$$y_k = h_0 a_k + \mathtt{i}_k + w_{R,k} \tag{7.230}$$

where $\mathtt{i}_k$ represents the ISI and is given by

$$\mathtt{i}_k = \sum_{i \neq 0} h_i a_{k-i} \tag{7.231}$$

and $w_{R,k}$ is Gaussian noise with statistical power σ^2 and statistically independent of the i.i.d. symbols of the message $\{a_k\}$.

We examine various methods to compute the symbol error probability in the presence of ISI.

Exhaustive method

We refer to the case of 4-PAM transmission with $N_i = 2$ interferers due to one non-zero precursor and one non-zero postcursor. Therefore we have

$$\mathtt{i}_k = a_{k-1} h_1 + a_{k+1} h_{-1} \tag{7.232}$$

We define the vector of symbols that contribute to ISI as

$$\mathbf{a}'_k = [a_{k-1}, a_{k+1}] \tag{7.233}$$

Then $\mathtt{i}_k$ can be written as a function of $\boldsymbol{a}'_k$ as

$$\mathtt{i}_k = \mathtt{i}(\mathbf{a}'_k) \tag{7.234}$$

Therefore, $\mathtt{i}_k$ is a random variable that assumes values in an alphabet with cardinality $L = M^{N_i} = 16$.

Starting from (7.230) the error probability can be computed by conditioning with respect to the values assumed by $\mathbf{a}'_k = [\alpha^{(1)}, \alpha^{(2)}] = \boldsymbol{\alpha} \in \mathcal{A}^2$. For equally likely symbols and thresholds given by (7.228) we have

$$\begin{aligned} P_e &= 2\left(1 - \frac{1}{M}\right) \sum_{\boldsymbol{\alpha} \in \mathcal{A}^2} Q\left(\frac{h_0 - \mathtt{i}(\mathbf{a}'_k)}{\sigma}\right) P[\mathbf{a}'_k = \boldsymbol{\alpha}] \\ &= 2\left(1 - \frac{1}{M}\right) \frac{1}{L} \sum_{\boldsymbol{\alpha} \in \mathcal{A}^2} Q\left(\frac{h_0 - \mathtt{i}(\boldsymbol{\alpha})}{\sigma}\right) \end{aligned} \tag{7.235}$$

This method gives the exact value of the error probability in the presence of interferers, but requires the computation of L terms. This method can be costly, especially if the number of interferers is large: it is therefore convenient to consider approximations of the error probability obtained by simpler computational methods.

Gaussian approximation

If interferers have a similar amplitude and their number is large, we can use the central limit theorem and approximate $\mathtt{i}_k$ as a Gaussian random variable. As the process $w_{R,k}$ is Gaussian, the process

$$z_k = \mathtt{i}_k + w_{R,k} \tag{7.236}$$

is also Gaussian with variance

$$\sigma_z^2 = \sigma_{\mathtt{i}}^2 + \sigma^2 \tag{7.237}$$

where $\sigma_{\mathtt{i}}^2$ is given by (7.72). Then

$$P_e = 2\left(1 - \frac{1}{M}\right) Q\left(\frac{h_0}{\sigma_z}\right) \tag{7.238}$$

It is seen that this method, although very convenient, is rather pessimistic, especially for large values of Γ. As a matter of fact, we observe that the amplitude of $\mathtt{i}_k$ is limited by the value

$$\mathtt{i}_{max} = (M-1) \sum_{i \neq 0} |h_i| \tag{7.239}$$

whereas the Gaussian approximation implies that the values of $\mathtt{i}_k$ are unlimited.

Worst-case bound

This method substitutes $\mathtt{i}_k$ with the constant $\mathtt{i}_{max}$ defined in (7.239). In this case P_e is equal to

$$P_e = 2\left(1 - \frac{1}{M}\right) Q\left(\frac{h_0 - \mathtt{i}_{max}}{\sigma}\right) \tag{7.240}$$

This bound is typically too pessimistic, however, it yields a good approximation if $\mathtt{i}_k$ is mainly due to one dominant interferer.

Saltzberg bound

With reference to (7.230), defining z_k as the total disturbance given by (7.236), in general we have

$$P_e = 2\left(1 - \frac{1}{M}\right) P[z_k > h_0] \tag{7.241}$$

Let

$$\alpha_{max} = \max_n \{\alpha_n\} = M - 1 \tag{7.242}$$

in the specific case, and $\mathcal{I}$ be any subset of the integers $\mathcal{Z}'$, excluding zero, such that

$$\sum_{i \in \mathcal{I}} |h_i| < \frac{h_0}{\alpha_{max}} \tag{7.243}$$

Moreover, let $\mathcal{I}^C$ be the complementary set of $\mathcal{I}$ with respect to $\mathcal{Z}'$. Saltzberg applied a Chernoff bound to the probability $P[z_k > h_0]$ [10], obtaining

$$P[z_k > h_0] < \exp\left(-\frac{\left(h_0 - \alpha_{max} \sum_{i \in \mathcal{I}} |h_i|\right)^2}{2\left(\sigma^2 + \sigma_a^2 \sum_{i \in \mathcal{I}^C} |h_i|^2\right)} \right) \tag{7.244}$$

The bound is particularly simple in the case of binary signaling, where $\{a_k\} \in \{-1, 1\}$,

$$P_e < \exp\left(-\frac{\left(h_0 - \sum_{i \in \mathcal{I}} |h_i|\right)^2}{2\left(\sigma^2 + \sum_{i \in \mathcal{I}^C} |h_i|^2\right)} \right) \tag{7.245}$$

where $\mathcal{I}$ is such that $\sum_{i \in \mathcal{I}} |h_i| < h_0$. In this case it is rather simple to choose the set $\mathcal{I}$ so that the limit is tighter. We begin with $\mathcal{I} = \mathcal{Z}'$. Then we remove from $\mathcal{I}$ one by one the indices i that correspond to the larger values of $|h_i|$; we stop when the exponent of (7.245) has reached the minimum. Considering the limit of the function Q given by (6.364), we observe that for $\mathcal{I} = \mathcal{Z}'$ and $\mathcal{I}^C = \emptyset$, the bound in (7.244) practically coincides with the worst-case limit in (7.240). Taking instead $\mathcal{I} = \emptyset$ and $\mathcal{I}^C = \mathcal{Z}'$ we obtain again the limit given by the Gaussian approximation for z_k that yields (7.238).

For the mathematical details we refer to [10]; for a comparison between the Saltzberg bound and other bounds we refer to [5, 11].

GQR method

The GQR method is based on a technique for the approximate computation of integrals called *Gauss quadrature rule* (GQR). It offers a good compromise between computational complexity and approximation accuracy.

If we assume a very large number of interferers, to the limit infinite, $\mathtt{i}_k$ can be modelled as a continuous random variable. Then P_e assumes the expression

$$P_e = 2\left(1 - \frac{1}{M}\right)\int_{-\infty}^{+\infty} Q\left(\frac{h_0 - \xi}{\sigma}\right) p_{\mathtt{i}_k}(\xi)\, d\xi = 2\left(1 - \frac{1}{M}\right) I \tag{7.246}$$

By the GQR method we obtain an approximation of the integral, given by

$$I = \sum_{j=1}^{N_\mathrm{w}} \mathtt{w}_j\, Q\left(\frac{h_0 - \xi_j}{\sigma}\right) \tag{7.247}$$

In this expression the parameters $\{\xi_j\}$ and $\{\mathtt{w}_j\}$ are called, respectively, *abscissae* and *weights* of the quadrature rule, and are obtained by a numerical algorithm based on the first $2N_\mathrm{w}$ moments of $\mathtt{i}_k$. The quality of the approximation depends on the choice of N_w [5].

Appendix 7.C *Coherent PAM-DSB transmission*

General scheme

For transmission over a passband channel, a PAM signal must be suitably shifted in frequency by a sinusoidal carrier at frequency f_0. This task is achieved by DSB modulation (see Example 1.6.3 on page 41) of the signal $s(t)$ at the output of the baseband PAM modulator filter.

In the case of a coherent receiver, the passband scheme is given in Figure 7.52. For the baseband equivalent model, we refer to Figure 7.53a.

Now we consider the study of the PAM-DSB transmission system in the unified framework of Figure 7.12. Assuming the receive filter $\overline{g}_{Rc}$ real-valued, we apply the operator $Re\,[\ ]$ to the channel filter impulse response and to the noise signal, and we split the factor $1/2$ evenly among the channel filter and the receive filter responses; setting $g_{Rc}(t) = \overline{g}_{Rc}(t)\frac{1}{\sqrt{2}}$, we thus obtain the simplified scheme of Figure 7.53b, where the noise signal contains only the in-phase component $w_I'(t)$ with PSD

$$\mathcal{P}_{w_I'}(f) = \frac{N_0}{2}\ (\mathrm{V}^2/\mathrm{Hz}) \tag{7.248}$$

and

$$g_C(t) = Re\left[\frac{e^{-j(\varphi_1-\varphi_0)}g_{Ch}^{(bb)}(t)}{2\sqrt{2}}\right] \tag{7.249}$$

or, in the frequency domain,

$$\mathcal{G}_C(f) = \frac{e^{-j(\varphi_1-\varphi_0)}\mathcal{G}_{Ch}(f+f_0)1(f+f_0) + e^{j(\varphi_1-\varphi_0)}\mathcal{G}_{Ch}^*(-f+f_0)1(-f+f_0)}{4\sqrt{2}} \tag{7.250}$$

For a non-coherent receiver we refer to the scheme developed in Example 6.11.6 on page 516.

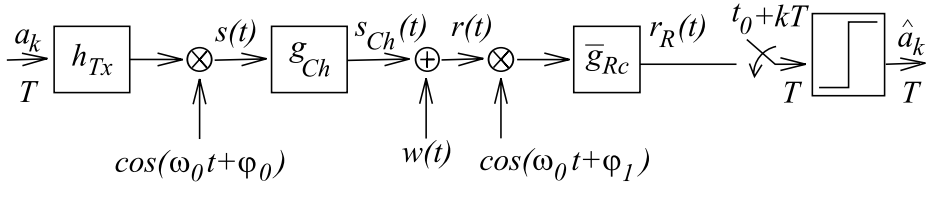
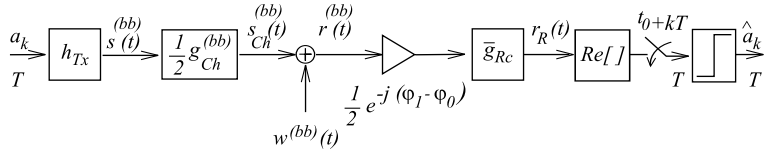
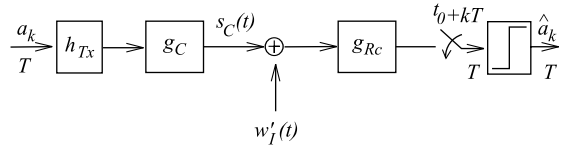


Figure 7.52. PAM-DSB passband transmission system.



(a) Baseband equivalent model



(b) Simplified model

Figure 7.53. PAM-DSB system.

Transmit signal PSD

Considering the PSD of the message sequence, the *average* PSD of the modulated signal $s(t)$ is given by (7.28),

$$\bar{\mathcal{P}}_s(f) = \frac{1}{4T^2}[\mathcal{P}_a(f - f_0)\,|\mathcal{H}_{Tx}(f - f_0)|^2 + \mathcal{P}_a(f + f_0)\,|\mathcal{H}_{Tx}(-f - f_0)|^2] \quad (7.251)$$

Consequently the transmitted signal bandwidth is equal to twice the bandwidth of h_{Tx}. The minimum bandwidth is given by

$$B_{min} = \frac{1}{T} \quad (7.252)$$

Recalling the definition (6.103), the spectral efficiency of the transmission system is given by

$$\nu = \log_2 M \ \text{(bit/s/Hz)} \quad (7.253)$$

which is halved with respect to M-PAM (see Table 6.9).

Signal-to-noise ratio

We assume the function

$$\frac{e^{-j(\varphi_1 - \varphi_0)} g_{Ch}^{(bb)}(t)}{2\sqrt{2}} \quad (7.254)$$

is real-valued; then from Figure 7.53a, using (1.295), we have the following relation:

$$E[|s_C(t)|^2] = \frac{E[|s_{Ch}^{(bb)}(t)|^2]}{2} = E[|s_{Ch}(t)|^2] \quad (7.255)$$

Setting

$$q_C(t) = h_{Tx} * g_C(t) \tag{7.256}$$

from (6.105) and (7.252) we have

$$\Gamma = \frac{\mathtt{M}_a E_{q_C}}{N_0} \tag{7.257}$$

where, for an M-PAM system (6.110),

$$\mathtt{M}_a = \frac{M^2 - 1}{3} \tag{7.258}$$

In the absence of ISI, for γ defined in (7.106), (7.107) still holds; moreover, using (7.257), for a matched filter receiver, (7.113) yields

$$\gamma_{MF} = \frac{E_{q_C}}{N_0/2} = \frac{2\Gamma}{\mathtt{M}_a} \tag{7.259}$$

Then the error probability is given by

$$P_e = 2\left(1 - \frac{1}{M}\right) Q\left(\sqrt{\frac{6\Gamma}{M^2 - 1}}\right) \tag{7.260}$$

We observe that the performance of an M-PAM-DSB system and that of an M-PAM system are the same, in terms of P_e as a function of the received power. However, because of DSB modulation, the required bandwidth is doubled with respect to both baseband PAM transmission and PAM-SSB modulation.[14] This explains the limited usage of PAM-DSB for digital transmission.

[14] The PAM-SSB scheme presents in practice considerable difficulties because the filter for modulation is non-ideal: in fact, this causes distortion of the signal $s(t)$ at low frequencies that may be compensated for only by resorting to line coding (see Appendix 7.A).

Appendix 7.D *Implementation of a QAM transmitter*

Three structures, which differ by the position of the digital-to-analog converter, may be considered for the implementation of a QAM transmitter. In Figure 7.54 the modulator employs for both in-phase and quadrature signals a DAC after the interpolator filter h_{Tx}, followed by an analog mixer that shifts the signal to passband. This scheme works if the sampling frequency $1/T_c$ is much greater than twice the bandwidth B of h_{Tx}.

For applications where the symbol rate is very high, the DAC is placed right after the bit mapper and the various filters are analog (see Chapter 19).

In the implementation illustrated in Figure 7.55, the DAC is placed instead at an intermediate stage with respect to the case of Figure 7.54. Samples are premodulated by a digital mixer to an intermediate frequency f_1, interpolated by the DAC and subsequently remodulated by a second analog mixer that shifts the signal to the desired band. The intermediate frequency f_1 must be greater than the bandwidth B and smaller than $1/(2T_c) - B$, thus avoiding overlap among spectral components. We observe that this scheme requires only one DAC, but the sampling frequency must be at least double as compared to the previous scheme.

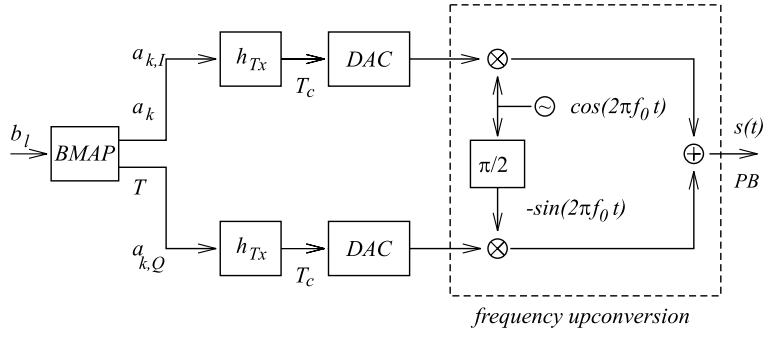


Figure 7.54. QAM with analog mixer.

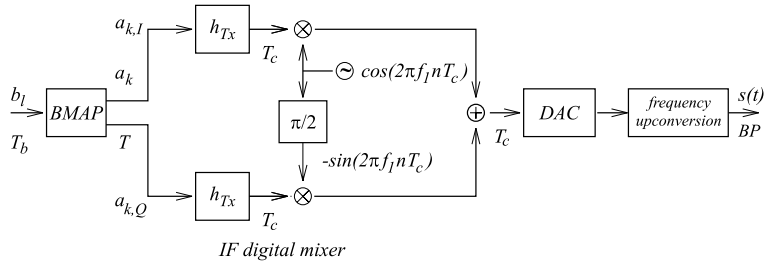


Figure 7.55. QAM with digital and analog mixers.

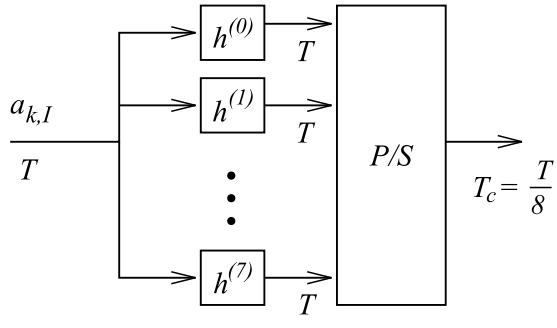


Figure 7.56. Polyphase implementation of the filter h_{Tx} for $T_c = T/8$.

For the first implementation, as the system is typically oversampled with a sampling interval $T_c = T/4$ or $T_c = T/8$, the frequency response of the DAC, $\mathcal{G}_I(f)$, may be considered as a constant in the passband of both the in-phase and quadrature signals. For the second implementation, unless $f_1 \ll 1/T_c$, the distortion introduced by the DAC should be considered and equalized by one of these methods (see page 338):

- including the compensation for $\mathcal{G}_I(f)$ in the frequency response of the filter h_{Tx},
- inserting a digital filter before the DAC,
- inserting an analog filter after the DAC.

We recall that an efficient implementation of interpolator filters h_{Tx} is obtained by the polyphase representation, as shown in Figure 7.56 for $T_c = T/8$, where

$$h^{(\ell)}(m) = h_{Tx}\left(mT + \ell\frac{T}{8}\right) \qquad \ell = 0, 1, \ldots, 7 \quad m = -\infty, \ldots, +\infty \tag{7.261}$$

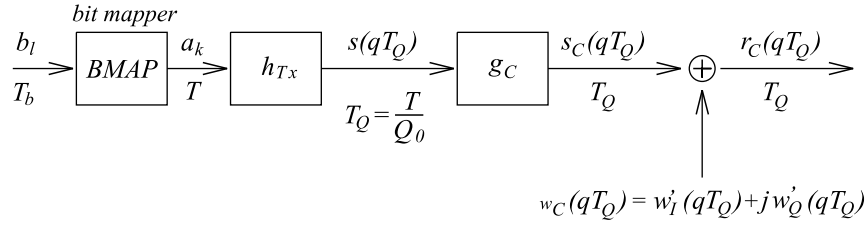
To implement the scheme of Figure 7.56, once the impulse response is known, it may be convenient to precompute the possible values of the filter output and store them in a table or RAM. The symbols $\{a_{k,I}\}$ are then used as pointers for the table itself. The same approach may be followed to generate the values of the signals $\cos(2\pi f_1 n T_c)$ and $\sin(2\pi f_1 n T_c)$ in Figure 7.55, using an additional table and the index n as a cyclic pointer.

Appendix 7.E *Simulation of a QAM system*

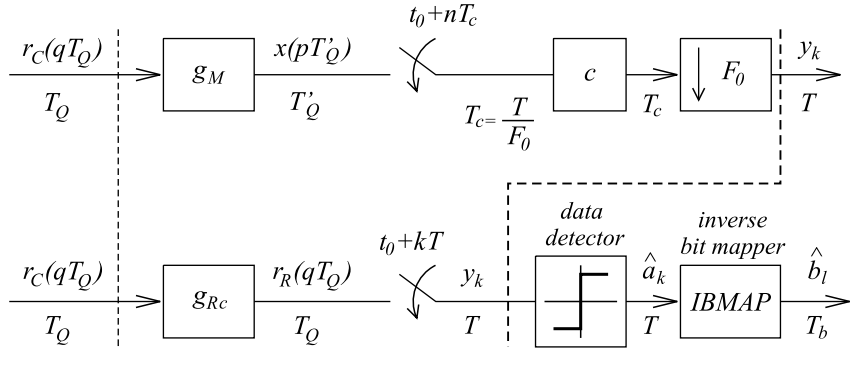
In Figure 7.12 we consider the baseband equivalent scheme of a QAM system. The aim is to simulate the various transformations in the discrete-time domain and to estimate the bit error probability.

This simulation method, also called Monte Carlo, is simple and general because it does not require any special assumption on the processes involved; however, it is intensive from the computational point of view. For alternative methods, for example semi-analytical, to estimate the error probability, we refer to specific texts on the subject [12].

We describe the various transformations in the overall discrete-time system depicted in Figure 7.57, where the only difference with respect to the scheme of Figure 7.12 is that the



(a) Transmitter and channel block diagram.



(b) Receiver block diagram.

Figure 7.57. Baseband equivalent model of a QAM system with discrete-time filters and sampling period $T_Q = T/Q_0$. At the receiver, in addition to the general scheme, a multirate structure to obtain samples of the received signal at the timing phase t_0 is also shown.

filters are discrete-time with quantum $T_Q = T/Q_0$, Which is chosen to accurately represent the various signals.

Binary sequence $\{b_\ell\}$. The sequence $\{b_\ell\}$ is generated as a random sequence or as a PN sequence (see Appendix 3.A), and has length K.

Bit mapper. The bit mapper maps patterns of information bits to symbols; the symbol constellation depends on the modulator (see Figure 7.6 for two constellations).

Interpolator filter h_{Tx} from period T to T_Q. The interpolator filter is efficiently implemented by using the polyphase representation (see Appendix 1.A). For a bandlimited pulse of the *raised cosine* or *square root raised cosine* type, the maximum value of T_Q, submultiple of T, is $T/2$. In any case, the implementation of filters, for example, the filter representing the channel, and non-linear transformations, for example, the transformation due to a power amplifier operating near saturation (not considered in Figure 7.57), typically require a larger bandwidth, leading, for example, to the choice $T_Q = T/4$ or $T/8$. In the following examples we choose $T_Q = T/4$.

For the design of h_{Tx} the window method can be used (N_h odd):

$$h_{Tx}(qT_Q) = h_{id}\left[\left(q - \frac{N_h - 1}{2}\right)T_Q\right]\mathtt{w}_{N_h}(q) \qquad q = 0, 1, \ldots, N_h - 1 \tag{7.262}$$

where typically $\mathtt{w}_{N_h}$ is the discrete-time rectangular window or the Hamming window, and h_{id} is the ideal impulse response.

Frequency responses of h_{Tx} are illustrated in Figure 7.58 for h_{id} *square root raised cosine* pulse with roll-off factor $\rho = 0.3$, and $\mathtt{w}_{N_h}$ rectangular window of length N_h, for various values of N_h ($T_Q = T/4$). The corresponding impulse responses are shown in Figure 7.59.

Transmission channel. For a radio channel the discrete-time model of Figure 4.35 can be used, where in the case of channel affected by fading, the coefficients of the FIR filter that model the channel impulse response are random variables with a given power delay profile. For a transmission line the discrete-time model of (4.150) can be adopted.

We assume the statistical power of the signal at output of the transmission channel is given by $\mathtt{M}_{s_{Ch}} = \mathtt{M}_{s_C}$.

Additive white Gaussian noise. Let $\bar{w}_I(qT_Q)$ and $\bar{w}_Q(qT_Q)$ be two Gaussian statistically independent r.v.s, each with zero mean and variance $1/2$, generated according to (1.655). To generate the complex-valued noise signal $\{w_C(qT_Q)\}$ with spectrum N_0, it is sufficient to use the relation

$$w_C(qT_Q) = \sigma_{w_C}[\bar{w}_I(qT_Q) + j\bar{w}_Q(qT_Q)] \tag{7.263}$$

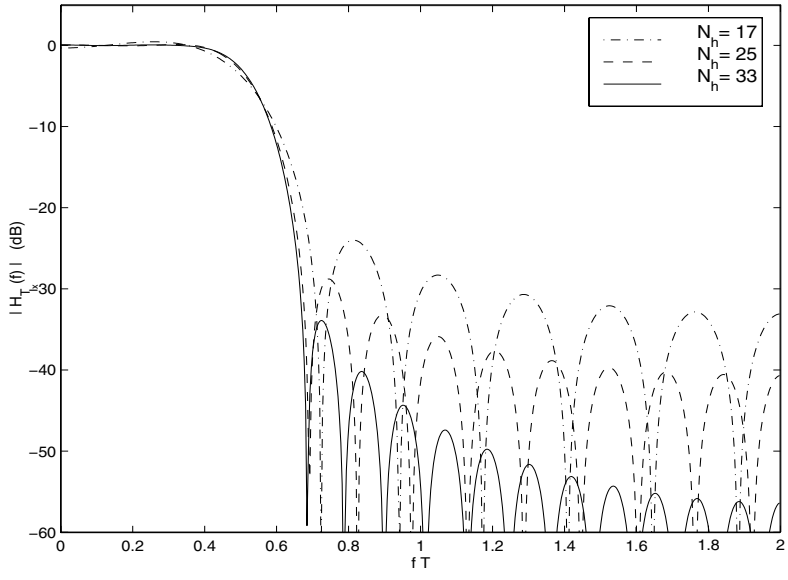


Figure 7.58. Magnitude of the transmit filter frequency response, for a windowed *square root raised cosine* pulse with roll-off factor $\rho = 0.3$, for three values of N_h ($T_Q = T/4$).

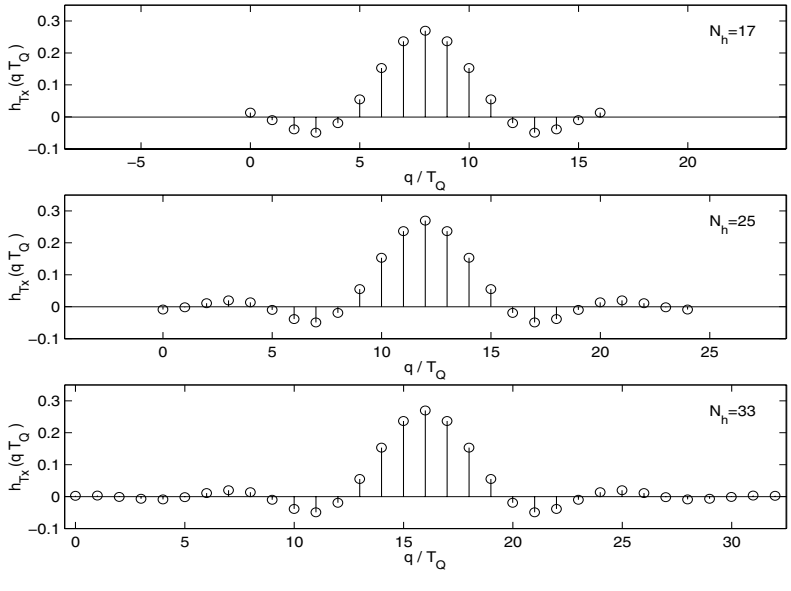


Figure 7.59. Transmit filter impulse response, $\{h_{Tx}(qT_Q)\}$, $q = 0, \ldots, N_{h-1}$, for a windowed *square root raised cosine* pulse with roll-off factor $\rho = 0.3$, for three values of N_h ($T_Q = T/4$).

where

$$\sigma_{w_C}^2 = N_0 \frac{1}{T_Q} \tag{7.264}$$

Usually the signal-to-noise ratio Γ given by (6.105) is given. For a QAM system, from (7.51) and (7.55) we have

$$\Gamma = \frac{\mathtt{M}_{s_C}}{N_0(1/T)} = \frac{\mathtt{M}_{s_C}}{\sigma_{w_C}^2 (T_Q/T)} \tag{7.265}$$

The standard deviation of the noise to be inserted in (7.263) is given by

$$\sigma_{w_C} = \sqrt{\frac{\mathtt{M}_{s_C} Q_0}{\Gamma}} \tag{7.266}$$

We note that σ_{w_C} is a function of $\mathtt{M}_{s_C}$, of the oversampling ratio $Q_0 = T/T_Q$, and of the given ratio Γ. In place of Γ, the ratio $E_b/N_0 = \Gamma/\log_2 M$ may be assigned.

Receive filter. As will be discussed in Chapter 8, there are several possible solutions for the receive filter. The most common choice is a matched filter g_M, matched to h_{Tx}, of the *square root raised cosine* type. Alternatively, the receive filter may be a simple *anti-aliasing FIR filter* g_{AA}, with passband at least equal to that of the desired signal. The filter attenuation in the stopband must be such that the statistical power of the noise evaluated in the passband is larger by a factor of 5–10 with respect to the power of the noise evaluated in the stopband, so that we can ignore the contribution of the noise in the stopband at the output of the filter g_{AA}.

If we adopt as bandwidth of g_{AA} the Nyquist frequency $1/(2T)$, the stopband of an ideal filter with unit gain goes from $1/(2T)$ to $1/(2T_Q)$: therefore the ripple δ_s in the stopband must satisfy the constraint

$$\frac{N_0 \frac{1}{2T}}{\delta_s N_0 \left(\frac{1}{2T_Q} - \frac{1}{2T} \right)} > 10 \tag{7.267}$$

from which we get the condition

$$\delta_s < \frac{10^{-1}}{Q_0 - 1} \tag{7.268}$$

Usually the presence of other interfering signals forces the selection of a value of δ_s that is smaller than that obtained in (7.268).

Interpolator filter. The interpolator filter is used to increase the sampling rate from $1/T_Q$ to $1/T_Q'$: this is useful when T_Q is insufficient to obtain the accuracy needed to represent the timing phase t_0. This filter can be part of g_M or g_{AA}. From Appendix 1.A, the efficient implementation of $\{g_M(pT_Q')\}$ is obtained by the polyphase representation with T_Q/T_Q' branches.

To improve the accuracy of the desired timing phase, further interpolation, for example, linear, may be employed.

Timing phase. Assuming a training sequence is available, for example, of the PN type $\{a_0 = p(0), a_1 = p(1), \ldots, a_{L-1} = p(L-1)\}$, a simple method to determine t_0 is to choose the timing phase in relation to of the peak of the overall impulse response. Let $\{x(pT'_Q)\}$ be the signal before downsampling. If we evaluate

$$\begin{aligned} m_{opt} &= \arg\max_m |\mathrm{r}_{xa}(mT'_Q)| \\ &= \arg\max_m \left| \frac{1}{L} \sum_{\ell=0}^{L-1} x(\ell T + mT'_Q) p^*(\ell) \right| \qquad m_{min}T'_Q < mT'_Q < m_{max}T'_Q \end{aligned} \tag{7.269}$$

then

$$t_0 = m_{opt} T'_Q \tag{7.270}$$

In (7.269) $m_{min}T'_Q$ and $m_{max}T'_Q$ are estimates of minimum and maximum system delay, respectively. Moreover, we note that the accuracy of t_0 is equal to T'_Q and that the amplitude of the desired signal is $h_0 = \mathrm{r}_{xa}(m_{opt}T'_Q)/\mathrm{r}_a(0)$.

Downsampler. The sampling period after downsampling is usually T or $T_c = T/2$, with timing phase t_0. The interpolator filter and the downsampler can be jointly implemented, according to the scheme of Figure 1.81. For example, for $T_Q = T/4$, $T'_Q = T/8$, and $T_c = T/2$ the polyphase representation of the interpolator filter with output $\{x(pT'_Q)\}$ requires two branches. Also the polyphase representation of the interpolator-decimator requires two branches.

Equalizer. After downsampling, the signal is usually input to an equalizer (LE, FSE or DFE, see Chapter 8). The output signal of the equalizer has always sampling period equal to T. As observed several times, to decimate simply means to evaluate the output at the desired instants.

Data detection. The simplest method resorts to a threshold detector, with thresholds determined by the constellation and the amplitude of the pulse at the decision point.

Viterbi algorithm. An alternative to the threshold detector, which operates on a symbol by symbol basis, is represented by maximum likelihood sequence detection by the Viterbi algorithm (see Chapter 8).

Inverse bit mapper. The inverse bit mapper performs the inverse function of the bit mapper. It translates the detected symbols into bits that represent the recovered information bits.

Simulations are typically used to estimate the bit error probability of the system, for a certain set of values of Γ. We recall that caution must be taken at the beginning and at the end of a simulation to consider transients of the system. Let $\bar{K}$ be the number of recovered bits. The estimate of the bit error probability P_{bit} is given by

$$\hat{P}_{bit} = \frac{\text{number of bits received with errors}}{\text{number of received bits, } \bar{K}} \tag{7.271}$$

It is known that as $\bar{K} \to \infty$, the estimate $\hat{P}_{bit}$ has a Gaussian probability distribution with mean P_{bit} and variance $P_{bit}(1 - P_{bit})/\bar{K}$. From this we can deduce, by varying $\bar{K}$, the confidence interval $[P_-, P_+]$ within which the estimate $\hat{P}_{bit}$ approximates P_{bit} with an assigned probabilty, that is

$$P[P_- \leq \hat{P}_{bit} \leq P_+] = P_{conf} \tag{7.272}$$

For example, we find that with $P_{bit} = 10^{-\ell}$ and $\bar{K} = 10^{\ell+1}$, we have a confidence interval of about a factor 2 with a probability of 95%, that is $P[1/2P_{bit} \leq \hat{P}_{bit} \leq 2P_{bit}] \simeq 0.95$. This is in good agreement with the experimental rule of selecting

$$\bar{K} = 3 \cdot 10^{\ell+1} \tag{7.273}$$

For a channel affected by fading, the average P_{bit} is not very significant: in this case it is meaningful to compute the distribution of P_{bit} for various channel realizations. In pratice we assume the transmission of a sequence of N_p packets, each one with $\bar{K}_p$ information bits to be recovered: typically $\bar{K}_p = 1000–10000$ bits and $N_p = 100–1000$ packets. Moreover, the channel realization changes at every packet. For a given average signal-to-noise ratio $\bar{\Gamma}$ (see (6.347)), the probability $\hat{P}_{bit}(n_p)$, $n_p = 1, \ldots, N_p$ is computed for each packet. As a performance measure we use the percentage of packets with $\hat{P}_{bit}(n_p) < P_{bit}$, also called bit error probability cumulative distribution function (cdf), where P_{bit} assumes values in a certain set.

This performance measure is more significant than the average P_{bit} evaluated for a very long, continuous transmission of $N_p\bar{K}_p$ information bits. In fact the average P_{bit} does not show that, in the presence of fading, the system may occasionally have a very large P_{bit}, and consequently an outage.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 8

Channel equalization and symbol detection

With reference to PAM and QAM systems, in this chapter we will discuss several methods to compensate for linear distortion introduced by the transmission channel. Next, as an alternative to a memoryless threshold detector, we will analyze detection methods that operate on sequences of samples.

Recalling the analysis of Section 7.3, we first review three techniques relying on the zero-forcing filter, linear equalizer, and DFE, respectively, that attempt to reduce the ISI in addition to maximizing the ratio γ defined in (7.106).

8.1 Zero-forcing equalizer (LE-ZF)

From the relation (7.66), assuming that $\mathcal{H}_{Tx}(f)$ and $\mathcal{G}_C(f)$ are known, and $\mathcal{H}(f)$ is given, for example, by (7.84), then the equation

$$\mathcal{H}(f) = \mathcal{Q}_R(f)e^{j2\pi f t_0} = \mathcal{H}_{Tx}(f)\mathcal{G}_C(f)\mathcal{G}_{Rc}(f)e^{j2\pi f t_0} \tag{8.1}$$

can be solved with respect to the receive filter, yielding

$$\mathcal{G}_{Rc}(f) = \frac{T\ \mathrm{rcos}\left(\dfrac{f}{1/T}, \rho\right)}{\mathcal{H}_{Tx}(f)\mathcal{G}_C(f)} e^{-j2\pi f t_0} \tag{8.2}$$

From (8.2), the magnitude and phase responses of $\mathcal{G}_{Rc}$ can be obtained. In practice, however, although the condition (8.2) leads to the suppression of the ISI, hence the filter g_{Rc} is called *linear equalizer zero-forcing* (LE-ZF), it may also lead to the enhancement of the noise power at the decision point, as expressed by (7.75).

In fact, if the frequency response $\mathcal{G}_C(f)$ exhibits strong attenuation at certain frequencies in the range $[-(1+\rho)/(2T), (1+\rho)/(2T)]$, then $\mathcal{G}_{Rc}(f)$ presents peaks that determine a large value of $\sigma^2_{w_R}$. In any event, the choice (8.2) guarantees the absence of ISI at the decision point, and from (7.109) we get

$$\gamma_{LE-ZF} = \frac{2}{N_0 E_{g_{Rc}}} \tag{8.3}$$

Obviously, based on the considerations of Section 7.3, it is

$$\gamma_{LE-ZF} \leq \gamma_{MF} \tag{8.4}$$

where γ_{MF} is defined in (7.113).

In the particular case of an ideal channel, that is $\mathcal{G}_{Ch}(f) = \mathcal{G}_0$ in the passband of the system, and assuming h_{Tx} is given by

$$\mathcal{H}_{Tx}(f) = T\sqrt{\text{rcos}\left(\frac{f}{1/T}, \rho\right)} \tag{8.5}$$

then from (7.42)

$$\mathcal{Q}_C(f) = \mathcal{H}_{Tx}(f)\,\mathcal{G}_C(f) = k_1\mathcal{H}_{Tx}(f) \tag{8.6}$$

where from (7.38), $k_1 = \mathcal{G}_0$ for a PAM system, whereas $k_1 = (\mathcal{G}_0/\sqrt{2})e^{-j(\varphi_1-\varphi_0)}$ for a QAM system. Moreover, from (8.2), neglecting a constant delay, i.e. for $t_0 = 0$, it results that

$$\mathcal{G}_{Rc}(f) = \frac{1}{k_1}\sqrt{\text{rcos}\left(\frac{f}{1/T}, \rho\right)} \tag{8.7}$$

In other words $g_{Rc}(t)$ is matched to $q_C(t) = k_1 h_{Tx}(t)$, and

$$\gamma_{LE-ZF} = \gamma_{MF} \tag{8.8}$$

Methods for the design of a LE-ZF filter with a finite number of coefficients are given in Section 8.7.

8.2 Linear equalizer (LE)

We introduce an optimization criterion for $\mathcal{G}_{Rc}$ that takes into account the ISI as well as the statistical power of the noise at the decision point.

8.2.1 Optimum receiver in the presence of noise and ISI

With reference to the scheme of Figure 7.12 for a QAM system, the criterion of choosing the receive filter such that the signal y_k is *as close as possible to* a_k *in the mean-square sense* is widely used.[1]

Let h_{Tx} and g_C be known. Defining the error

$$e_k = a_k - y_k \tag{8.9}$$

the receive filter g_{Rc} is chosen such that the mean-square error

$$J = E[|e_k|^2] = E[|a_k - y_k|^2] \tag{8.10}$$

is minimized.

[1] It would be desirable to find the filter such that $P[\hat{a}_k \neq a_k]$ is minimum. This problem, however, is usually very difficult to solve. Therefore we resort to the criterion of minimizing $E[|y_k - a_k|^2]$ instead.

The following assumptions are made:

1. the sequence $\{a_k\}$ is wide sense stationary (WSS) with spectral density $\mathcal{P}_a(f)$;
2. the noise w_C is complex-valued and WSS. In particular we assume it is white with spectral density $\mathcal{P}_{w_C}(f) = N_0$;
3. the sequence $\{a_k\}$ and the noise w_C are statistically independent.

The minimization of J in this situation differs from the classical problem of the optimum Wiener filter because h_{Tx} and g_C are continuous-time pulses. By resorting to the *calculus of variations* (see Appendix 8.A), we obtain the general solution

$$\mathcal{G}_{Rc}(f) = \frac{\mathcal{Q}_C^*(f)}{N_0} e^{-j2\pi f t_0} \frac{\mathcal{P}_a(f)}{T + \mathcal{P}_a(f) \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \frac{1}{N_0} \left| \mathcal{Q}_C\left(f - \frac{\ell}{T}\right) \right|^2} \tag{8.11}$$

where $\mathcal{Q}_C(f) = \mathcal{H}_{Tx}(f)\mathcal{G}_C(f)$.

Considerations on the joint optimization of the transmit and receive filters are discussed in Appendix 8.A.

If the symbols are *statistically independent* and have *zero mean*, then $\mathcal{P}_a(f) = T\sigma_a^2$, and (8.11) becomes:

$$\mathcal{G}_{Rc}(f) = \mathcal{Q}_C^*(f) e^{-j2\pi f t_0} \frac{\sigma_a^2}{N_0 + \sigma_a^2 \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \left| \mathcal{Q}_C\left(f - \frac{\ell}{T}\right) \right|^2} \tag{8.12}$$

The expression of the cost function J in correspondence of the optimum filter (8.12) is given in (8.40).

From the decomposition (7.62) of $\mathcal{G}_{Rc}(f)$, in (8.12) we have the following correspondences:

$$\mathcal{G}_M(f) = \mathcal{Q}_C^*(f) e^{-j2\pi f t_0} \tag{8.13}$$

and

$$C(e^{j2\pi fT}) = \frac{\sigma_a^2}{N_0 + \sigma_a^2 \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \left| \mathcal{Q}_C\left(f - \frac{\ell}{T}\right) \right|^2} \tag{8.14}$$

The optimum receiver thus assumes the structure of Figure 8.1. We note that g_M is the matched filter to the impulse response of the QAM system at the receiver input.[2]

The filter c is called *linear equalizer* (LE). It attempts to find the optimum trade-off between removing the ISI and enhancing the noise at the decision point.

[2] As derived later in the text (see Observation 8.13 on page 681) the output signal of the matched filter, sampled at the modulation rate $1/T$, forms a "sufficient statistic" if all the channel parameters are known.

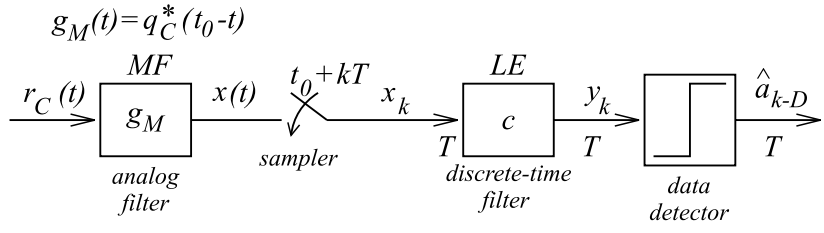


Figure 8.1. Optimum receiver structure for a channel with additive white noise.

We analyze two particular cases of the solution (8.12).

1. In the absence of noise, $w_C(t) \simeq 0$, and

$$C(e^{j2\pi fT}) = \frac{1}{\frac{1}{T}\sum_{\ell=-\infty}^{+\infty} \left| \mathcal{Q}_C\left(f - \frac{\ell}{T}\right)\right|^2} \tag{8.15}$$

   Note that the system is perfectly equalized, i.e. there is no ISI. In this case the filter (8.15) is the *linear equalizer zero-forcing*, as it completely eliminates the ISI.

2. In the absence of ISI at the output of g_M, that is if $|\mathcal{Q}_C(f)|^2$ is a Nyquist pulse, then $C(e^{j2\pi fT})$ is constant and the equalizer can be removed.

Alternative derivation of the IIR equalizer

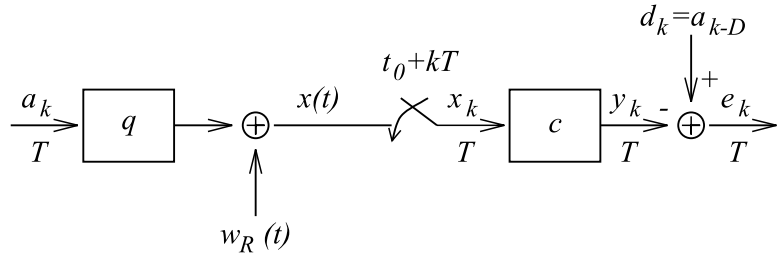
Starting from the receiver of Figure 8.1 and for *any type of filter* g_M, not necessarily matched, it is possible to determine the coefficients of the FIR equalizer filter c using the Wiener formulation, with the following definitions:

- filter input signal, x_k;
- filter output signal, y_k;
- desired output signal, $d_k = a_{k-D}$;
- estimation error, $e_k = d_k - y_k$.

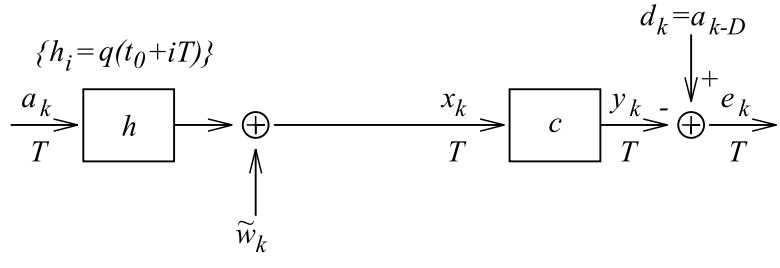
We notice the presence of the parameter D that denotes the lag of the desired signal: this parameter, which must be suitably estimated, expresses in number of symbol intervals the delay introduced by the equalizer. The overall delay from the emission of a_k to the generation of the detected symbol $\hat{a}_k$ is equal to $t_0 + DT$ seconds.

However, the particular case of a matched filter, for which $g_M(t) = q_C^*(-(t - t_0))$, is very interesting from a theoretical point of view. We assume that the filter c may have an infinite number of coefficients, i.e. it may be IIR. With reference to the scheme of Figure 8.2a, q is the overall impulse response of the system at the sampler input:

$$q(t) = h_{Tx} * g_C * g_M(t) = q_C * g_M(t) = \mathrm{r}_{q_C}(t - t_0) \tag{8.16}$$



(a) Continuous-time model.



(b) Discrete-time equivalent.

Figure 8.2. Linear equalizer as a Wiener filter.

where $\mathtt{r}_{q_C}$ is the autocorrelation of the deterministic pulse q_C, given by

$$\mathtt{r}_{q_C}(t) = [q_C(t') * q_C^*(-t')](t) \tag{8.17}$$

The Fourier transform of $\mathtt{r}_{q_C}(t)$ is given by

$$\mathcal{P}_{q_C}(f) = |\mathcal{Q}_C(f)|^2 \tag{8.18}$$

We note that if q_C has a finite support $(0, t_{q_C})$, then $g_M(t) = q_C^*(t_0 - t)$ has support $(t_0 - t_{q_C}, t_0)$. Hence, to obtain a causal filter g_M the minimum value of t_0 is t_{q_C}. In any case from (8.16), as $\mathtt{r}_{q_C}$ is a correlation function, the desired sample $q(t_0)$ is taken in relation to the maximum value of $|q(t)|$.

In Figure 8.2a, assuming w_C is white noise, we have

$$w_R(t) = w_C * g_M(t) \tag{8.19}$$

with autocorrelation function given by

$$\begin{aligned} \mathtt{r}_{w_R}(\tau) &= \mathtt{r}_{w_C} * \mathtt{r}_{q_C}(\tau) \\ &= N_0 \mathtt{r}_{q_C}(\tau) \end{aligned} \tag{8.20}$$

Then the spectrum of w_R is given by:

$$\mathcal{P}_{w_R}(f) = N_0 \mathcal{P}_{q_C}(f) = N_0 |\mathcal{Q}_C(f)|^2 \tag{8.21}$$

In Figure 8.2a, sampling at instants $t_k = t_0 + kT$ yields the sampled QAM signal

$$x_k = \sum_{i=-\infty}^{+\infty} a_i h_{k-i} + \tilde{w}_k \tag{8.22}$$

The discrete-time equivalent model is illustrated in Figure 8.2b. The discrete-time overall impulse response is given by

$$h_n = q(t_0 + nT) = \mathtt{r}_{q_C}(nT) \tag{8.23}$$

In particular, it results in

$$h_0 = \mathtt{r}_{q_C}(0) = E_{q_C} \tag{8.24}$$

The sequence $\{h_n\}$ has z-transform given by

$$\Phi(z) = \mathcal{Z}[h_n] = P_{q_C}(z) \tag{8.25}$$

which, by the Hermitian symmetry of an autocorrelation sequence, $\mathtt{r}_{q_C}(nT) = \mathtt{r}^*_{q_C}(-nT)$, satisfies the relation:

$$\Phi(z) = \Phi^* \left(\frac{1}{z^*} \right) \tag{8.26}$$

On the other hand, from (1.90), the Fourier transform of (8.23) is given by

$$\Phi(e^{j2\pi fT}) = \mathcal{F}[h_n] = \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \left| \mathcal{Q}_C \left(f - \frac{\ell}{T} \right) \right|^2 \tag{8.27}$$

Moreover, using the properties of Table 1.3, the correlation sequence of $\{h_n\}$ has z-transform equal to

$$\mathcal{Z}[\mathtt{r}_h(m)] = \Phi(z) \Phi^* \left(\frac{1}{z^*} \right) \tag{8.28}$$

Also, from (8.20), the z-transform of the autocorrelation of the noise samples $\tilde{w}_k = w_R(t_0 + kT)$ is given by:

$$\mathcal{Z}[\mathtt{r}_{\tilde{w}}(n)] = \mathcal{Z}[\mathtt{r}_{w_R}(nT)] = N_0 \Phi(z) \tag{8.29}$$

The Wiener solution that gives the optimum coefficients is given in the z-transform domain by (2.50):

$$C_{opt}(z) = \mathcal{Z}[c_n] = \frac{P_{dx}(z)}{P_x(z)} \tag{8.30}$$

where

$$P_{dx}(z) = \mathcal{Z}[\mathtt{r}_{dx}(n)] \qquad \text{and} \quad P_x(z) = \mathcal{Z}[\mathtt{r}_x(n)] \tag{8.31}$$

We assume the following assumptions hold:

1. The sequence $\{a_k\}$ is WSS, with symbols that are statistically independent and with zero mean,

$$\mathtt{r}_a(n) = \sigma_a^2 \delta_n \qquad \text{and} \quad \mathcal{P}_a(f) = T\sigma_a^2 \tag{8.32}$$

2. $\{a_k\}$ and $\{\tilde{w}_k\}$ are statistically independent and hence uncorrelated.

Then the cross-correlation between $\{d_k\}$ and $\{x_k\}$ is given by

$$\begin{aligned} \mathtt{r}_{dx}(n) &= E[d_k x_{k-n}^*] \\ &= E\left[a_{k-D}\left(\sum_{i=-\infty}^{+\infty} a_i h_{k-n-i} + \tilde{w}_{k-n}\right)^*\right] \\ &= \sum_{i=-\infty}^{+\infty} h_{k-n-i}^* E[a_{k-D} a_i^*] \end{aligned} \tag{8.33}$$

using assumption 2. Finally, from assumption 1,

$$\mathtt{r}_{dx}(n) = \sigma_a^2 h_{D-n}^* \tag{8.34}$$

Under the same assumptions 1 and 2, the computation of the autocorrelation of the process $\{x_k\}$ yields (see also Table 1.6):

$$\mathtt{r}_x(n) = E[x_k x_{k-n}^*] = \sigma_a^2 \mathtt{r}_h(n) + \mathtt{r}_{\tilde{w}}(n) \tag{8.35}$$

Thus, using (8.28), we obtain

$$\begin{aligned} P_{dx}(z) &= \sigma_a^2 \Phi^*\left(\frac{1}{z^*}\right) z^{-D} \\ P_x(z) &= \sigma_a^2 \Phi(z) \Phi^*\left(\frac{1}{z^*}\right) + N_0 \Phi(z) \end{aligned} \tag{8.36}$$

Therefore, from (8.30),

$$C_{opt}(z) = \frac{\sigma_a^2 \Phi^*\left(\frac{1}{z^*}\right) z^{-D}}{\sigma_a^2 \Phi(z) \Phi^*\left(\frac{1}{z^*}\right) + N_0 \Phi(z)} \tag{8.37}$$

Taking into account the property (8.26), (8.37) is simplified as

$$C_{opt}(z) = \frac{\sigma_a^2 z^{-D}}{N_0 + \sigma_a^2 \Phi(z)} \tag{8.38}$$

It can be observed that, for $z = e^{j2\pi fT}$, (8.38) corresponds to (8.14), apart from the term z^{-D}, which accounts for a possible delay introduced by the equalizer.

In relation to the *optimum filter* $C_{opt}(z)$, we determine the minimum value of the cost function. We recall the general expression for the Wiener filter (2.53):

$$\begin{aligned} J_{min} &= \sigma_d^2 - \sum_{i=0}^{N-1} c_{opt,i}\, r_{dx}^*(i) \\ &= \sigma_d^2 - \int_{-\frac{1}{2T}}^{\frac{1}{2T}} \mathcal{P}_{dx}^*(f)\, C_{opt}(e^{j2\pi fT})\, df \end{aligned} \tag{8.39}$$

Finally, substitution of the relations (8.36) in (8.39) yields

$$\begin{aligned} J_{min} &= \sigma_d^2 - T \int_{-\frac{1}{2T}}^{\frac{1}{2T}} P_{dx}^*(e^{j2\pi fT}) C_{opt}(e^{j2\pi fT})\, df \\ &= \sigma_a^2 T \int_{-\frac{1}{2T}}^{\frac{1}{2T}} \frac{N_0}{N_0 + \sigma_a^2 \Phi(e^{j2\pi fT})}\, df \end{aligned} \tag{8.40}$$

If $\Phi(z)$ is a rational function of z, the integral (8.40) may be computed by evaluating the coefficient of the term z^0 of the function $\sigma_a^2 N_0/(N_0 + \sigma_a^2 \Phi(z))$, which can be obtained by series expansion of the integrand, or by using the partial fraction expansion method (see (1.131)).

We note that in the absence of ISI, at the output of the MF we get $\Phi(z) = h_0 = E_{qC}$, and

$$J_{min} = \frac{\sigma_a^2 N_0}{N_0 + \sigma_a^2 E_{qC}} \tag{8.41}$$

Signal-to-noise ratio γ

We define the overall impulse response at the equalizer output, sampled with a sampling rate equal to the modulation rate $1/T$, as

$$\psi_i = (h_n * c_{opt,n})_i \tag{8.42}$$

where $\{h_n\}$ is given by (8.23) and $c_{opt,n}$ is the impulse response of the optimum filter (8.38).

At the decision point we have

$$y_k = \psi_D a_{k-D} + \sum_{\substack{i=-\infty \\ i \neq D}}^{+\infty} \psi_i a_{k-D-i} + (\widetilde{w}_n * c_{opt,n})_k \tag{8.43}$$

We assume that in (8.43) the total disturbance given by ISI plus noise is modeled as Gaussian noise with variance $2\sigma_I^2$. Hence, for a minimum distance among symbols of the constellation equal to 2, (7.106) yields

$$\gamma_{LE} = \left(\frac{\psi_D}{\sigma_I} \right)^2 \tag{8.44}$$

In case the approximation $\psi_D \simeq 1$ holds, the total disturbance in (8.43) coincides with e_k, hence $2\sigma_I^2 \simeq J_{min}$, and (8.44) becomes

$$\gamma_{LE} \simeq \frac{2}{J_{min}} \tag{8.45}$$

where J_{min} is given by (8.40).

8.3 LE with a finite number of coefficients

In practice, if the channel is either unknown *a priori* or it is time variant, it is necessary to design a receiver that tries to identify the channel characteristics and at the same time to equalize it through suitable adaptive algorithms.

Two alternative approaches are usually considered.

First solution. The classical block diagram of an adaptive receiver is shown in Figure 8.3.

The *matched filter* g_M is designed assuming an ideal channel. Therefore the equalization task is left to the filter c; otherwise, if it is possible to rely on some *a priori* knowledge of the channel, the filter g_M may be designed according to the average characteristics of the channel. The filter c is then an adaptive transversal filter that attempts, in real time, to equalize the channel by adapting its coefficients to the channel variations.

Second solution. The receiver is represented in Figure 8.4.

The *anti-aliasing* filter g_{AA} is designed according to specifications imposed by the sampling theorem. In particular if the desired signal s_C has a bandwidth B and x is sampled with period $T_c = T/F_0$, where F_0 is the *oversampling index*, with $F_0 \geq 2$, then the passband of g_{AA} should extend at least up to frequency B. Moreover, because the noise w_C

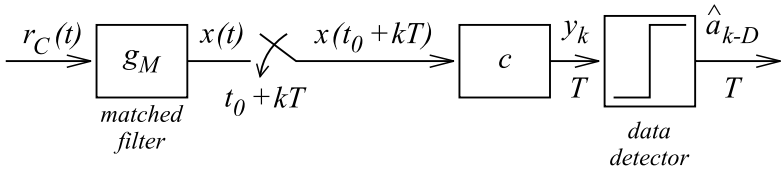


Figure 8.3. Receiver implementation by an analog matched filter followed by a sampler and a discrete-time linear equalizer.

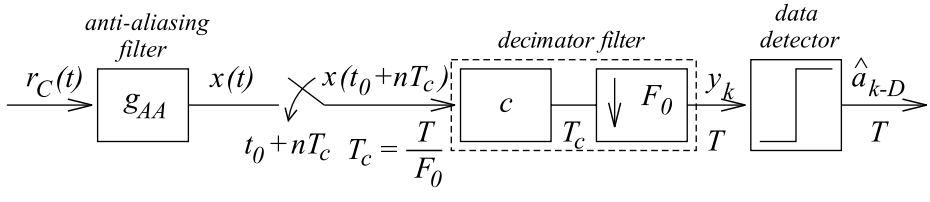


Figure 8.4. Receiver implementation by discrete-time filters.

is considered as a wideband signal, g_{AA} should also attenuate the noise components outside the passband of the desired signal s_C, hence the cut-off frequency of g_{AA} is between B and $F_0/(2T)$. In practice, to simplify the implementation of the filter g_{AA}, it is convenient to consider a wide transition band.

Thus the discrete-time filter c needs to accomplish the following tasks:

1. to filter the residual noise outside the passband of the desired signal s_C;
2. to act as a matched filter;
3. to equalize the channel.

Note that the filter c of Figure 8.4 is implemented as a decimator filter (see Appendix 1.A), where the input signal $x_n = x(t_0 + nT_c)$ is defined over a discrete-time domain with period $T_c = T/F_0$, and the output signal y_k is defined over a discrete-time domain with period T.

In turn, two strategies may be used to determine an equalizer filter c with N coefficients:

1. the *direct method*, which employs the Wiener formulation and requires the computation of the matrix $\mathbf{R}$ and the vector $\mathbf{p}$. The description of the direct method is postponed to Section 8.5 (see Observation 8.2 on page 641);
2. the *adaptive method*, which we will describe next (see Chapter 3).

Adaptive LE

We analyze now the solution illustrated in Figure 8.3: the discrete-time equivalent scheme is illustrated in Figure 8.5, where $\{h_n\}$ is the discrete-time impulse response of the overall system, given by

$$h_n = q(t_0 + nT) = h_{Tx} * g_C * g_M(t)|_{t=t_0+nT} \tag{8.46}$$

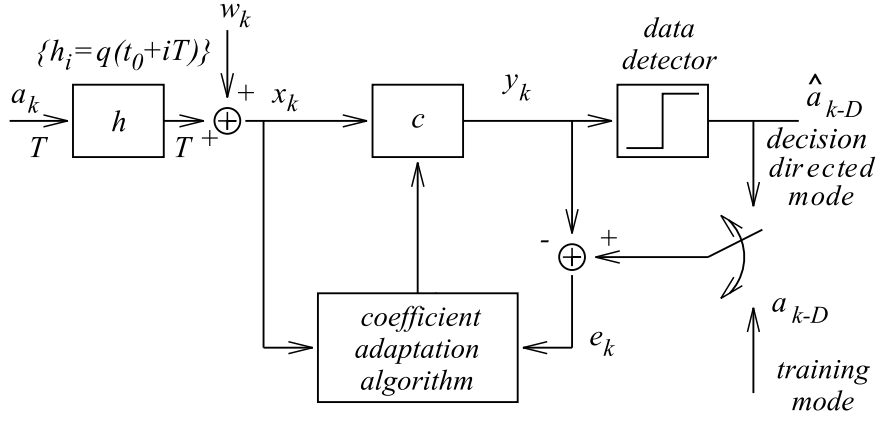


Figure 8.5. Discrete-time equivalent scheme associated with the implementation of Figure 8.3.

and

$$\tilde{w}_k = w_R(t_0 + kT) \tag{8.47}$$

with $w_R(t) = w_C * g_M(t)$.

The design strategy consists of the following steps.

1. Define the performance measure of the system. The MSE criterion is typically adopted:

$$J(k) = E[|e_k|^2] \tag{8.48}$$

2. Select the law of coefficient update. For example, for an FIR filter c with N coefficients using the LMS algorithm (see Section 3.1.2) we have

$$\mathbf{c}_{k+1} = \mathbf{c}_k + \mu e_k \mathbf{x}_k^* \tag{8.49}$$

where

a) *input vector*

$$\mathbf{x}_k^T = [x_k, x_{k-1}, \ldots, x_{k-N+1}] \tag{8.50}$$

b) *coefficient vector*

$$\mathbf{c}_k^T = [c_{0,k}, c_{1,k}, \ldots, c_{N-1,k}] \tag{8.51}$$

c) *adaptation gain*

$$0 < \mu < \frac{2}{N r_x(0)} \tag{8.52}$$

3. To evaluate the signal error e_k to be used in the adaptive algorithm we distinguish two modes.

a) *Training mode*

$$e_k = a_{k-D} - y_k \qquad k = D, \ldots, L_{TS} + D - 1 \tag{8.53}$$

Evaluation of the error in training mode is possible if a sufficiently long sequence of L_{TS} symbols known at the receiver, called *training sequence* (TS), $\{a_k\}$, $k = 0, 1, \ldots, L_{TS} - 1$, is transmitted. The duration of the transmission of TS is equal to $L_{TS}T$. During this time interval, the automatic identification of the channel characteristics takes place, allowing the computation of the optimum coefficients of the equalizer filter c, and consequently channel equalization. As the spectrum of the training sequence must be wide, typically a PN sequence is used (see Appendix 3.A). We note that even the direct method requires a training sequence to determine the vector $\mathbf{p}$ and the matrix $\mathbf{R}$ (see Observation 8.3 on page 641).

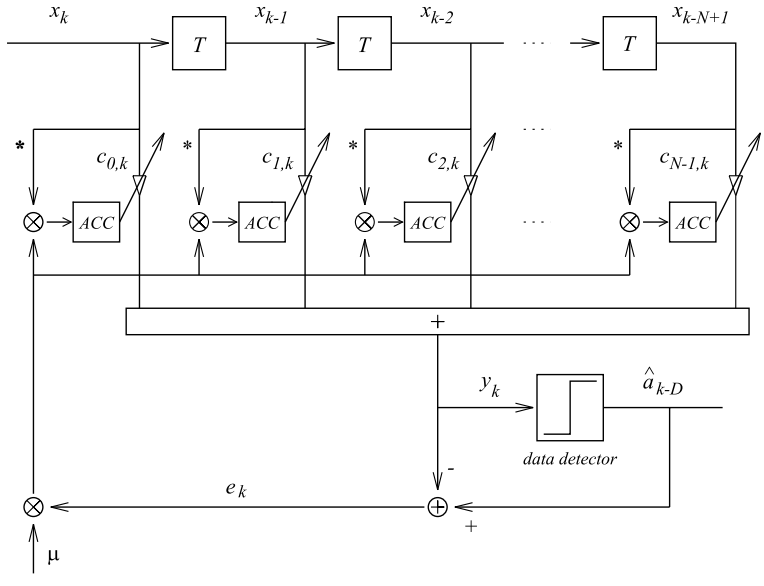


Figure 8.6. Linear adaptive equalizer implemented as a transversal filter with N coefficients.

b) *Decision directed mode*

$$e_k = \hat{a}_{k-D} - y_k \qquad k \geq L_{TS} + D \tag{8.54}$$

Once the transmission of the TS is completed, we assume that the equalizer has arrived at convergence. Therefore $\hat{a}_k \simeq a_k$, and the transmission of information symbols may start. In (8.53) we then substitute the known transmitted symbol with the detected symbol to obtain (8.54).

The implementation of the above equations is illustrated in Figure 8.6.

8.4 Fractionally spaced equalizer (FSE)

We consider the receiver structure with oversampling illustrated in Figure 8.4. The discrete-time overall system, shown in Figure 8.7, has impulse response given by

$$h_i = q(t_0 + iT_c) \tag{8.55}$$

where

$$q(t) = h_{Tx} * g_C * g_{AA}(t) \tag{8.56}$$

The noise is given by

$$\tilde{w}_n = w_R(t_0 + nT_c) = w_C * g_{AA}(t)|_{t=t_0+nT_c} \tag{8.57}$$

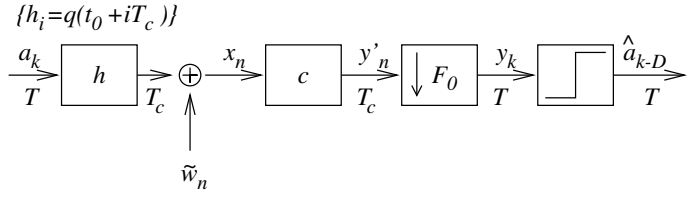


Figure 8.7. Fractionally spaced linear equalizer (FSE).

For the analysis, the filter c is decomposed into a discrete-time filter with sampling period T_c that is cascaded with a downsampler.

The input signal to the filter c is the sequence $\{x_n\}$ with sampling period $T_c = T/F_0$; the n-th sample of the sequence is given by

$$x_n = \sum_{k=-\infty}^{+\infty} h_{n-kF_0} a_k + \tilde{w}_n \tag{8.58}$$

The output of the filter c is given by

$$y'_n = \sum_{i=0}^{N-1} c_i x_{n-i} \tag{8.59}$$

We note that the overall impulse response at the filter output, defined on the discrete-time domain with sampling period T_c, is given by

$$\psi_i = h * c_i \tag{8.60}$$

If we denote by $\{y'_n\}$ the sequence of samples at the filter output, and by $\{y_k\}$ the downsampled sequence, we have

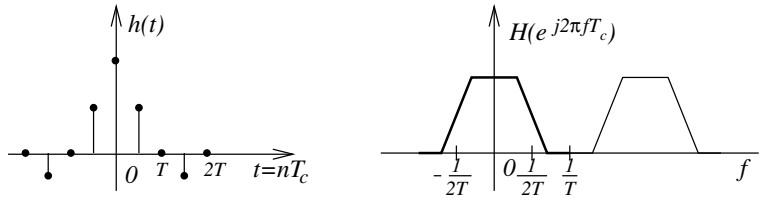
$$y_k = y'_{kF_0} \tag{8.61}$$

As mentioned earlier, in a practical implementation of the filter the sequence $\{y'_n\}$ is not explicitly generated; only the sequence $\{y_k\}$ is produced (see Appendix 1.A). However, introducing the downsampler helps to illustrate the advantages of operating with an oversampling index $F_0 > 1$.

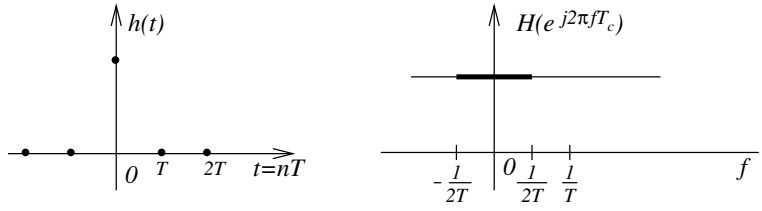
Before analyzing this system, we recall the Nyquist problem. Let us consider a QAM system with pulse $h(t)$:

$$s_R(t) = \sum_{n=-\infty}^{+\infty} a_n h(t - nT) \qquad t \in \Re \tag{8.62}$$

In Section 7.3 we considered continuous-time Nyquist pulses $h(t), t \in \Re$. Let $h(t)$ be defined now on a discrete-time domain $\{nT_c\}$, n integer, where $T_c = T/F_0$. If F_0 is an integer, the discrete-time pulse satisfies the Nyquist criterion if $h(0) \neq 0$, and $h(\ell F_0 T_c) = 0$, for all integers $\ell \neq 0$. In the particular case $F_0 = 1$ we have $T_c = T$, and the Nyquist conditions



(a) $F_0 = 2$.



(b) $F_0 = 1$.

Figure 8.8. Discrete-time Nyquist pulses and relative Fourier transforms.

impose that $h(0) \neq 0$, and $h(nT) = 0$, for $n \neq 0$. Recalling the input–output downsampler relations in the frequency domain, it is easy to deduce the behavior of a discrete-time Nyquist pulse in the frequency domain: two examples are given in Figure 8.8, for $F_0 = 2$ and $F_0 = 1$. We note that, for $F_0 = 1$, in the frequency domain a discrete-time Nyquist pulse is equal to a constant.

With reference to the scheme of Figure 8.7, the QAM pulse defined on the discrete-time domain with period T_c is given by (8.55), where $q(t)$ is defined in (8.56). From (8.60), using (8.55), the pulse $\{\psi_i\}$ at the equalizer output before the downsampler has the following Fourier transform:

$$\begin{aligned}\Psi(e^{j2\pi f T_c}) &= C(e^{j2\pi f T_c}) H(e^{j2\pi f T_c}) \\ &= C(e^{j2\pi f T_c}) \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \mathcal{Q}\left(f - \ell \frac{F_0}{T}\right) e^{j2\pi\left(f-\ell\frac{F_0}{T}\right)t_0}\end{aligned} \tag{8.63}$$

The task of the equalizer is to yield a pulse $\{\psi_i\}$ that approximates a Nyquist pulse, i.e. a pulse of the type shown in Figure 8.8. We see that choosing $F_0 = 1$, i.e. sampling the equalizer input signal with period equal to T, it may happen that $H(e^{j2\pi f T_c})$ assumes very small values at frequencies near $f = 1/(2T)$, because of an *incorrect choice of the timing phase* t_0. In fact, let us assume $q(t)$ is real with a bandwidth smaller than $1/T$. Using the polar notation for $\mathcal{Q}(\frac{1}{2T})$ we have

$$\mathcal{Q}\left(\frac{1}{2T}\right) = Ae^{j\varphi} \qquad \text{and} \quad \mathcal{Q}\left(-\frac{1}{2T}\right) = Ae^{-j\varphi} \tag{8.64}$$

as $\mathcal{Q}(f)$ is Hermitian. Therefore, from (8.63),

$$\begin{aligned} H(e^{j2\pi fT})\Big|_{f=\frac{1}{2T}} &= \mathcal{Q}\left(\frac{1}{2T}\right)e^{j2\pi\frac{t_0}{2T}} + \mathcal{Q}\left(\frac{1}{2T}-\frac{1}{T}\right)e^{j2\pi\left(\frac{1}{2T}-\frac{1}{T}\right)t_0} \\ &= 2A\cos\left(\varphi+\pi\frac{t_0}{T}\right) \end{aligned} \tag{8.65}$$

If t_0 is such that

$$\varphi+\pi\frac{t_0}{T}=\frac{2i+1}{2}\pi \qquad i \text{ integer} \tag{8.66}$$

then

$$H(e^{j2\pi\frac{1}{2T}T})=0 \tag{8.67}$$

In this situation the equalizer will enhance the noise around $f=1/(2T)$, or converge with difficulty in the adaptive case. If $F_0\geq 2$ is chosen, this problem is avoided because *aliasing* between replicas of $\mathcal{Q}(f)$ does not occur. Therefore the choice of t_0 may be less accurate. In fact, as we will see in Chapter 14, if c has an input signal sampled with sampling period $T/2$ it also acts as an interpolator filter, whose output can be used to determine the optimum timing phase.

In conclusion, the FSE receiver presents two advantages over T-spaced equalizers:

1. It is an optimum structure according to the MSE criterion, in the sense that it carries out the task of both matched filter (better rejection of the noise) and equalizer (reduction of ISI).

2. It is less sensible to the choice of t_0. In fact, the correlation method (7.269) with accuracy $T'_Q=T/2$ is usually sufficient to determine the timing phase.

Adaptive FSE

The direct method to compute the coefficients of a FSE is described in Section 8.5 (see Observation 8.7 on page 644); we consider now the adaptive method as depicted in Figure 8.9.

The choice of the oversampling index $F_0=2$ is very common. For this choice, the input samples of the filter c have sampling period $T/2$, and the output samples have sampling period T. Note that coefficient update takes place every T seconds. With respect to the basic scheme of Figure 8.7, in a practical implementation the equalizer output is not generated at every sampling instant multiple of $T/2$, but only at alternate sampling instants. The LMS adaptation equation is given by:

$$\mathbf{c}_{k+1}=\mathbf{c}_k+\mu e_k\mathbf{x}^*_{2k} \tag{8.68}$$

The adaptive FSE may incur a difficulty in the presence of noise with variance that is small with respect to the level of the desired signal: in this case some eigenvalues of the autocorrelation matrix of $\mathbf{x}^*_{2k}$ may assume a value that is almost zero and consequently

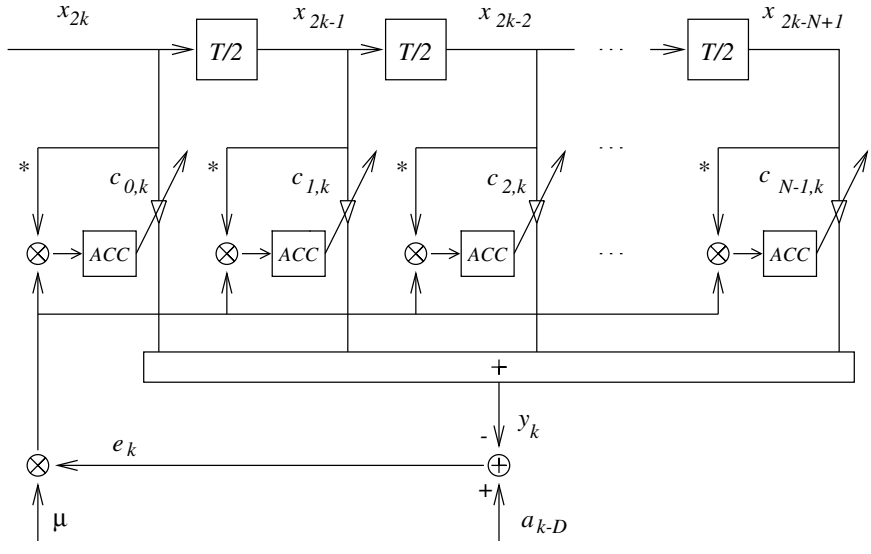


Figure 8.9. Adaptive FSE ($F_0 = 2$).

the problem of finding the optimum coefficients become ill-conditioned, with numerous solutions that present the same minimum value of the cost function. This effect can be illustrated also in the frequency domain: outside the passband of the input signal the filter c may assume arbitrary values, in the limit case of absence of noise. As a result, the coefficients of the filter c may vary in time and also assume very large values.

To mitigate this problem, recalling the leaky LMS algorithm (see page 187), we consider two methods that slightly modify the cost function. In both cases we attempt to impose a constraint on the amplitude that the coefficients may assume.

1. The leaky LMS algorithm. Let

$$J_1 = J + \alpha\, E\left[\sum_{i=0}^{N-1} |c_i|^2\right] \tag{8.69}$$

then

$$\begin{aligned} \mathbf{c}_{k+1} &= \mathbf{c}_k + \mu(e_k \mathbf{x}_{2k}^* - \alpha \mathbf{c}_k) \\ &= (1 - \mu\alpha)\mathbf{c}_k + \mu e_k \mathbf{x}_{2k}^* \end{aligned} \tag{8.70}$$

with $0 < \alpha \ll \mathtt{r}_x(0)$.

2. Let

$$J_2 = J + \alpha\, E\left[\sum_{i=0}^{N-1} |c_i|\right] \tag{8.71}$$

then

$$\begin{aligned}\mathbf{c}_{k+1} &= \mathbf{c}_k + \mu(e_k \mathbf{x}_{2k}^* - \alpha \operatorname{sgn} \mathbf{c}_k) \\ &= \mathbf{c}_k - \mu\alpha \operatorname{sgn} \mathbf{c}_k + \mu e_k \mathbf{x}_{2k}^*\end{aligned} \tag{8.72}$$

For an analysis of the performance of the FSE, including convergence properties, we refer to [1]. Simulations results also demonstrate better performance of the FSE with respect to an equalizer that operates at the symbol rate [2].

8.5 Decision feedback equalizer (DFE)

We consider the sampled signal at the output of the analog receive filter (see Figure 8.5 or Figure 8.7). For example, in the scheme of Figure 8.5, the desired signal is given by:

$$s_k = s_R(t_0 + kT) = \sum_{i=-\infty}^{+\infty} a_i h_{k-i} \tag{8.73}$$

where the sampled pulse $\{h_n\}$ is defined in (8.46). In the presence of noise we have

$$x_k = s_k + \tilde{w}_k \tag{8.74}$$

where $\tilde{w}_k$ is the noise, given by (8.47).

We assume, as illustrated in Figure 8.10, that $\{h_n\}$ has finite duration and support $[-N_1, -N_1+1, \ldots, N_2-1, N_2]$. The samples with positive time indices are called *postcursors*, and those with negative time indices *precursors*. Explicitly writing terms that include precursors and postcursors, (8.74) becomes:

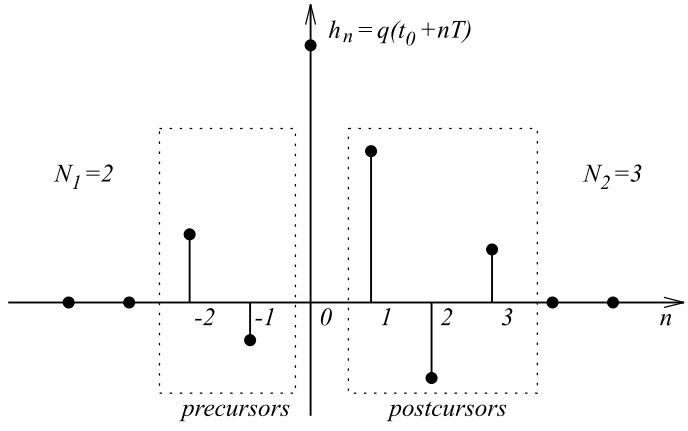
$$x_k = h_0 a_k + (h_{-N_1} a_{k+N_1} + \cdots + h_{-1} a_{k+1}) + (h_1 a_{k-1} + \cdots + h_{N_2} a_{k-N_2}) + \tilde{w}_k \tag{8.75}$$

In addition to the *actual symbol* a_k that we desire to detect from the observation of x_k, in (8.75) two terms are identified in parentheses: one that depends only on *past symbols* $a_{k-1}, \ldots, a_{k-N_2}$, and another that depends only on *future symbols*, $a_{k+1}, \ldots, a_{k+N_1}$.

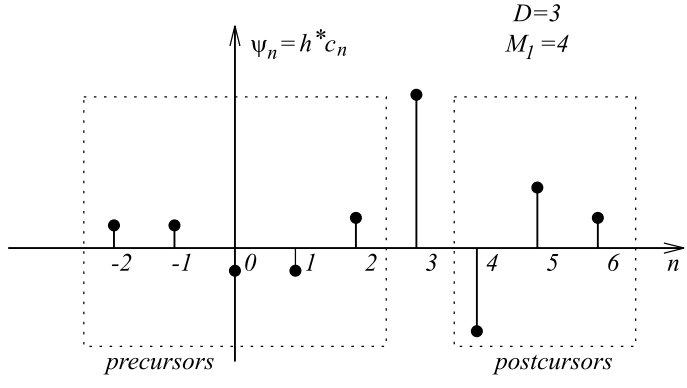
If the *past symbols* and the impulse response $\{h_n\}$ were perfectly known, we could use an ISI cancellation scheme limited only to postcursors. Substituting the *past symbols* with their detected versions $\{\hat{a}_{k-1}, \ldots, \hat{a}_{k-N_2}\}$, we obtain a scheme to cancel in part ISI, as illustrated in Figure 8.11, where, in general, the *feedback filter* has impulse response $\{b_n\}$, $n = 1, \ldots, M_2$, and output given by

$$x_{FB,k} = b_1 \hat{a}_{k-1} + \cdots + b_{M_2} \hat{a}_{k-M_2} \tag{8.76}$$

If $M_2 \geq N_2$, $b_n = -h_n$, for $n = 1, \ldots, N_2$, $b_n = 0$, for $n = N_2 + 1, \ldots, M_2$, and $\hat{a}_{k-i} = a_{k-i}$, for $i = 1, \ldots, N_2$, then the DFE cancels the ISI due to postcursors. We note that this is done without changing the noise $\tilde{w}_k$ that is present in x_k.



(a) Before the FF filter.



(b) After the FF filter.

Figure 8.10. Discrete-time pulses in a DFE.

The general structure of a DFE is shown in Figure 8.12, where two filters and the detection delay are outlined:

1. *Feedforward (FF) filter* c, with M_1 coefficients,

$$z_k = x_{FF,k} = \sum_{i=0}^{M_1-1} c_i x_{k-i} \tag{8.77}$$

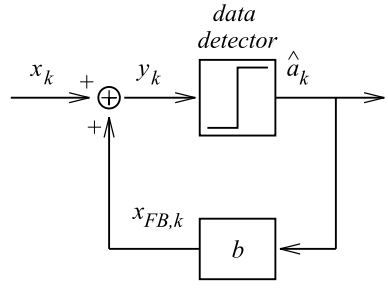


Figure 8.11. Simplified scheme of a DFE, where only the feedback filter is included.

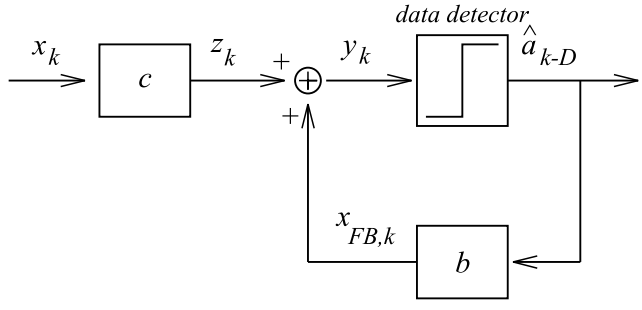


Figure 8.12. General structure of the DFE.

2. *Feedback (FB) filter* b, with M_2 coefficients,

$$x_{FB,k} = \sum_{i=1}^{M_2} b_i a_{k-i-D} \tag{8.78}$$

Moreover,

$$y_k = x_{FF,k} + x_{FB,k} \tag{8.79}$$

We recall that for a LE the goal is to obtain a pulse $\{\psi_n\}$ free of ISI, with respect to the desired sample ψ_D. Now (see Figure 8.10) ideally the task of the feedforward filter is to obtain an overall impulse response $\{\psi_n = h * c_n\}$ with very small precursors and a transfer function $\mathcal{Z}[\psi_n]$ that is minimum phase (see Example 1.4.3). In this manner almost all the ISI is cancelled by the FB filter. We note that the FF filter may be implemented as a FSE, whereas the feedback filter operates with sampling period equal to T.

The choice of the various parameters depends on $\{h_n\}$. The following guidelines, however, are usually observed.

1. $M_1 T/F_0$ (time-span of the FF filter) at least equal to $(N_1 + N_2 + 1)T/F_0$ (time-span of h), so the FF filter can effectively equalize;

2. M_2T (time-span of the FB filter) equal to or less than $(M_1 - 1)T/F_0$ (time-span of the FF filter minus one); M_2 depends also on the delay D, which determines the number of postcursors.

3. For very dispersive channels, for which we have $N_1 + N_2 \gg M_1$, it results $M_2T \gg \frac{M_1T}{F_0}$.

4. The choice of DT, equal to the detection delay, is obtained by initially choosing a large delay, $D \geqslant (M_1 - 1)/F_0$, to simplify the FB filter. If the precursors are not negligible, to reduce the constraints on the coefficients of the FF filter the value of D is lowered and the system is iteratively designed. In practice, DT is equal to or smaller than $(M_1 - 1)T/F_0$.

   For a LE, instead, DT is approximately equal to $\frac{N-1}{2}\frac{T}{F_0}$; the criterion is that the center of gravity of the coefficients of the filter c is approximately equal to $(N - 1)/2$.

The detection delays discussed above are referred to a pulse $\{h_n\}$ "centered" in the origin, that does not introduce any delay.

Adaptive DFE

We consider the scheme implemented in Figure 8.13, where the output signal is given by

$$y_k = \sum_{i=0}^{M_1-1} c_i x_{k-i} + \sum_{j=1}^{M_2} b_j a_{k-D-j} \tag{8.80}$$

Defining the coefficient vector

$$\boldsymbol{\zeta} = [c_0, c_1, \ldots, c_{M_1-1}, b_1, b_2, \ldots, b_{M_2}]^T \tag{8.81}$$

and the input vector

$$\boldsymbol{\xi}_k = [x_k, x_{k-1}, \ldots, x_{k-M_1+1}, \hat{a}_{k-D-1}, \hat{a}_{k-D-2}, \ldots, \hat{a}_{k-D-M_2}]^T \tag{8.82}$$

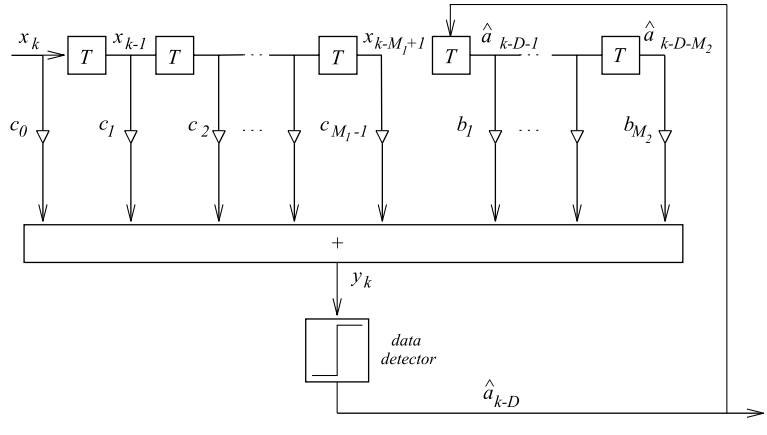


Figure 8.13. Implementation of a DFE.

we express (8.80) in vector form:

$$y_k = \boldsymbol{\zeta}^T \boldsymbol{\xi}_k \tag{8.83}$$

The error is given by

$$e_k = \hat{a}_{k-D} - y_k \tag{8.84}$$

We recall that, during the transmission of a training sequence, $\hat{a}_{k-D} = a_{k-D}$.

The LMS adaptation is given by

$$\boldsymbol{\zeta}_{k+1} = \boldsymbol{\zeta}_k + \mu e_k \boldsymbol{\xi}_k^* \tag{8.85}$$

Design of a DFE with a finite number of coefficients

If the channel impulse response $\{h_i\}$ and the autocorrelation function of the noise $\mathtt{r}_{\tilde{w}}(n)$ are known, for the MSE criterion with

$$J = E[|a_{k-D} - y_k|^2] \tag{8.86}$$

the Wiener filter theory may be applied to determine the optimum coefficients of the DFE filter in the case $\hat{a}_k = a_k$, with the usual assumptions of symbols that are i.d.d. and statistically independent of the noise.

For a generic sequence $\{h_i\}$ in (8.73), we recall the following results:

1. *cross-correlation* between a_k and x_k

$$\mathtt{r}_{ax}(n) = \sigma_a^2 h_{-n}^* \tag{8.87}$$

2. *autocorrelation* of x_k

$$\mathtt{r}_x(n) = \sigma_a^2 \mathtt{r}_h(n) + \mathtt{r}_{\tilde{w}}(n) \tag{8.88}$$

where

$$\mathtt{r}_h(n) = \sum_{j=-N_1}^{N_2} h_j h_{j-n}^* \qquad \mathtt{r}_{\tilde{w}}(n) = N_0 \, \mathtt{r}_{g_M}(nT) \tag{8.89}$$

Defining

$$\psi_p = h * c_p = \sum_{\ell=0}^{M_1-1} c_\ell h_{p-\ell} \tag{8.90}$$

equation (8.80) becomes

$$y_k = \sum_{p=-N_1}^{N_2+M_1-1} \psi_p a_{k-p} + \sum_{i=0}^{M_1-1} c_i \tilde{w}_{k-i} + \sum_{j=1}^{M_2} b_j a_{k-D-j} \tag{8.91}$$

Observing (8.91), the optimum choice of the feedback filter coefficients is given by

$$b_i = -\psi_{i+D} \qquad i = 1, \ldots, M_2 \tag{8.92}$$

Substitution of (8.92) in (8.80) yields

$$y_k = \sum_{i=0}^{M_1-1} c_i \left(x_{k-i} - \sum_{j=1}^{M_2} h_{j+D-i} a_{k-j-D} \right) \tag{8.93}$$

To obtain the Wiener–Hopf solution, the following correlations are needed:

$$[\mathbf{p}]_p = E\left[a_{k-D} \left(x_{k-p} - \sum_{j=1}^{M_2} h_{j+D-p} a_{k-D-j} \right)^* \right] = \sigma_a^2 h_{D-p}^* \qquad p = 0, 1, \ldots, M_1 - 1 \tag{8.94}$$

$$\begin{aligned} [\mathbf{R}]_{p,q} &= E\left[\left(x_{k-q} - \sum_{j_1=1}^{M_2} h_{j_1+D-q} a_{k-D-j_1} \right) \left(x_{k-p} - \sum_{j_2=1}^{M_2} h_{j_2+D-p} a_{k-D-j_2} \right)^* \right] \\ &= \sigma_a^2 \left(\sum_{j=-N_1}^{N_2} h_j h_{j-(p-q)}^* - \sum_{j=1}^{M_2} h_{j+D-q} h_{j+D-p}^* \right) + r_{\tilde{w}}(p-q) \\ &\qquad p, q = 0, 1, \ldots, M_1 - 1 \end{aligned} \tag{8.95}$$

Therefore the optimum feedforward filter coefficients are given by

$$\mathbf{c}_{opt} = \mathbf{R}^{-1}\mathbf{p} \tag{8.96}$$

and, from (8.92), the optimum feedback filter coefficients are given by

$$b_i = -\sum_{\ell=0}^{M_1-1} c_{opt,\ell} h_{i+D-\ell} \qquad i = 1, 2, \ldots, M_2 \tag{8.97}$$

Moreover, using (8.94), we get

$$\begin{aligned} J_{min} &= \sigma_a^2 - \sum_{\ell=0}^{M_1-1} c_{opt,\ell} [\mathbf{p}]_\ell^* \\ &= \sigma_a^2 \left(1 - \sum_{\ell=0}^{M_1-1} c_{opt,\ell} h_{D-\ell} \right) \end{aligned} \tag{8.98}$$

Observation 8.1
In the particular case in which all the postcursors are cancelled by the feedback filter, that is for

$$M_2 + D = N_2 + M_1 - 1 \tag{8.99}$$

(8.95) is simplified as

$$[\mathbf{R}]_{p,q} = \sigma_a^2 \sum_{j=-N_1}^{D} h_{j-q} h_{j-p}^* + \mathtt{r}_{\tilde{w}}(p-q) \tag{8.100}$$

Observation 8.2
The equations to determine $\mathbf{c}_{opt}$ for a LE are identical to (8.94)–(8.98), with $M_2 = 0$. In particular, the vector $\mathbf{p}$ in (8.94) is not modified, while the expression of the elements of the matrix $\mathbf{R}$ in (8.95) is modified by the terms including the detected symbols.

Observation 8.3
For white noise w_C, the autocorrelation of $\tilde{w}_k$ is proportional to the autocorrelation of the receive filter impulse response: consequently, if the statistical power of $\tilde{w}_k$ is known, the autocorrelation $\mathtt{r}_{\tilde{w}}(n)$ is easily determined. Finally, the coefficients of the channel impulse response $\{h_n\}$ and the statistical power of $\tilde{w}_k$ used in $\mathbf{R}$ and $\mathbf{p}$ can be determined by the methods given in Appendix 3.B.

Observation 8.4
For a LE, the matrix $\mathbf{R}$ is Hermitian and Toeplitz, while for a DFE it is only Hermitian; in any case it is (semi-)definite positive. Efficient methods to determine the inverse of the matrix are described in Section 2.3.2.

Observation 8.5
The definition of $\{h_n\}$ depends on the value of t_0, which is determined by methods described in Chapter 14. A particularly useful method to determine the impulse response $\{h_n\}$ in wireless systems (see Chapter 18) resorts to a short training sequence to achieve fast synchronization. We recall that a fine estimate of $t_{0,MF}$ is needed if the sampling period of the signal at the MF output is equal to T. The overall discrete-time system impulse response obtained by sampling the output signal of the anti-aliasing filter g_{AA} (see Figure 8.4) is assumed to be known, e.g. by estimation. The sampling period, for example $T/8$, is in principle determined by the accuracy with which we desire to estimate the timing phase $t_{0,MF}$ at the MF output. To reduce implementation complexity, however, a larger sampling period of the signal at the MF input is considered, for example, $T/2$. We then implement the MF g_M by choosing, among the four polyphase components (see Section 1.A.9 on page 119) of the impulse response, the component with largest energy, thus realizing the MF criterion (see also (8.16)). This is equivalent to selecting among the four possible components with sampling period $T/2$ of the sampled output signal of the filter g_{AA} the component with largest statistical power. This method is similar to the timing estimator (14.117).

The timing phase $t_{0,AA}$ for the signal at the input of g_M is determined during the estimation of the channel impulse response. It is usually chosen either as the time at which the first useful sample of the overall impulse response occurs, or the time at which the peak of the impulse response occurs, shifted by a number of modulation intervals corresponding to a given number of precursors. Note that, if t_{MF} denotes the duration of g_M, then the timing phase at the output of g_M is given by $t_{0,MF} = t_{0,AA} + t_{MF}$. The criterion (7.269), according to which t_0 is chosen in correspondence of the correlation peak, is a particular case of this procedure.

Observation 8.6
In systems where the training sequence is placed at the end of a block of data (see the GSM frame in Appendix 17.A), it is convenient to process the observed signal $\{x_k\}$ starting from the end of the block, let's say from $k = K - 1$ to 0, thus exploiting the knowledge of the training sequence. Now if $\{\psi_n = h * c_n\}$ and $\{b_n\}$ are the optimum impulse responses if the signal is processed in the forward mode, i.e. for $k = 0, 1, \ldots, K - 1$, it is easy to verify that $\{\psi_n^{B*}\}$ and $\{b_n^{B*}\}$, where B is the backward operator defined on page 27, are the optimum impulse responses in the backward mode for $k = K - 1, \ldots, 1, 0$, apart from a constant delay. In fact, if $\{\psi_n\}$ is ideally minimum phase and causal with respect to the timing phase, now $\{\psi_n^{B*}\}$ is maximum phase and anticausal with respect to the new instant of optimum sampling. Also the FB filter will be anticausal. In the particular case $\{h_n\}$ is a correlation sequence, then $\{\psi_n^{B*}\}$ can be obtained using as FF filter the filter having impulse response $\{c_n^{B*}\}$

Design of a fractionally spaced DFE (FS-DFE)

We briefly describe the equations to determine the coefficients of a DFE comprising a fractionally spaced FF filter with M_1 coefficients and sampling period of the input signal equal to $T/2$, and an FB filter with M_2 coefficients; the extension to an FSE with an oversampling factor $F_0 > 2$ is straightforward.

We consider the scheme of Figure 8.12, where the FF filter c is now a fractionally spaced filter, as illustrated in Figure 8.7. The overall receiver structure is shown in Figure 8.14, where the filter g_{AA} may be more sophisticated than a simple anti-aliasing filter and partly perform the function of the MF. Otherwise the function of the MF may be performed by a discrete-time filter placed in front of the filter c. We recall that the MF, besides reducing the complexity of c (see task 2 on page 628), facilitates the optimum choice of t_0 (see Chapter 14).

As usual, the symbols are assumed i.i.d. and statistically independent of the noise signal. If $\{x_n\}$ is the input signal of the FF filter, we have

$$x_n = \sum_{k=-\infty}^{+\infty} h_{n-2k} a_k + \tilde{w}_n \tag{8.101}$$

The signal $\{y_k\}$ at the DFE output is given by

$$y_k = \sum_{i=0}^{M_1-1} c_i x_{2k-i} + \sum_{j=1}^{M_2} b_j a_{k-D-j} \tag{8.102}$$

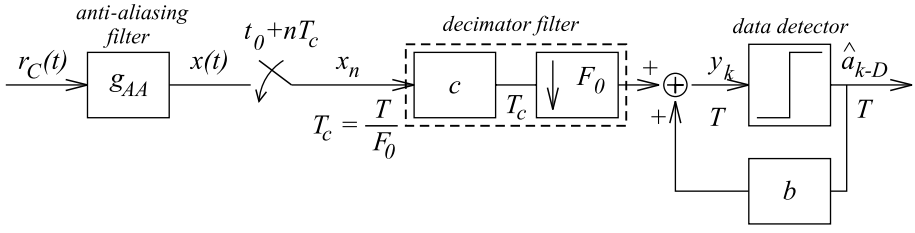


Figure 8.14. FS-DFE structure.

Let

$$\psi_p = h * c_p = \sum_{\ell=0}^{M_1-1} c_\ell h_{p-\ell} \tag{8.103}$$

then the optimum choice of the coefficients $\{b_i\}$ is given by

$$b_i = -\psi_{2(i+D)} = -\sum_{\ell=0}^{M_1-1} c_\ell h_{2(i+D)-\ell} \qquad i = 1, \ldots, M_2 \tag{8.104}$$

With this choice (8.102) becomes

$$y_k = \sum_{i=0}^{M_1-1} c_i \left(x_{2k-i} - \sum_{j=1}^{M_2} h_{2(j+D)-i} a_{k-(j+D)} \right) \tag{8.105}$$

Using the following relations (see also Example 1.9.10 on page 72)

$$E[a_{k-K}\, x^*_{2k-i}] = \sigma_a^2\, h^*_{2K-i} \tag{8.106}$$

$$E[x_{2k-q}\, x^*_{2k-p}] = \sigma_a^2 \sum_{n=-\infty}^{+\infty} h_{2n-q}\, h^*_{2n-p} + \mathtt{r}_{\tilde{w}}(p-q) \tag{8.107}$$

where, from (8.57),

$$\mathtt{r}_{\tilde{w}}(m) = N_0\, \mathtt{r}_{g_{AA}}\left(m\,\frac{T}{2}\right) \tag{8.108}$$

the components of the vector $\mathbf{p}$ and the matrix $\mathbf{R}$ of the Wiener problem associated with (8.105) are given by

$$[\mathbf{p}]_p = \sigma_a^2 h^*_{2D-p} \qquad p = 0, 1, \ldots, M_1 - 1 \tag{8.109}$$

$$[\mathbf{R}]_{p,q} = \sigma_a^2 \left(\sum_{n=-\infty}^{+\infty} h_{2n-q} h^*_{2n-p} - \sum_{j=1}^{M_2} h_{2(j+D)-q} h^*_{2(j+D)-p} \right) + \mathtt{r}_{\tilde{w}}(p-q)$$
$$p, q = 0, 1, \ldots, M_1 - 1 \tag{8.110}$$

The feedforward filter is obtained by solving the system of equations

$$\mathbf{R}\,\mathbf{c}_{opt} = \mathbf{p} \tag{8.111}$$

and the feedback filter is determined from (8.104). The minimum value of the cost function is given by

$$\begin{aligned} J_{min} &= \sigma_a^2 - \sum_{\ell=0}^{M_1-1} c_{opt,\ell}[\mathbf{p}]_\ell^* \\ &= \sigma_a^2 \left(1 - \sum_{\ell=0}^{M_1-1} c_{opt,\ell} h_{2D-\ell}\right) \end{aligned} \tag{8.112}$$

A problem encountered with this method is the inversion of the matrix $\mathbf{R}$ in (8.111), because it may be ill-conditioned. Similarly to the procedure outlined on page 187, a solution consists in adding a positive constant to the elements on the diagonal of $\mathbf{R}$, so that $\mathbf{R}$ becomes invertible; obviously the value of this constant must be rather small, so that the performance of the optimum solution does not change significantly.

Observation 8.7
Observations similar to the observations 8.1–8.3 hold for a FS-DFE, with appropriate changes. In this case the timing phase t_0 after the filter g_{AA} can be determined with accuracy $T/2$, for example by the correlation method (7.269).

For an FSE, or FS-LE, the equations to determine $\mathbf{c}_{opt}$ are given by (8.109)–(8.111) with $M_2 = 0$. Note that the matrix $\mathbf{R}$ is Hermitian but in general it is no longer Toeplitz.

Observation 8.8
Two matrix formulations of the direct method to determine the coefficients of a DFE and an FS-DFE are given in Appendix 8.B. In particular, a formulation uses the correlation of the equalizer input signal $\{x_n\}$, the correlation of the sequence $\{a_k\}$, and the cross-correlation of the two signals: using suitable estimates of the various correlations (see the correlation method and the covariance method considered in Section 2.3), this method avoids the need for the estimate of the overall channel impulse response; however, it requires a greater computational complexity with respect to the method described in this section.

Signal-to-noise ratio γ

Using FF and FB filters with an infinite number of coefficients, it is possible to achieve the minimum value of J_{min}. Salz derived the expression of J_{min} for this case, given by [3]

$$J_{min} = \sigma_a^2 \exp\left(T \int_{-\frac{1}{2T}}^{\frac{1}{2T}} \ln \frac{N_0}{N_0 + \sigma_a^2 \Phi(e^{j2\pi fT})}\, df\right) \tag{8.113}$$

where Φ is defined in (8.27).

Applying the Jensen's inequality,

$$e^{\int f(a)\,da} \leq \int e^{f(a)}\,da \tag{8.114}$$

to (8.113), we can compare the performance of a linear equalizer given by (8.40) with that of a DFE given by (8.113): the result is that, assuming $\Phi(e^{j2\pi fT}) \neq$ constant and the absence of detection errors in the DFE, for infinite-order filters the value J_{min} of a DFE is always smaller than J_{min} of a LE.

If FF and FB filters with a finite number of coefficients are employed, in analogy with (8.44), also for a DFE we have

$$\gamma_{DFE} \backsimeq \frac{2}{J_{min}} \tag{8.115}$$

where J_{min} is given by (8.98). An analogous relation holds for an FS-DFE, with J_{min} given by (8.112).

Remarks

1. In the absence of errors of the data detector, the DFE has better asymptotic ($M_1, M_2 \to \infty$) performance than the linear equalizer. However, for a given finite number of coefficients $M_1 + M_2$, the performance is a function of the channel and of the choice of M_1.

2. The DFE is definitely superior to the linear equalizer for channels that exhibit large variations of the attenuation in the passband, as in such cases the linear equalizer tends to enhance the noise.

3. Detection errors tend to spread, because they produce *incorrect cancellations*. Error propagation leads to an increase of the error probability. However, simulations indicate that for typical channels and symbol error probability smaller than $5 \cdot 10^{-2}$, error propagation is not catastrophic.

4. For channels with impulse response $\{h_i\}$, such that detection errors may spread catastrophically, instead of the DFE structure it is better to implement the linear FF equalizer at the receiver, and the FB filter at the transmitter as a precoder, using the *precoding* method discussed in Appendix 7.A and Chapter 13.

8.6 Convergence behavior of adaptive equalizers

We consider the digital transmission model of Figure 8.5, and observe the performance of adaptive LE and DFE by resorting to a specific channel realization. In particular, we analyze two cases in which the discrete-time overall impulse response of the system, $\{h_n\}$, is either minimum phase, h_{min}, as in Figure 1.15a, or non-minimum phase, h_{nom}, as in Figure 1.15c. The additive channel noise $\tilde{w}_k$ is AWGN with statistical power $\sigma^2_{\tilde{w}}$, such that

the signal-to-noise ratio $\Gamma = \sigma_a^2 \cdot r_h(0)/\sigma_{\tilde{w}}^2$ at the equalizer input is equal to 20 dB. The sequence of symbols $a_k \in \{-1, 1\}$ is a PN training sequence of length $L = 63$.

Adaptive LE

With reference to the scheme of Figure 8.6, we consider a LE with $N = 15$ coefficients. In terms of mean-square error at convergence, the best results are obtained for a delay $D = 0$ in the case of h_{min} and $D = 8$ in the case of h_{nom}; we observe that the overall impulse response h_{nom} is not centered at the origin, and the delay D is the sum of the delays introduced by $\{h_n\}$ and $\{c_n\}$.

Figures 8.15c, d and 8.16c, d show curves of mean-square error convergence for standard LMS and RLS algorithms (see Section 3.1.2) for minimum and non-minimum phase channels, respectively; in the plots, J_{min} represents the minimum value of J achieved with optimum coefficients computed by the direct method. We note that J_{min} is 4 dB higher than the value given by $\sigma_{\tilde{w}}^2$, because of the noise and residual ISI at the decision point. The impulse response $\{c_{opt,n}\}$ of the optimum LE and the overall system impulse response $\{\psi_n = h * c_{opt,n}\}$ are depicted in Figures 8.15a, b and 8.16a, b for the two channels.

The curves of convergence of $J(k)$ indicate that the RLS algorithm succeeds in achieving convergence by the end of the training sequence, whereas the LMS algorithm still presents a considerable offset from the optimum conditions, even though a large adaptation gain μ is chosen.

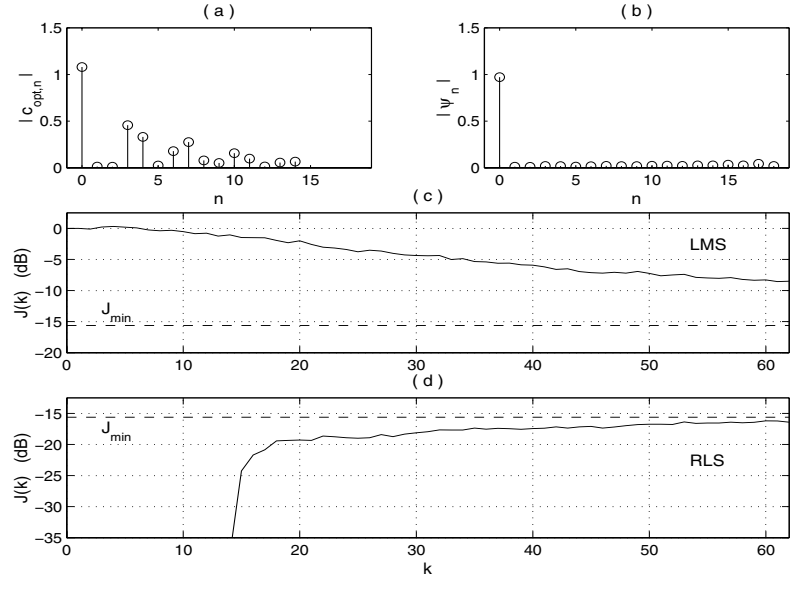


Figure 8.15. System impulse responses and curves of mean-square error convergence, estimated over 500 realizations, for a channel with minimum phase impulse response, and a LE employing the LMS with $\mu = 0.062$ or the RLS.

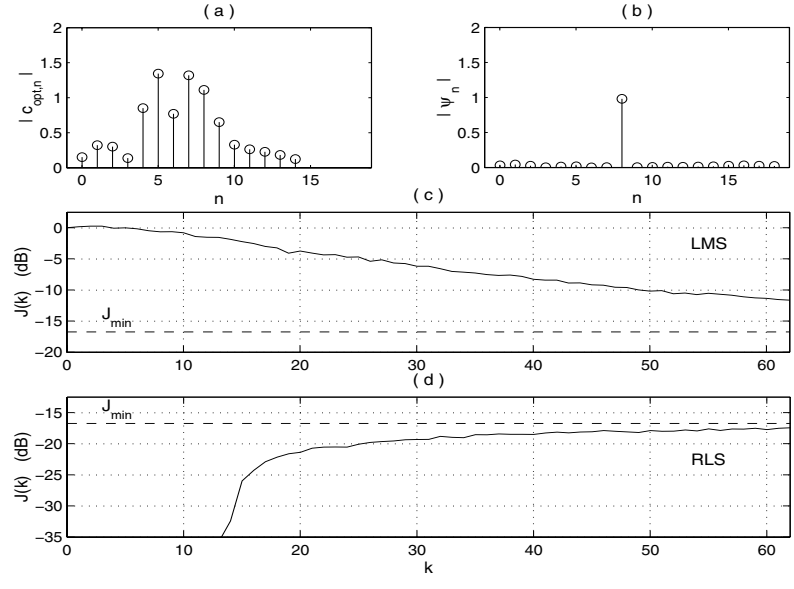


Figure 8.16. System impulse responses and curves of mean-square error convergence, estimated over 500 realizations, for a channel with non-minimum phase impulse response, and a LE employing the LMS with $\mu = 0.343$ or the RLS.

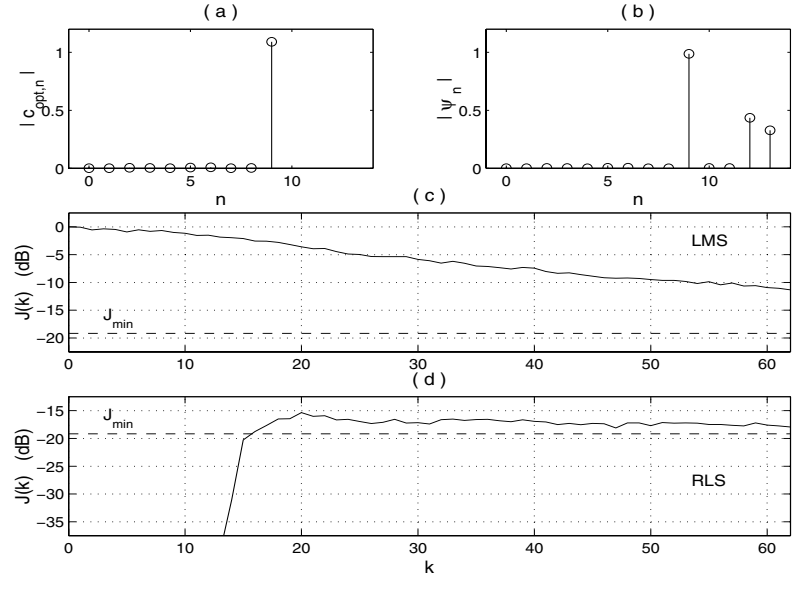


Figure 8.17. System impulse responses and curves of mean-square error convergence, estimated over 500 realizations, for a channel with minimum phase impulse response, and a DFE employing the LMS with $\mu = 0.063$ or the RLS.

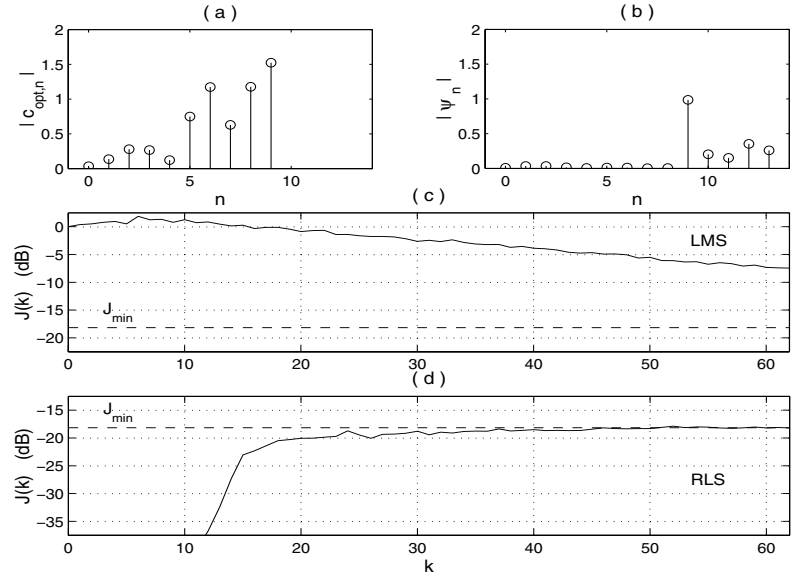


Figure 8.18. System impulse responses and curves of mean-square error convergence, estimated over 500 realizations, for a channel with non-minimum phase impulse response, and a DFE employing the LMS with $\mu = 0.143$ or the RLS.

Adaptive DFE

We consider now the performance of a DFE as illustrated in Figure 8.13, with parameters $M_1 = 10$, $M_2 = 5$, and $D = 9$, for both h_{min} and h_{nom}. Also in this case the chosen value of D gives the best results in terms of the value of J at convergence.

Figures 8.17c, d and 8.18c, d show curves of mean-square error convergence for standard LMS and RLS algorithms, for minimum and non-minimum phase channels, respectively; The impulse response $\{c_{opt,n}\}$ of the optimum FF filter and the overall system impulse response $\{\psi_n = h * c_{opt,n}\}$ are depicted in Figures 8.17a, b and 8.18a, b, for the two channels.

8.7 LE-ZF with a finite number of coefficients

Ignoring the noise, the signal at the output of a LE with N coefficients (see (8.91)) is given by

$$y_k = \sum_{i=0}^{N-1} c_i \, x_{k-i} = \sum_{p=-N_1}^{N_2+N-1} \psi_p \, a_{k-p} \tag{8.116}$$

where, from (8.90),

$$\psi_p = \sum_{\ell=0}^{N-1} c_\ell \, h_{p-\ell} = c_0 \, h_p + c_1 \, h_{p-1} + \cdots + c_{N-1} \, h_{p-(N-1)} \qquad (8.117)$$

$$p = -N_1, \ldots, 0, \ldots, N_2 + N - 1$$

For a LE-ZF it must be

$$\psi_p = \delta_{p-D} \qquad (8.118)$$

where D is a suitable delay.

If the overall impulse response $\{h_n\}$, $n = -N_1, \ldots, N_2$, is known, a method to determine the coefficients of the LE-ZF consists in considering the system (8.117) with $N_t = N_1 + N_2 + N$ equations and N unknowns, that can be solved by the method of the pseudoinverse (see (2.185)); alternatively, the solution can be found in the frequency domain by taking the N_t-point DFT of the various signals (see (1.108)), and the result windowed in the time domain so that the filter coefficients are given by the N consecutive coefficients that maximize the energy of the filter impulse response.

An approximate solution is obtained by forcing the condition (8.118) only for N values of p in (8.117) centered around D; then the matrix of the system (8.117) is square and, if the determinant is different from zero, it can be inverted.

Note that all these methods require an accurate estimate of the overall impulse response, otherwise the equalizer coefficients may deviate considerably from the desired values. An alternative robust method, which does not require the knowledge of $\{h_n\}$ and can be extended to FSE-ZF systems, will be presented in Section 8.15.

An adaptive ZF equalization method is discussed in Appendix 8.C.

8.8 DFE: alternative configurations

We determine the expressions of the FF and FB filters of a DFE in the case of IIR filter structure.

DFE-ZF

We consider a receiver with a matched filter g_M (see Figure 8.1) followed by the DFE illustrated in Figure 8.19, where, to simplify the notation, we assume $t_0 = 0$ and $D = 0$. The z-transform of the QAM system impulse response is given by $\Phi(z)$, as defined in (8.25).

With reference to Figure 8.19, the matched filter output x_k is input to a linear equalizer *zero forcing* (LE-ZF) with transfer function $1/\Phi(z)$ to remove the ISI: therefore the LE-ZF output is given by

$$x_{E,k} = a_k + w_{E,k} \qquad (8.119)$$

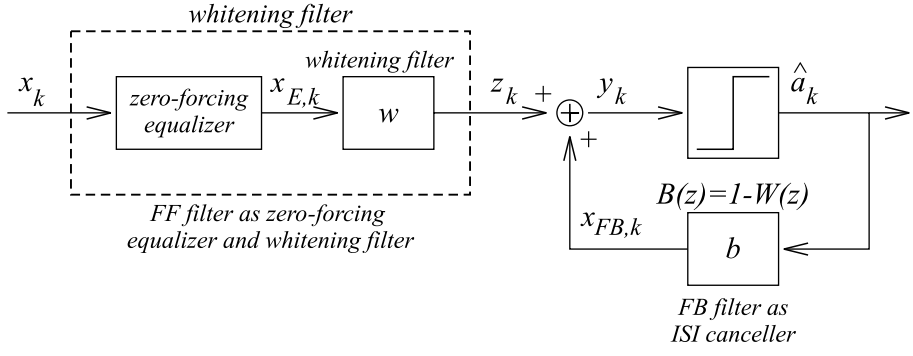


Figure 8.19. DFE zero-forcing.

From (8.29), using the property (8.26), we obtain that the spectrum $P_{w_E}(z)$ of $w_{E,k}$ is given by

$$\begin{aligned} P_{w_E}(z) &= N_0 \Phi(z) \frac{1}{\Phi(z)\Phi^*\left(\dfrac{1}{z^*}\right)} \\ &= N_0 \frac{1}{\Phi(z)} \end{aligned} \tag{8.120}$$

As $\Phi(z)$ is the z-transform of a correlation sequence, it can be factorized as (see page 53)

$$\Phi(z) = F(z)\, F^*\left(\frac{1}{z^*}\right) \tag{8.121}$$

where

$$F(z) = \sum_{n=0}^{+\infty} \mathtt{f}_n\, z^{-n} \tag{8.122}$$

is a minimum-phase function, that is with poles and zeros inside the unit circle, associated with a causal sequence $\{\mathtt{f}_n\}$.

Observation 8.9

A useful method to determine the filter $F(z)$ in (8.121), with a computational complexity that is proportional to the square of the number of filter coefficients, is obtained by considering a minimum-phase prediction error filter $A(z) = 1 + \mathtt{a}'_{1,N}\, z^{-1} + \cdots + \mathtt{a}'_{N,N}\, z^{-N}$ (see page 147), designed using the ACS $\{\mathtt{r}_{q_C}(nT)\}$ defined by (8.23). The equation to determine the coefficients of $A(z)$ is given by (2.85), where $\{\mathtt{r}_x(n)\}$ is now substituted by $\{\mathtt{r}_{q_C}(nT)\}$. The final result is $F(z) = \mathtt{f}_0/A(z)$.

On the other hand, $F^*(1/z^*)$ is a function with zeros and poles outside the unit circle, associated with an anticausal sequence $\{f^*_{-n}\}$.

We choose as transfer function of the filter w in Figure 8.19 the function

$$W(z) = F(z)\frac{1}{\mathtt{f}_0} \tag{8.123}$$

The ISI term in z_k is determined by $W(z)-1$, hence *there are no precursors*; the noise is white with statistical power$(N_0/\mathtt{f}_0^2)$. Therefore the filter w is called *whitening filter* (WF). In any case, the filter composed of the cascade of LE-ZF and w is also a WF.

If $\hat{a}_k = a_k$ then, for

$$B(z) = 1 - W(z) \tag{8.124}$$

the FB filter removes the ISI present in z_k and leaves the white noise unchanged. As y_k is not affected by ISI, this structure is called DFE-ZF, for which we obtain

$$\gamma_{DFE-ZF} = \frac{2|\mathtt{f}_0|^2}{N_0} \tag{8.125}$$

Summarizing, the relation between x_k and z_k is given by

$$\frac{1}{\Phi(z)}F(z)\frac{1}{\mathtt{f}_0} = \frac{1}{\mathtt{f}_0 F^*\left(\dfrac{1}{z^*}\right)} \tag{8.126}$$

With this filter the noise in z_k is white. The relation between a_k and the desired signal in z_k is instead governed by $\Psi(z) = F(z)/\mathtt{f}_0$ and $B(z) = 1 - \Psi(z)$. In other words, the overall discrete-time system is causal and minimum phase, that is the energy of the impulse response is mostly concentrated at the beginning of the pulse. The overall receiver structure is illustrated in Figure 8.20, where the block including the *matched filter*, *sampler*, and *whitening filter*, is called *whitened matched filter* (WMF). Note that the impulse response at the WF output has no precursors.

In principle, the WF of Figure 8.20 is non-realizable, because it is anticausal. In practice we can implement it in two ways:

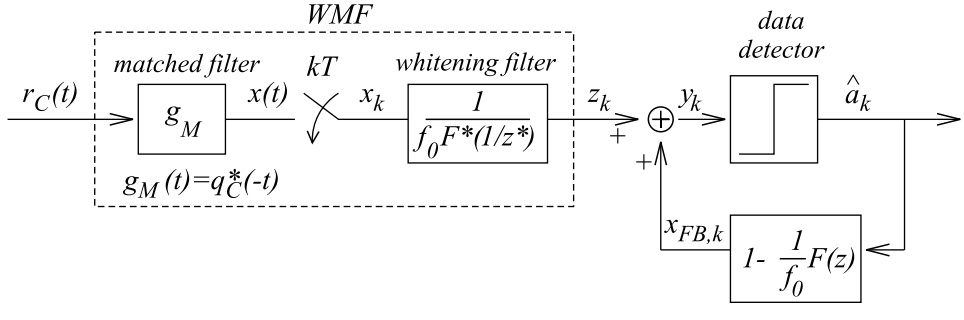


Figure 8.20. DFE-ZF as whitened matched filter followed by a canceller of ISI.

a) by introducing an appropriate delay in the impulse response of an FIR WF, and processing the output samples in the forward mode for $k = 0, 1, \ldots$;

b) by processing the output samples of the IIR WF in the backward mode, for $k = K-1, K-2, \ldots, 0$, starting from the end of the block of samples.

We observe that the choice $F(z) = \mathtt{f}_0/A(z)$, where $A(z)$ is discussed in Observation 8.9 on page 650, leads to an FIR WF with transfer function $\frac{1}{\mathtt{f}_0^2}A^*(1/z^*)$.

Observation 8.10
With reference to the scheme of Figure 8.20, using a LE-ZF instead of a DFE structure means that a filter with transfer function $\mathtt{f}_0/F(z)$ is placed after the WF to produce the signal $x_{E,k}$ given by (8.119). For a data detector based on $x_{E,k}$, the ratio γ is

$$\gamma_{LE-ZF} = \frac{2}{\mathtt{r}_{w_E}(0)} \tag{8.127}$$

where $\mathtt{r}_{w_E}(0)$ is determined as the coefficient of z^0 in $N_0/\Phi(z)$. This expression is alternative to (8.3).

Example 8.8.1 (WF for a channel with exponential impulse response)
A method to determine the WF in the scheme of Figure 8.20 is illustrated by an example. Let

$$q_C(t) = \sqrt{E_{q_C} 2\beta} e^{-\beta t} 1(t) \tag{8.128}$$

be the overall system impulse response at the MF input; in (8.128) E_{q_C} is the energy of q_C.

The autocorrelation of q_C, sampled at instant nT, is given by

$$\mathtt{r}_{q_C}(nT) = E_{q_C} a^{|n|} \qquad a = e^{-\beta T} < 1 \tag{8.129}$$

Then

$$\begin{aligned} \Phi(z) &= \mathcal{Z}[\mathtt{r}_{q_C}(nT)] \\ &= E_{q_C} \frac{(1-a^2)}{-az^{-1} + (1+a^2) - az} \\ &= E_{q_C} \frac{(1-a^2)}{(1-az^{-1})(1-az)} \end{aligned} \tag{8.130}$$

We note that the frequency response of (8.129) is

$$\Phi(e^{j2\pi fT}) = E_{q_C} \frac{(1-a^2)}{1+a^2-2a\cos(2\pi fT)} \tag{8.131}$$

and presents a minimum for $f = 1/(2T)$.

With reference to the factorization (8.121), it is easy to identify the poles and zeros of $\Phi(z)$ inside the unit circle, hence

$$\begin{aligned} F(z) &= \sqrt{E_{q_C}(1-a^2)}\frac{1}{1-az^{-1}} \\ &= \sqrt{E_{q_C}(1-a^2)}\sum_{n=0}^{+\infty} a^n z^{-n} \end{aligned} \tag{8.132}$$

In particular, the coefficient of z^0 is given by

$$\mathtt{f}_0 = \sqrt{E_{q_C}(1-a^2)} \tag{8.133}$$

The WF of Figure 8.20 is expressed as

$$\begin{aligned} \frac{1}{\mathtt{f}_0 F^*\left(\frac{1}{z^*}\right)} &= \frac{1}{E_{q_C}(1-a^2)}(1-az) \\ &= \frac{1}{E_{q_C}(1-a^2)} z(-a+z^{-1}) \end{aligned} \tag{8.134}$$

In this case the WF, apart from a delay of one sample ($D = 1$), can be implemented by a simple FIR with two coefficients, whose values are equal to $-a/(E_{q_C}(1-a^2))$ and $1/(E_{q_C}(1-a^2))$.

The FB filter is a first-order IIR filter with transfer function

$$\begin{aligned} 1 - \frac{1}{\mathtt{f}_0}F(z) &= -\sum_{n=1}^{+\infty} a^n z^{-n} \\ &= 1 - \frac{1}{1-az^{-1}} \\ &= \frac{-az^{-1}}{1-az^{-1}} \end{aligned} \tag{8.135}$$

Example 8.8.2 (WF for a two-ray channel)
In this case we directly specify the autocorrelation sequence at the matched filter output:

$$\Phi(z) = Q_{CC}(z)Q_{CC}^*\left(\frac{1}{z^*}\right) \tag{8.136}$$

where

$$Q_{CC}(z) = \sqrt{E_{q_C}}(q_0 + q_1 z^{-1}) \tag{8.137}$$

with q_0 and q_1 such that

$$|q_0|^2 + |q_1|^2 = 1 \tag{8.138}$$

$$|q_0| > |q_1| \tag{8.139}$$

In this way E_{q_C} is the energy of $\{q_{CC}(nT)\}$ and $Q_{CC}(z)$ is minimum phase.

Equation (8.137) represents the discrete-time model of a wireless system with a two-ray channel. The impulse response is given by

$$q_{CC}(nT) = \sqrt{E_{q_C}}(q_0\delta_n + q_1\delta_{n-1}) \tag{8.140}$$

The frequency response is given by

$$\mathcal{Q}_{CC}(f) = \sqrt{E_{q_C}}(q_0 + q_1 e^{-j2\pi fT}) \tag{8.141}$$

We note that if $q_0 = q_1$, the frequency response has a zero for $f = 1/(2T)$.

From (8.136) and (8.137) we get

$$\Phi(z) = E_{q_C}(q_0 + q_1 z^{-1})(q_0^* + q_1^* z) \tag{8.142}$$

hence, recalling assumption (8.139),

$$F(z) = \sqrt{E_{q_C}}(q_0 + q_1 z^{-1}) = Q_{CC}(z) \tag{8.143}$$

and

$$\mathtt{f}_0 = \sqrt{E_{q_C}} q_0 \tag{8.144}$$

The WF is given by

$$\frac{1}{\mathtt{f}_0 F^*\left(\dfrac{1}{z^*}\right)} = \frac{1}{E_{q_C}|q_0|^2(1 - bz)} \tag{8.145}$$

where

$$b = \left(-\frac{q_1}{q_0}\right)^*$$

We note that the WF has a pole for $z = b^{-1}$, which, recalling (8.139), lies outside the unit circle. In this case, in order to have a stable filter, it is convenient to associate the z-transform $1/(1 - bz)$ with an anticausal sequence,

$$\frac{1}{1 - bz} = \sum_{i=-\infty}^{0} (bz)^{-i} = \sum_{n=0}^{\infty} (bz)^n \tag{8.146}$$

On the other hand, as $|b| < 1$, we can approximate the series by considering only the first $(N-1)$ terms, obtaining

$$\begin{aligned}\frac{1}{\mathtt{f}_0 F^*\left(\frac{1}{z^*}\right)} &\simeq \frac{1}{E_{q_C}|q_0|^2}\sum_{n=0}^{N}(bz)^n \\ &= \frac{1}{E_{q_C}|q_0|^2} z^N[b^N + \cdots + bz^{-(N-1)} + z^{-N}]\end{aligned} \tag{8.147}$$

Consequently the WF, apart from a delay $D = N$, can be implemented by an FIR filter with $N+1$ coefficients.

The FB filter in this case is a simple FIR filter with one coefficient

$$1 - \frac{1}{\mathtt{f}_0}F(z) = -\frac{q_1}{q_0}z^{-1} \tag{8.148}$$

DFE-ZF as a noise predictor

Let $A(z) = \mathcal{Z}[a_k]$. From the identity

$$\begin{aligned}Y(z) &= X_E(z)W(z) + (1 - W(z))A(z) \\ &= X_E(z) + (1 - W(z))(A(z) - X_E(z))\end{aligned} \tag{8.149}$$

the scheme of Figure 8.19 is redrawn as in Figure 8.21, where the FB filter acts as a noise predictor. In fact, for $\hat{a}_k = a_k$, the FB filter input is colored noise. By removing the correlated noise from $x_{E,k}$, we obtain y_k that is composed of white noise, with minimum variance, plus the desired symbol a_k.

DFE as ISI and noise predictor

A variation of the scheme of Figure 8.21 consists in using as FF filter, a minimum-MSE linear equalizer. We refer to the scheme of Figure 8.22, where the filter c is given

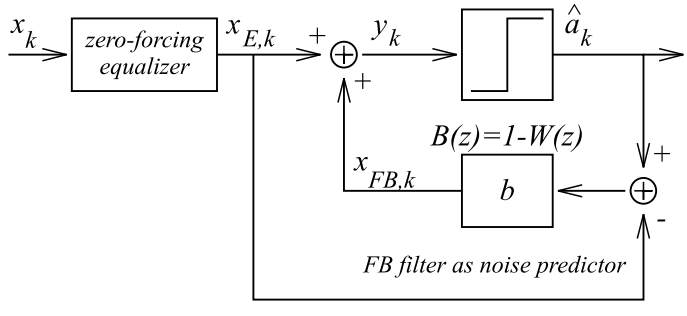


Figure 8.21. Predictive DFE: the FF filter is a linear equalizer zero forcing.

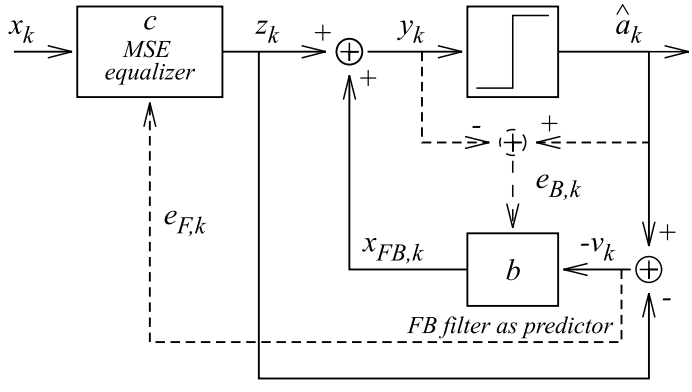


Figure 8.22. Predictive DFE with the FF filter as a minimum-MSE linear equalizer.

by (8.38). The z-transform of the overall impulse response at the FF filter output is given by:

$$\Phi(z)C(z) = \frac{\sigma_a^2 \Phi(z)}{N_0 + \sigma_a^2 \Phi(z)} \tag{8.150}$$

As $t_0 = 0$, the ISI in z_k is given by:

$$\Phi(z)C(z) - 1 = -\frac{N_0}{N_0 + \sigma_a^2 \Phi(z)} \tag{8.151}$$

Hence, the spectral density of the ISI has the following expression:

$$\begin{aligned} P_{ISI}(z) &= P_a(z) \frac{N_0}{N_0 + \sigma_a^2 \Phi(z)} \frac{N_0}{N_0 + \sigma_a^2 \Phi^* \left(\frac{1}{z^*}\right)} \\ &= \frac{\sigma_a^2 (N_0)^2}{(N_0 + \sigma_a^2 \Phi(z))^2} \end{aligned} \tag{8.152}$$

using (8.26) and the fact that the symbols are uncorrelated with $P_a(z) = \sigma_a^2$.

The spectrum of the noise in z_k is given by

$$\begin{aligned} P_{noise}(z) &= N_0 \Phi(z) C(z) C^* \left(\frac{1}{z^*}\right) \\ &= N_0 \frac{\Phi(z)\sigma_a^4}{(N_0 + \sigma_a^2 \Phi(z))^2} \end{aligned} \tag{8.153}$$

Therefore the spectrum of the disturbance v_k in z_k, composed of ISI and noise, is given by

$$P_v(z) = \frac{N_0 \sigma_a^2}{N_0 + \sigma_a^2 \Phi(z)} \tag{8.154}$$

We note that the FF filter could be an FSE and the result (8.154) would not change.

To minimize the power of the disturbance in y_k, the FB filter, with input $\hat{a}_k - z_k = a_k - z_k = -v_k$ (assuming $\hat{a}_k = a_k$), needs to remove the predictable components of z_k. For a predictor of infinite length, we set

$$B(z) = \sum_{n=1}^{+\infty} \mathtt{b}_n z^{-n} \tag{8.155}$$

An alternative form is

$$B(z) = 1 - A(z) \tag{8.156}$$

where

$$A(z) = \sum_{n=0}^{+\infty} \mathtt{a}'_n z^{-n} \qquad \mathtt{a}'_0 = 1 \tag{8.157}$$

is the forward prediction error filter defined in (2.83). To determine $B(z)$ we use the spectral factorization in (1.526):

$$\begin{aligned} P_v(z) &= \frac{\sigma_y^2}{A(z)A^*\left(\dfrac{1}{z^*}\right)} \\ &= \frac{\sigma_y^2}{[1 - B(z)]\left[1 - B^*\left(\dfrac{1}{z^*}\right)\right]} \end{aligned} \tag{8.158}$$

with $P_v(z)$ given by (8.154).

In conclusion, it results that the *prediction error* signal y_k is a white noise process with statistical power equal to σ_y^2.

An adaptive version of the basic scheme of Figure 8.22 suggests that the two filters, c and b, are separately adapted through the error signals $\{e_{F,k}\}$ and $\{e_{B,k}\}$, respectively. This configuration, although sub-optimum with respect to the DFE, is used in conjunction with trellis-coded modulation (see Chapter 12) [4].

8.9 Benchmark performance for two equalizers

We compare limits on the performance of the two equalizers, LE-ZF and DFE-ZF, in terms of the signal-to-noise ratio at the decision point, γ.

Performance comparison

From (8.120), the noise sequence $\{w_{E,k}\}$ can be modeled as the output of a filter having transfer function

$$C_F(z) = \frac{1}{F(z)} \tag{8.159}$$

and input given by white noise with PSD N_0. Because $F(z)$ is causal, also $C_F(z)$ is causal:

$$C_F(z) = \sum_{n=0}^{\infty} c_{F,n} z^{-n} \tag{8.160}$$

where $\{c_{F,n}\}$, $n \geq 0$, is the filter impulse response.

Then we can express the statistical power of $w_{E,k}$ as:

$$\mathtt{r}_{w_E}(0) = N_0 \sum_{n=0}^{\infty} |c_{F,n}|^2 \tag{8.161}$$

where, from (8.159),

$$c_{F,0} = C(\infty) = \frac{1}{F(\infty)} = \frac{1}{\mathtt{f}_0} \tag{8.162}$$

Using the inequality

$$\sum_{n=0}^{\infty} |c_{F,n}|^2 \geq |c_{F,0}|^2 = \frac{1}{|\mathtt{f}_0|^2} \tag{8.163}$$

the comparison between (8.125) and (8.127) yields

$$\gamma_{LE-ZF} \leq \gamma_{DFE-ZF} \tag{8.164}$$

Equalizer performance for two channel models

We now analyze the value of γ for the two simple systems introduced in Examples 8.8.1 and 8.8.2.

LE-ZF

Channel with exponential impulse response. From (8.130) the coefficient of z^0 in $N_0/\Phi(z)$ is equal to

$$\mathtt{r}_{w_E}(0) = \frac{N_0}{E_{q_C}} \frac{1+a^2}{1-a^2} \tag{8.165}$$

and, consequently, from (8.127),

$$\gamma_{LE-ZF} = 2 \frac{E_{q_C}}{N_0} \frac{1-a^2}{1+a^2} \tag{8.166}$$

Using the expression of γ_{MF} (7.113), obtained for a MF receiver in the absence of ISI, we get

$$\gamma_{LE-ZF} = \gamma_{MF} \frac{1-a^2}{1+a^2} \tag{8.167}$$

Therefore the loss due to the ISI, given by the factor $(1-a^2)/(1+a^2)$, can be very large if a is close to 1. In this case the frequency response in (8.131) assumes a minimum value close to zero.

Two-ray channel. From (8.142) we have

$$\frac{1}{\Phi(z)} = \frac{1}{E_{qC}(q_0 + q_1 z^{-1})(q_0^* + q_1^* z)} \tag{8.168}$$

By a partial fraction expansion, we find only the pole for $z = -q_1/q_0$ lies inside the unit circle, hence

$$\mathtt{r}_{w_E}(0) = \frac{N_0}{E_{qC} q_0 \left(q_0^* - q_1^* \dfrac{q_1}{q_0}\right)} = \frac{N_0}{E_{qC}(|q_0|^2 - |q_1|^2)} \tag{8.169}$$

Then

$$\gamma_{LE-ZF} = \gamma_{MF}(|q_0|^2 - |q_1|^2) \tag{8.170}$$

Also in this case we find that the LE is unable to equalize channels with a spectral zero.

DFE-ZF

Channel with exponential impulse response. Substituting the expression of $\mathtt{f}_0$ given by (8.133) in (8.125), we get

$$\begin{aligned} \gamma_{DFE-ZF} &= \frac{2}{N_0} E_{qC}(1 - a^2) \\ &= \gamma_{MF}(1 - a^2) \end{aligned} \tag{8.171}$$

We note that γ_{DFE-ZF} is better than γ_{LE-ZF} by the factor $(1 + a^2)$.

Two-ray channel. Substitution of (8.144) in (8.125) yields

$$\gamma_{DFE-ZF} = \frac{2}{N_0} E_{qC} |q_0|^2 = \gamma_{MF} |q_0|^2 \tag{8.172}$$

In this case the advantage with respect to LE-ZF is given by the factor $|q_0|^2/(|q_0|^2 - |q_1|^2)$, which may be substantial if $|q_1| \simeq |q_0|$.

We recall that in case $E_{qC} \gg N_0$, that is for low noise levels, the performance of LE and DFE are similar to the performance of LE-ZF and DFE-ZF, respectively. Anyway, for the two systems of Examples 8.8.1 and 8.8.2 the values of γ in terms of J_{min} are given in [4], for both LE and DFE.

8.10 Optimum methods for data detection

Adopting an *MSE criterion* at the decision point, we have derived the configuration of Figure 8.1 for an LE, and that of Figure 8.12 for a DFE. In both cases, the decision on a transmitted symbol a_{k-D} is based only on y_k through a memoryless threshold detector.

Actually, the decision criterion that minimizes the probability that an error occurs in the detection of a symbol of the sequence $\{a_k\}$ requires in general that the entire sequence of received samples is considered for symbol detection.

We assume a *sampled signal* having the following structure:

$$z_k = u_k + w_k \qquad k = 0, 1, \ldots, K-1 \tag{8.173}$$

where:

- u_k is the desired signal that carries the information,

$$u_k = \sum_{n=-L_1}^{L_2} \eta_n \, a_{k-n} \tag{8.174}$$

  where η_0 is the sample of the overall system impulse response, obtained in correspondence of the optimum timing phase; $\{\eta_{-L_1}, \ldots, \eta_{-1}\}$ are the precursors, which are typically negligible with respect to η_0. Recalling the expression of the pulse $\{\psi_p\}$ given by (8.90), we have $\eta_n = \psi_{n+D}$.

  We assume the coefficients $\{\eta_n\}$ are known; in practice, however, they are estimated by the methods discussed in Appendix 3.B;

- w_k is a circularly symmetric *white Gaussian* noise, with equal statistical power in the two I and Q components given by $\sigma_I^2 = \sigma_w^2/2$; hence the samples $\{w_k\}$ are uncorrelated and therefore *statistically independent*.

In this section, a general derivation of optimum detection methods is considered; possible applications span, e.g., decoding of convolutional codes (see Chapter 11), coherent demodulation of CPM signals (see Appendix 18.A), and obviously detection of sequences transmitted over channels with ISI.

We introduce the following vectors with K components (K may be very large):

1. Sequence of transmitted symbols, or information message, modeled as a sequence of r.v.s from a finite alphabet

$$\mathbf{a} = [a_{L_1}, a_{L_1+1}, \ldots, a_{L_1+K-1}]^T \qquad a_i \in \mathcal{A} \tag{8.175}$$

2. Sequence of detected symbols, modeled as a sequence of r.v.s from a finite alphabet

$$\hat{\mathbf{a}} = [\hat{a}_{L_1}, \hat{a}_{L_1+1}, \ldots, \hat{a}_{L_1+K-1}]^T \qquad \hat{a}_i \in \mathcal{A} \tag{8.176}$$

3. Sequence of detected symbol values

$$\boldsymbol{\alpha} = [\alpha_{L_1}, \alpha_{L_1+1}, \ldots, \alpha_{L_1+K-1}]^T \qquad \alpha_i \in \mathcal{A} \tag{8.177}$$

4. Sequence of received samples, modeled as a sequence of complex r.v.s

$$\mathbf{z} = [z_0, z_1, \ldots, z_{K-1}] \tag{8.178}$$

5. Sequence of received sample values, or observed sequence

$$\boldsymbol{\rho} = [\rho_0, \rho_1, \ldots, \rho_{K-1}]^T \qquad \rho_i \in \mathcal{C} \quad \boldsymbol{\rho} \in \mathcal{C}^K \tag{8.179}$$

Let M be the cardinality of the alphabet $\mathcal{A}$. By analogy with the analysis of Section 6.1, we divide the vector space of the received samples, $\mathcal{C}^K$, into M^K non-overlapping regions

$$\mathcal{R}_{\boldsymbol{\alpha}} \qquad \boldsymbol{\alpha} \in \mathcal{A}^K \tag{8.180}$$

such that, if $\boldsymbol{\rho}$ belongs to $\mathcal{R}_{\boldsymbol{\alpha}}$, then the sequence $\boldsymbol{\alpha}$ is detected:

$$\text{if } \boldsymbol{\rho} \in \mathcal{R}_{\boldsymbol{\alpha}} \Longrightarrow \hat{\mathbf{a}} = \boldsymbol{\alpha} \tag{8.181}$$

The probability of a correct decision is computed as follows:

$$\begin{aligned} P[\mathrm{C}] &= P[\hat{\mathbf{a}} = \mathbf{a}] \\ &= \sum_{\boldsymbol{\alpha} \in \mathcal{A}^K} P[\hat{\mathbf{a}} = \boldsymbol{\alpha} \mid \mathbf{a} = \boldsymbol{\alpha}] P[\mathbf{a} = \boldsymbol{\alpha}] \\ &= \sum_{\boldsymbol{\alpha} \in \mathcal{A}^K} P[\mathbf{z} \in \mathcal{R}_{\boldsymbol{\alpha}} \mid \mathbf{a} = \boldsymbol{\alpha}] P[\mathbf{a} = \boldsymbol{\alpha}] \\ &= \sum_{\boldsymbol{\alpha} \in \mathcal{A}^K} \int_{\mathcal{R}_\alpha} p_{\mathbf{z}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) d\boldsymbol{\rho}\, P[\mathbf{a} = \boldsymbol{\alpha}] \end{aligned} \tag{8.182}$$

As in (6.18), the following criteria may be adopted to minimize (8.182).

Maximum a posteriori probability (MAP) criterion

$$\boldsymbol{\rho} \in \mathcal{R}_{\boldsymbol{\alpha}} \quad (\text{and } \hat{\mathbf{a}} = \boldsymbol{\alpha}) \quad \text{if} \quad \boldsymbol{\alpha} : \max_{\boldsymbol{\alpha}} p_{\mathbf{z}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) P[\mathbf{a} = \boldsymbol{\alpha}] \tag{8.183}$$

Using the identity

$$p_{\mathbf{z}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) = p_{\mathbf{z}}(\boldsymbol{\rho}) \frac{P[\mathbf{a} = \boldsymbol{\alpha} \mid \mathbf{z} = \boldsymbol{\rho}]}{P[\mathbf{a} = \boldsymbol{\alpha}]} \tag{8.184}$$

the decision criterion becomes:

$$\hat{\mathbf{a}} = \arg\max_{\boldsymbol{\alpha}} P[\mathbf{a} = \boldsymbol{\alpha} \mid \mathbf{z} = \boldsymbol{\rho}] \tag{8.185}$$

An efficient realization of the MAP criterion will be developed in Section 8.10.2.

If all data sequences are equally likely, the MAP criterion coincides with the

maximum likelihood sequence detection (MLSD) criterion

$$\hat{\mathbf{a}} = \arg\max_{\boldsymbol{\alpha}} p_{\mathbf{z}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) \tag{8.186}$$

In other words, *the sequence $\boldsymbol{\alpha}$ is chosen, for which the probability to observe $\mathbf{z} = \boldsymbol{\rho}$ is maximum.*

8.10.1 Maximum likelihood sequence detection

We now discuss a computationally efficient method to find the solution indicated by (8.186). As the vector $\mathbf{a}$ of transmitted symbols is hypothesized to assume the value $\boldsymbol{\alpha}$, both $\mathbf{a}$ and $\mathbf{u} = [u_0, \ldots, u_{K-1}]^T$ are fixed. Recalling that the noise samples are statistically independent, from (8.173) we get:[3]

$$p_{\mathbf{z}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) = \prod_{k=0}^{K-1} p_{z_k|\mathbf{a}}(\rho_k \mid \boldsymbol{\alpha}) \tag{8.187}$$

As w_k is a complex-valued Gaussian r.v. with zero mean and variance σ_w^2, it follows

$$p_{\mathbf{z}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) = \prod_{k=0}^{K-1} \frac{1}{\pi \sigma_w^2} e^{-\frac{1}{\sigma_w^2}|\rho_k - u_k|^2} \tag{8.188}$$

Taking the logarithm, which is a monotonic increasing function, of both members we get

$$-\ln p_{\mathbf{z}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) \propto \sum_{k=0}^{K-1} |\rho_k - u_k|^2 \tag{8.189}$$

where non-essential constant terms have been neglected.

Then the MLSD criterion is formulated as

$$\hat{\mathbf{a}} = \arg\min_{\boldsymbol{\alpha}} \sum_{k=0}^{K-1} |\rho_k - u_k|^2 \tag{8.190}$$

where u_k, defined by (8.174), is a function of the transmitted symbols expressed by the general relation

$$u_k = \tilde{f}(a_{k+L_1}, \ldots, a_k, \ldots, a_{k-L_2}) \tag{8.191}$$

We note that (8.190) is a particular case of the minimum distance criterion (6.30), and it suggests a detecting the vector $\mathbf{u}$ that is closest to the observed vector $\boldsymbol{\rho}$. However, we are interested in detecting the symbols $\{a_k\}$ and not the components $\{u_k\}$.

A *direct computation method* requires that, given the sequence of observed samples, for each possible data sequence $\boldsymbol{\alpha}$ of length K, the corresponding K output samples, elements of the vector $\mathbf{u}$, should be determined, and the relative distance, or metric, should be computed as

$$\Gamma(K-1) = \sum_{k=0}^{K-1} |\rho_k - u_k|^2 \tag{8.192}$$

The detected sequence is the sequence that yields the smallest value of $\Gamma(K-1)$; as in the case of i.i.d. symbols there are M^K possible sequences, this method has a complexity $O(M^K)$.

[3] We note that (8.187) formally requires that the vector $\mathbf{a}$ is extended to include the symbols $a_{-L_2}, \ldots, a_{L_1-1}$.

Lower limit to error probability using the MLSD criterion

We interpret the vector $\mathbf{u}$ as a function of the sequence $\mathbf{a}$, that is $\mathbf{u} = \mathbf{u}(\mathbf{a})$. In the signal space spanned by $\mathbf{u}$, we compute the distances

$$d^2(\mathbf{u}(\boldsymbol{\alpha}), \mathbf{u}(\boldsymbol{\beta})) = \|\mathbf{u}(\boldsymbol{\alpha}) - \mathbf{u}(\boldsymbol{\beta})\|^2 \tag{8.193}$$

for each possible pair of distinct $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in $\mathcal{A}^K$. As the noise is white, we define

$$d^2_{min} = \min_{\boldsymbol{\alpha},\boldsymbol{\beta}} d^2(\mathbf{u}(\boldsymbol{\alpha}), \mathbf{u}(\boldsymbol{\beta})) \tag{8.194}$$

Then the lower limit (6.86) can be used, and we get

$$P_e \geq \frac{N_{min}}{M^K}\, Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{8.195}$$

where M^K is the number of vectors $\mathbf{u}(\mathbf{a})$, and N_{min} is the number of vectors $\mathbf{u}(\mathbf{a})$ whose distance from another vector is d_{min}.

In practice, the exhaustive method for the computation of the expressions in (8.192) and (8.194) is not used; the Viterbi algorithm, that will be discussed in the next section, is utilized instead.

The Viterbi algorithm (VA)

The Viterbi algorithm efficiently implements the ML criterion. With reference to (8.191), it is convenient to describe $\{u_k\}$ as the output sequence of a *finite state machine* (FSM), as discussed in Appendix 8.D. In this case the input is a_{k+L_1}, the state is

$$\mathbf{s}_k = (a_{k+L_1}, a_{k+L_1-1}, \ldots, a_k, \ldots, a_{k-L_2+1}) \tag{8.196}$$

and the output is given in general by (8.191).

We denote by $\mathcal{S}$ the set of the states, that is the set of possible values of $\mathbf{s}_k$:

$$\mathbf{s}_k \in \mathcal{S} = \{\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \ldots, \boldsymbol{\sigma}_{N_s}\} \tag{8.197}$$

With the assumption of i.i.d. symbols, the number of states is equal to $N_s = M^{L_1+L_2}$.

Observation 8.11
We denote by $\boldsymbol{s}'_{k-1}$ the vector that is obtained by removing from $\boldsymbol{s}_{k-1}$ the oldest symbol, a_{k-L_2}. Then

$$\mathbf{s}_k = (a_{k+L_1}, \boldsymbol{s}'_{k-1}) \tag{8.198}$$

From (8.191) and (8.196) we may define u_k as a function of $\boldsymbol{s}_k$ and $\boldsymbol{s}_{k-1}$ as

$$u_k = f(\mathbf{s}_k, \mathbf{s}_{k-1}) \tag{8.199}$$

Defining the metric

$$\Gamma_k = \sum_{i=0}^{k} |\rho_i - u_i|^2 \tag{8.200}$$

the following recursive equation holds:

$$\Gamma_k = \Gamma_{k-1} + |\rho_k - u_k|^2 \tag{8.201}$$

or, using (8.199),

$$\Gamma_k = \Gamma_{k-1} + |\rho_k - f(\mathbf{s}_k, \mathbf{s}_{k-1})|^2 \tag{8.202}$$

Thus we have interpreted $\{u_k\}$ as the output sequence of a finite state machine, and we have expressed recursively the metric (8.192). Note that the metric is a function of the sequence of states $\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_k$, associated with the sequence of output samples $u_0, u_1, \ldots, u_k$. The following example illustrates how to describe the transitions between states of the finite state machine.

Example 8.10.1
Let us consider a transmission system with symbols taken from a binary alphabet, that is $M = 2$, $a_k \in \{-1, 1\}$, and overall impulse response characterized by $L_1 = L_2 = 1$.

In this case we have

$$\mathbf{s}_k = (a_{k+1}, a_k) \tag{8.203}$$

and the set of states contains $N_s = 2^2 = 4$ elements:

$$\mathcal{S} = \{\boldsymbol{\sigma}_1 = (-1, -1), \boldsymbol{\sigma}_2 = (-1, 1), \boldsymbol{\sigma}_3 = (1, -1), \boldsymbol{\sigma}_4 = (1, 1)\} \tag{8.204}$$

The possible transitions

$$\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i \longrightarrow \mathbf{s}_k = \boldsymbol{\sigma}_j \tag{8.205}$$

are represented in Figure 8.23, where a dot indicates a possible value of the state at a certain instant, and a branch indicates a possible transition between two states at consecutive instants. According to (8.198), the variable that determines a transition is a_{k+L_1}. Figure 8.23, extended for all instants k, is called a *trellis diagram*. We note that in this case there are exactly M transitions that leave each state $\mathbf{s}_{k-1}$; likewise there are M transitions that arrive to each state $\mathbf{s}_k$.

With each state $\boldsymbol{\sigma}_j$, $j = 1, \ldots, N_s$, at instant k we associate two quantities:

1. the *path metric*, or *cost* function, defined as:

$$\Gamma(\mathbf{s}_k = \boldsymbol{\sigma}_j) = \min_{\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_k = \boldsymbol{\sigma}_j} \Gamma_k \tag{8.206}$$

2. the *survivor sequence*, defined as the sequence of symbols that ends in that state and determines $\Gamma(\mathbf{s}_k = \boldsymbol{\sigma}_j)$:

$$\mathcal{L}(\mathbf{s}_k = \boldsymbol{\sigma}_j) = (\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_k = \boldsymbol{\sigma}_j) = (a_{L_1}, \ldots, a_{k+L_1}) \tag{8.207}$$

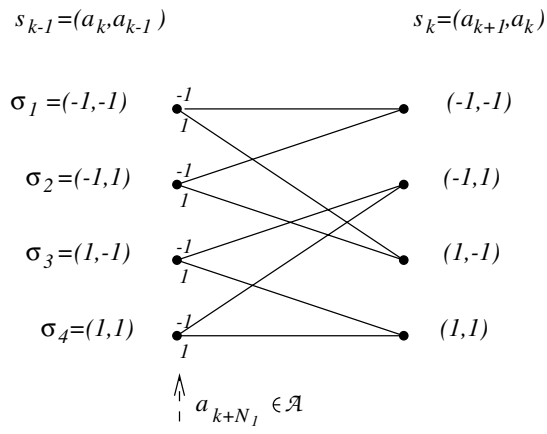


Figure 8.23. Portion of the trellis diagram showing the possible transitions from state $\mathbf{s}_{k-1}$ to state $\mathbf{s}_k$, as a function of the symbol $a_{k+L_1} \in \mathcal{A}$.

Note that the notion of survivor sequence can be equivalently applied to a sequence of symbols or to a sequence of states.

These two quantities are determined recursively. In fact, it is easy to verify that if, at instant k, a survivor sequence of states includes $\mathbf{s}_k = \boldsymbol{\sigma}_j$ then, at instant $k-1$, the same sequence includes $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_{i_{opt}}$, which is determined as follows:

$$\boldsymbol{\sigma}_{i_{opt}} = \arg \min_{\mathbf{s}_{k-1}=\boldsymbol{\sigma}_i \in \mathcal{S} \to \mathbf{s}_k = \boldsymbol{\sigma}_j} \Gamma(\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i) + |\rho_k - f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)|^2 \tag{8.208}$$

The term $|\rho_k - f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)|^2$ is called *branch metric*. Therefore, we obtain:

$$\Gamma(\mathbf{s}_k = \boldsymbol{\sigma}_j) = \Gamma(\mathbf{s}_{k-1} = \boldsymbol{\sigma}_{i_{opt}}) + |\rho_k - f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_{i_{opt}})|^2 \tag{8.209}$$

and the survivor sequence is augmented as follows:

$$\mathcal{L}(\mathbf{s}_k = \boldsymbol{\sigma}_j) = (\mathcal{L}(\mathbf{s}_{k-1} = \boldsymbol{\sigma}_{i_{opt}}), \boldsymbol{\sigma}_j) \tag{8.210}$$

Starting from $k = 0$, with initial state $\mathbf{s}_{-1}$, which may be known or arbitrary, the procedure is repeated until $k = K-1$. The optimum sequence of states is given by the survivor sequence $\mathcal{L}(\mathbf{s}_{K-1} = \boldsymbol{\sigma}_{j_{opt}})$ associated with $\mathbf{s}_{K-1} = \boldsymbol{\sigma}_{j_{opt}}$ having minimum cost.

If the state $\mathbf{s}_{-1}$ is known and equal to $\boldsymbol{\sigma}_{i_0}$, it is convenient to assign to the states $\mathbf{s}_{-1}$ the following costs:

$$\Gamma(\mathbf{s}_{-1} = \boldsymbol{\sigma}_i) = \begin{cases} 0 & \text{for } \boldsymbol{\sigma}_i = \boldsymbol{\sigma}_{i_0} \\ \infty & \text{otherwise} \end{cases} \tag{8.211}$$

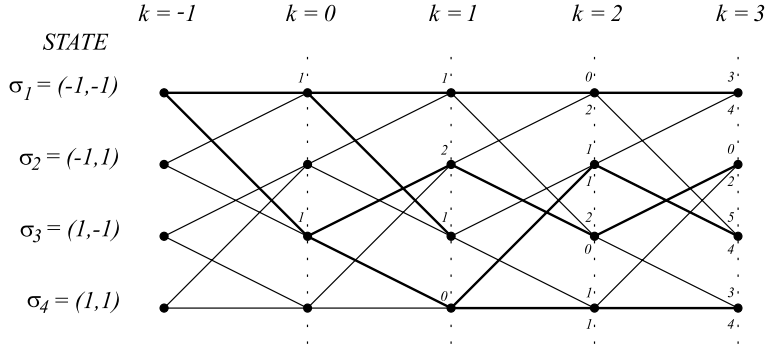
Analogously, if the final state $\mathbf{s}_K$ is equal to $\boldsymbol{\sigma}_{f_0}$, the optimum sequence of states coincides with the survivor sequence associated with the state $\mathbf{s}_{K-1} = \boldsymbol{\sigma}_j$ having minimum cost *among* those that admit a transition into $\mathbf{s}_K = \boldsymbol{\sigma}_{f_0}$.

Example 8.10.2

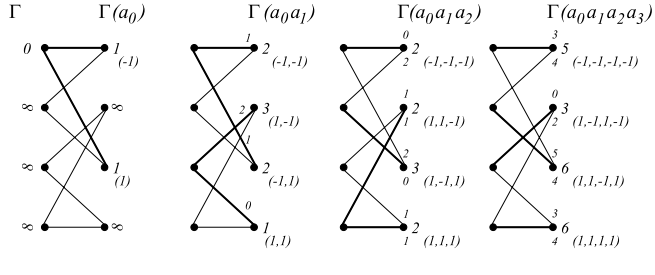
Let us consider a system with the following characteristics: $a_k \in \{-1, 1\}$, $L_1 = 0$, $L_2 = 2$, $K = 4$, and $\mathbf{s}_{-1} = (-1, -1)$. The development of the survivor sequences on the *trellis diagram* from $k = 0$ to $k = K - 1 = 3$ is represented in Figure 8.24a. The branch metric, $|\rho_k - f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)|^2$, associated with each transition is given in this example and is written above each branch. The survivor paths associated with each state are represented in bold; we note that some paths are abruptly interrupted and not extended at the following instant: for example the path ending at state $\boldsymbol{\sigma}_3$ at instant $k = 1$.

Figure 8.24b illustrates how the survivor sequences of Figure 8.24a are determined. Starting with $\mathbf{s}_{-1} = (-1, -1)$ we have

$$\begin{aligned} \Gamma(\mathbf{s}_{-1} = \boldsymbol{\sigma}_1) &= 0 \\ \Gamma(\mathbf{s}_{-1} = \boldsymbol{\sigma}_2) &= \infty \\ \Gamma(\mathbf{s}_{-1} = \boldsymbol{\sigma}_3) &= \infty \\ \Gamma(\mathbf{s}_{-1} = \boldsymbol{\sigma}_4) &= \infty \end{aligned} \tag{8.212}$$



(a) Trellis diagram from k = -1 to k = K-1 = 3



(b) Determination of the survivors

Figure 8.24. Trellis diagram and determination of the survivor sequences.

We apply (8.208) for $k = 0$; starting with $\mathbf{s}_0 = \boldsymbol{\sigma}_1$, we obtain

$$\begin{aligned}\Gamma(\mathbf{s}_0 = \boldsymbol{\sigma}_1) &= \min\{\Gamma(\mathbf{s}_{-1} = \boldsymbol{\sigma}_1) + 1, \Gamma(\mathbf{s}_{-1} = \boldsymbol{\sigma}_2) + 3\} \\ &= \min\{1, \infty\} \\ &= 1\end{aligned} \tag{8.213}$$

We observe that the result (8.213) is obtained for $\mathbf{s}_{-1} = \boldsymbol{\sigma}_1$. Then the survivor sequence associated with $\mathbf{s}_0 = \boldsymbol{\sigma}_1$ is $\mathcal{L}(\mathbf{s}_0 = \boldsymbol{\sigma}_1) = (-1)$, expressed as a sequence of symbols rather than states.

Considering now $\mathbf{s}_0 = \boldsymbol{\sigma}_2$, $\boldsymbol{\sigma}_3$, and $\boldsymbol{\sigma}_4$, in sequence, and applying (8.208), the first iteration is completed. Obviously, there is no interest in determining the survivor sequence for states with metric equal to ∞, because the corresponding path will not be extended.

Next, for $k = 1$, the final metrics and the survivor sequences are shown in the second diagram of Figure 8.24b, where we have

$$\Gamma(\mathbf{s}_1 = \boldsymbol{\sigma}_1) = \min\{1 + 1, \infty\} = 2 \tag{8.214}$$

$$\Gamma(\mathbf{s}_1 = \boldsymbol{\sigma}_2) = \min\{1 + 2, \infty\} = 3 \tag{8.215}$$

$$\Gamma(\mathbf{s}_1 = \boldsymbol{\sigma}_3) = \min\{1 + 1, \infty\} = 2 \tag{8.216}$$

$$\Gamma(\mathbf{s}_1 = \boldsymbol{\sigma}_4) = \min\{1 + 0, \infty\} = 1 \tag{8.217}$$

The same procedure is repeated for $k = 2, 3$, and the trellis diagram is completed. The minimum among the values assumed by $\Gamma(\mathbf{s}_3)$ is

$$\min_{i=1,\ldots,N_s} \Gamma(\mathbf{s}_3 = \boldsymbol{\sigma}_i) = \Gamma(\mathbf{s}_3 = \boldsymbol{\sigma}_2) = 3 \tag{8.218}$$

The associated optimum survivor sequence is

$$a_0 = 1 \qquad a_1 = -1 \qquad a_2 = 1 \qquad a_3 = -1 \tag{8.219}$$

In practice, if the parameter K is large, the length of the survivor sequences is limited to a value K_d, called *trellis depth* or *path memory depth*, that typically is between 3 and 10 times the length of the channel impulse response: this means that at every instant k we decide on $a_{k-K_d+L_1}$, a value that is then removed from the diagram. The decision is based on the survivor sequence associated with the minimum among the values of $\Gamma(\mathbf{s}_k)$.

In practice, k and $\Gamma(\mathbf{s}_k)$ may become very large; then the latter value is usually normalized by subtracting the same amount from all the metrics, for example the smallest of the metrics $\Gamma(\mathbf{s}_k = \boldsymbol{\sigma}_i)$, $i = 1, \ldots, N_s$.

Computational complexity of the VA

Memory. The memory to store the metrics $\Gamma(\mathbf{s}_k)$, $\Gamma(\mathbf{s}_{k-1})$ and the survivor sequences is proportional to the number of states N_s.

Computational complexity. The number of additions and comparisons is proportional to the number of transitions, $MN_s = M^{(L_1+L_2+1)}$. In any case, the complexity is linear in K.

We note that the values of $f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)$, that determine u_k in (8.199) can be memorized in a table, limited to the possible transitions $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i \rightarrow \mathbf{s}_k = \boldsymbol{\sigma}_j$, for $\boldsymbol{\sigma}_i, \boldsymbol{\sigma}_j \in \mathcal{S}$. However, at every instant k and for every transition $\boldsymbol{\sigma}_i \rightarrow \boldsymbol{\sigma}_j$, the branch metric $|\rho_k - f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)|^2$ needs to be evaluated: this computation, however, can be done outside the recursive algorithm.

8.10.2 Maximum *a posteriori* probability detector

We now discuss an efficient method to compute the *a posteriori* probability (APP) in (8.185) for an isolated symbol. Given a vector $\mathbf{z} = [z_0, z_1, \ldots, z_{K-1}]^T$, the notation $\mathbf{z}_\ell^m$ indicates the vector formed only by the components $[z_\ell, z_{\ell+1}, \ldots, z_m]^T$.

We introduce the *likelihood* function, defined as

$$\mathtt{L}_k(\beta) = P[a_{k+L_1} = \beta \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \qquad \beta \in \mathcal{A} \tag{8.220}$$

Then we have

$$\hat{a}_{k+L_1} = \arg\max_{\beta \in \mathcal{A}} \mathtt{L}_k(\beta) \qquad k = 0, 1, \ldots, K-1 \tag{8.221}$$

As for the VA, it is convenient to define the state

$$\mathbf{s}_k = (a_{k+L_1}, a_{k+L_1-1}, \ldots, a_{k-L_2+1}) \tag{8.222}$$

so that the desired signal at instant k, u_k, given by (8.174), turns out to be only a function of $\mathbf{s}_k$ and $\mathbf{s}_{k-1}$ (see (8.199)).

The formulation that we give for the solution of the problem (8.221) is called a *forward-backward algorithm* (FBA) and it follows the work by Rabiner [5]: it is seen that it coincides with the BCJR algorithm [6] for the decoding of convolutional codes.

We observe that, unlike in (8.222), in the two formulations [5, 6] the definition of state also includes the symbol a_{k-L_2} and consequently u_k is only a function of the state at the instant k.

Statistical description of a sequential machine

Briefly, we give a statistical description of the sequential machine associated with the state $\mathbf{s}_k$.

1. Let N_s be the number of values that $\mathbf{s}_k$ can assume. For a sequence of i.i.d. symbols $\{a_k\}$, we have $N_s = M^{L_1+L_2}$; as in (8.198), the values assumed by the state are denoted by $\boldsymbol{\sigma}_j$, $j = 1, \ldots, N_s$.

2. The sequence $\{\mathbf{s}_k\}$ is obtained by a time invariant sequential machine with transition probabilities

$$\Pi(j \mid i) = P[\mathbf{s}_k = \boldsymbol{\sigma}_j \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] \tag{8.223}$$

Now, if there is a transition from $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i$ to $\mathbf{s}_k = \boldsymbol{\sigma}_j$, determined by the symbol $a_{k+L_1} = \beta$, $\beta \in \mathcal{A}$, then

$$\Pi(j \mid i) = P[a_{k+L_1} = \beta] \tag{8.224}$$

which represents the *a priori* probability of the generic symbol. There are algorithms to iteratively estimate this probability from the output of another decoder or equalizer (see Section 11.5). Here, for the time being, we assume there is no *a priori* knowledge on the symbols. Consequently, for i.i.d. symbols, we have that every state $\mathbf{s}_k = \boldsymbol{\sigma}_j$ can be reached by M states, and

$$P[a_{k+L_1} = \beta] = \frac{1}{M} \qquad \beta \in \mathcal{A} \tag{8.225}$$

If there is no transition from $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i$ to $\mathbf{s}_k = \boldsymbol{\sigma}_j$, we set

$$\Pi(j \mid i) = 0 \tag{8.226}$$

3. The channel transition probabilities are given by

$$p_{z_k}(\rho_k \mid j, i) = P[z_k = \rho_k \mid \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] \tag{8.227}$$

assuming that there is a transition from $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i$ to $\mathbf{s}_k = \boldsymbol{\sigma}_j$. For a channel with complex-valued additive Gaussian noise, (8.188) holds, and

$$p_{z_k}(\rho_k \mid j, i) = \frac{1}{\pi \sigma_w^2} e^{-\frac{1}{\sigma_w^2}|\rho_k - u_k|^2} \tag{8.228}$$

where $u_k = f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)$.

4. We merge (8.223) and (8.227) by defining the variable

$$\begin{aligned} C_k(j \mid i) &= P[z_k = \rho_k, \mathbf{s}_k = \boldsymbol{\sigma}_j \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] \\ &= P[z_k = \rho_k \mid \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i]\, P[\mathbf{s}_k = \boldsymbol{\sigma}_j \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] \\ &= p_{z_k}(\rho_k \mid j, i)\, \Pi(j \mid i) \end{aligned} \tag{8.229}$$

5. Initial and final conditions are given by

$$\bar{p}_j = P[\mathbf{s}_{-1} = \boldsymbol{\sigma}_j] = \frac{1}{N_s} \qquad j = 1, \ldots, N_s \tag{8.230}$$

$$\bar{q}_j = P[\mathbf{s}_K = \boldsymbol{\sigma}_j] = \frac{1}{N_s} \qquad j = 1, \ldots, N_s \tag{8.231}$$

$$C_K(j \mid i) = \Pi(j \mid i) \qquad i, j = 1, \ldots, N_s \tag{8.232}$$

If the initial and/or final state are known, for example

$$\mathbf{s}_{-1} = \boldsymbol{\sigma}_{i_0} \qquad \boldsymbol{s}_K = \boldsymbol{\sigma}_{f_0} \tag{8.233}$$

we set

$$\bar{p}_j = \begin{cases} 1 & \text{for } \boldsymbol{\sigma}_j = \boldsymbol{\sigma}_{i_0} \\ 0 & \text{otherwise} \end{cases} \tag{8.234}$$

and

$$\bar{q}_j = \begin{cases} 1 & \text{for } \boldsymbol{\sigma}_j = \boldsymbol{\sigma}_{f_0} \\ 0 & \text{otherwise} \end{cases} \tag{8.235}$$

The forward-backward algorithm (FBA)

We consider the following four metrics.

a) Forward metric

$$F_k(j) = P[\mathbf{z}_0^k = \boldsymbol{\rho}_0^k, \mathbf{s}_k = \boldsymbol{\sigma}_j] \tag{8.236}$$

Equation (8.236) gives the probability of observing the sequence $\rho_0, \rho_1, \ldots, \rho_k$ up to instant k, and the state $\boldsymbol{\sigma}_j$ at instant k.

A recursive expression exists for $F_k(j)$.

1. Initialization

$$F_{-1}(j) = \bar{p}_j \qquad j = 1, \ldots, N_s \tag{8.237}$$

2. Updating for $k = 0, 1, \ldots, K-1$,

$$F_k(j) = \sum_{\ell=1}^{N_s} C_k(j \mid \ell)\, F_{k-1}(\ell) \qquad j = 1, \ldots, N_s \tag{8.238}$$

Proof. Using the total probability theorem, and conditioning the event on the possible values of $\boldsymbol{s}_{k-1}$, we express the probability in (8.236) as

$$\begin{aligned} F_k(j) &= \sum_{\ell=1}^{N_s} P[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}, z_k = \rho_k, \boldsymbol{s}_k = \boldsymbol{\sigma}_j, \boldsymbol{s}_{k-1} = \boldsymbol{\sigma}_\ell] \\ &= \sum_{\ell=1}^{N_s} P[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}, z_k = \rho_k \mid \boldsymbol{s}_k = \boldsymbol{\sigma}_j, \boldsymbol{s}_{k-1} = \boldsymbol{\sigma}_\ell]\, P[\boldsymbol{s}_k = \boldsymbol{\sigma}_j, \boldsymbol{s}_{k-1} = \boldsymbol{\sigma}_\ell] \end{aligned} \tag{8.239}$$

Because the noise samples are i.i.d., once the values of $\boldsymbol{s}_k$ and $\boldsymbol{s}_{k-1}$ are assigned, the event $[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}]$ is independent of the event $[z_k = \rho_k]$, and it results in

$$\begin{aligned} F_k(j) = \sum_{\ell=1}^{N_s} & P[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1} \mid \boldsymbol{s}_k = \boldsymbol{\sigma}_j, \boldsymbol{s}_{k-1} = \boldsymbol{\sigma}_\ell] \\ & P[z_k = \rho_k \mid \boldsymbol{s}_k = \boldsymbol{\sigma}_j, \boldsymbol{s}_{k-1} = \boldsymbol{\sigma}_\ell]\, P[\boldsymbol{s}_k = \boldsymbol{\sigma}_j, \boldsymbol{s}_{k-1} = \boldsymbol{\sigma}_\ell] \end{aligned} \tag{8.240}$$

Moreover, given s_{k-1}, the event $[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}]$ is independent of s_k, and we have

$$F_k(j) = \sum_{\ell=1}^{N_s} P[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1} \mid s_{k-1} = \boldsymbol{\sigma}_\ell] \\ P[z_k = \rho_k \mid s_k = \boldsymbol{\sigma}_j, s_{k-1} = \boldsymbol{\sigma}_\ell]\, P[s_k = \boldsymbol{\sigma}_j, s_{k-1} = \boldsymbol{\sigma}_\ell] \tag{8.241}$$

By applying Bayes' rule, (8.241) becomes

$$F_k(j) = \sum_{\ell=1}^{N_s} P[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}, s_{k-1} = \boldsymbol{\sigma}_\ell] \frac{1}{P[s_{k-1} = \boldsymbol{\sigma}_\ell]} P[z_k = \rho_k, s_k = \boldsymbol{\sigma}_j, s_{k-1} = \boldsymbol{\sigma}_\ell]$$

$$= \sum_{\ell=1}^{N_s} P[\mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}, s_{k-1} = \boldsymbol{\sigma}_\ell]\, P[z_k = \rho_k, s_k = \boldsymbol{\sigma}_j \mid s_{k-1} = \boldsymbol{\sigma}_\ell] \tag{8.242}$$

Substitution of (8.229) in (8.242) yields (8.238).

b) Backward metric

$$B_k(i) = P[\mathbf{z}_{k+1}^{K-1} = \boldsymbol{\rho}_{k+1}^{K-1} \mid \mathbf{s}_k = \boldsymbol{\sigma}_i] \tag{8.243}$$

Equation (8.243) is the probability of observing the sequence $\rho_{k+1}, \ldots, \rho_{K-1}$, from instant $k+1$ onwards, given the state $\boldsymbol{\sigma}_i$ at instant k.

A recursive expression also exists for $B_k(i)$.

1. Initialization

$$B_K(i) = \bar{q}_i \qquad i = 1, \ldots, N_s \tag{8.244}$$

2. Updating for $k = K-1, K-2, \ldots, 0$,

$$B_k(i) = \sum_{m=1}^{N_s} B_{k+1}(m)\, C_{k+1}(m \mid i) \qquad i = 1, \ldots, N_s \tag{8.245}$$

Proof. Using the total probability theorem, and conditioning the event on the possible values of s_{k+1}, we express the probability in (8.243) as

$$B_k(i) = \sum_{m=1}^{N_s} P[\mathbf{z}_{k+1}^{K-1} = \boldsymbol{\rho}_{k+1}^{K-1}, s_{k+1} = \boldsymbol{\sigma}_m \mid s_k = \boldsymbol{\sigma}_i] \\ = \sum_{m=1}^{N_s} P[\mathbf{z}_{k+1}^{K-1} = \boldsymbol{\rho}_{k+1}^{K-1} \mid s_{k+1} = \boldsymbol{\sigma}_m, s_k = \boldsymbol{\sigma}_i]\, P[s_{k+1} = \boldsymbol{\sigma}_m \mid s_k = \boldsymbol{\sigma}_i] \tag{8.246}$$

Now, given the values of $\boldsymbol{s}_{k+1}$ and $\boldsymbol{s}_k$, the event $[\mathbf{z}_{k+2}^{K-1} = \boldsymbol{\rho}_{k+2}^{K-1}]$ is independent of $[z_{k+1} = \rho_{k+1}]$. In turn, assigned the value of $\boldsymbol{s}_{k+1}$, the event $[\mathbf{z}_{k+2}^{K-1} = \boldsymbol{\rho}_{k+2}^{K-1}]$ is independent of $\boldsymbol{s}_k$. Then (8.246) becomes

$$\begin{aligned} B_k(i) &= \sum_{m=1}^{N_s} P[\mathbf{z}_{k+2}^{K-1} = \boldsymbol{\rho}_{k+2}^{K-1} \mid \boldsymbol{s}_{k+1} = \boldsymbol{\sigma}_m, \boldsymbol{s}_k = \boldsymbol{\sigma}_i] \\ &\qquad P[z_{k+1} = \rho_{k+1} \mid \boldsymbol{s}_{k+1} = \boldsymbol{\sigma}_m, \boldsymbol{s}_k = \boldsymbol{\sigma}_i]\, P[\boldsymbol{s}_{k+1} = \boldsymbol{\sigma}_m \mid \boldsymbol{s}_k = \boldsymbol{\sigma}_i] \\ &= \sum_{m=1}^{N_s} P[\mathbf{z}_{k+2}^{K-1} = \boldsymbol{\rho}_{k+2}^{K-1} \mid \boldsymbol{s}_{k+1} = \boldsymbol{\sigma}_m] \\ &\qquad P[z_{k+1} = \rho_{k+1} \mid \boldsymbol{s}_{k+1} = \boldsymbol{\sigma}_m, \boldsymbol{s}_k = \boldsymbol{\sigma}_i]\, P[\boldsymbol{s}_{k+1} = \boldsymbol{\sigma}_m \mid \boldsymbol{s}_k = \boldsymbol{\sigma}_i] \end{aligned} \tag{8.247}$$

Observing (8.243) and (8.229), (8.245) follows.

c) State metric

$$V_k(i) = P[\mathbf{s}_k = \boldsymbol{\sigma}_i \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \tag{8.248}$$

Equation (8.248) expresses the probability of being in the state $\boldsymbol{\sigma}_i$ at instant k, given the whole observation $\boldsymbol{\rho}_0^{K-1}$. It can be expressed as a function of the forward and backward metrics,

$$V_k(i) = \frac{F_k(i)\, B_k(i)}{\sum_{n=1}^{N_s} F_k(n)\, B_k(n)} \qquad i = 1, \ldots, N_s \tag{8.249}$$

Proof. Using the fact that, given the value of $\boldsymbol{s}_k$, the r.v.s $\{z_t\}$ with $t > k$ are statistically independent of $\{z_t\}$ with $t \leq k$, from (8.248) it follows

$$\begin{aligned} V_k(i) &= P[\mathbf{z}_0^k = \boldsymbol{\rho}_0^k, \mathbf{z}_{k+1}^{K-1} = \boldsymbol{\rho}_{k+1}^{K-1}, \boldsymbol{s}_k = \boldsymbol{\sigma}_i] \frac{1}{P[\mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}]} \\ &= P[\mathbf{z}_0^k = \boldsymbol{\rho}_0^k, \boldsymbol{s}_k = \boldsymbol{\sigma}_i]\, P[\mathbf{z}_{k+1}^{K-1} = \boldsymbol{\rho}_{k+1}^{K-1} \mid \boldsymbol{s}_k = \boldsymbol{\sigma}_i] \frac{1}{P[\mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}]} \end{aligned} \tag{8.250}$$

Observing the definitions of forward and backward metrics, (8.249) follows.

We note that the normalization factor

$$P[\mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] = \sum_{n=1}^{N_s} F_k(n)\, B_k(n) \tag{8.251}$$

makes $V_k(i)$ a probability, so that

$$\sum_{i=1}^{N_s} V_k(i) = 1 \tag{8.252}$$

d) Likelihood function of the generic symbol. Applying the total probability theorem to (8.220) we obtain the relation

$$\mathtt{L}_k(\beta) = \sum_{i=1}^{N_s} P[a_{k+L_1} = \beta, \mathbf{s}_k = \boldsymbol{\sigma}_i \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \tag{8.253}$$

From the comparison of (8.253) with (8.248), indicating with $[\boldsymbol{\sigma}_i]_m$, $m = 1, \ldots, L_1 + L_2$, the $m-$th component of the state $\boldsymbol{\sigma}_i$ (see (8.222)), we have

$$\mathtt{L}_k(\beta) = \sum_{\substack{i=1 \\ \text{condition} \\ [\boldsymbol{\sigma}_i]_1 = \beta}}^{N_s} V_k(i) \qquad \beta \in \mathcal{A} \tag{8.254}$$

In other words, at instant k the likelihood function coincides with the sum of the metrics $V_k(i)$ associated with the states whose first component is equal to the symbol of value β. Note that $\mathtt{L}_k(\beta)$ can also be obtained using the state metrics evaluated at different instants, that is

$$\mathtt{L}_k(\beta) = \sum_{\substack{i=1 \\ [\boldsymbol{\sigma}_i]_m = \beta}}^{N_s} V_{k+(m-1)}(i) \qquad \beta \in \mathcal{A} \tag{8.255}$$

for $m \in \{1, \ldots, L_1 + L_2\}$

Scaling

We see that, due to the exponential form of $p_{z_k}(\rho_k \mid j, i)$ in $C_k(j \mid i)$, in a few iterations the forward and backward metrics may assume very small values; this leads to numerical problems in the computation of the metrics: therefore we need to substitute equations (8.238) and (8.245) with analogous expressions that are scaled by a suitable coefficient.

We note that the state metric (8.249) does not change if we multiply $F_k(i)$ and $B_k(i)$, $i = 1, \ldots, N_s$, by the same coefficient $\mathcal{K}_k$. The idea [5] is to choose

$$\mathcal{K}_k = \frac{1}{\sum_{n=1}^{N_s} F_k(n)} \tag{8.256}$$

Indicating with $\bar{F}_k(i)$ and $\bar{B}_k(i)$ the normalized metrics, for

$$F_k(j) = \sum_{\ell=1}^{N_s} C_k(j \mid \ell)\, \bar{F}_{k-1}(\ell) \tag{8.257}$$

equation (8.238) becomes

$$\bar{F}_k(j) = \frac{\sum_{\ell=1}^{N_s} C_k(j \mid \ell)\, \bar{F}_{k-1}(\ell)}{\sum_{n=1}^{N_s} \sum_{\ell=1}^{N_s} C_k(n \mid \ell)\, \bar{F}_{k-1}(\ell)} \qquad \begin{matrix} j = 1, \ldots, N_s \\ k = 0, 1, \ldots, K-1 \end{matrix} \tag{8.258}$$

Correspondingly (8.245) becomes

$$\bar{B}_k(i) = \frac{\sum_{m=1}^{N_s} \bar{B}_{k+1}(m)\, C_{k+1}(m \mid i)}{\sum_{n=1}^{N_s} \sum_{\ell=1}^{N_s} C_k(n \mid \ell)\, \bar{F}_{k-1}(\ell)} \qquad \begin{matrix} i = 1, \ldots, N_s \\ k = K-1, K-2, \ldots, 0 \end{matrix} \tag{8.259}$$

Hence,

$$V_k(i) = \frac{\bar{F}_k(i)\, \bar{B}_k(i)}{\sum_{n=1}^{N_s} \bar{F}_k(n)\, \bar{B}_k(n)} \qquad \begin{matrix} i = 1, \ldots, N_s \\ k = 0, 1, \ldots, K-1 \end{matrix} \tag{8.260}$$

Likelihood function in the absence of ISI

In the absence of ISI, $u_k = a_k$. Therefore the state is expressed as $\mathbf{s}_k = (a_k)$, and is identified by the symbol value β. The metric in (8.250), for i.i.d. symbols and channel transition probabilities (8.228) becomes

$$\begin{aligned} V_k(\beta) &= P[z_k = \rho_k, \mathbf{s}_k = \sigma_i] \\ &= P[z_k = \rho_k, a_k = \beta] \\ &= K\, e^{-\frac{1}{\sigma_w^2}|\rho_k - \beta|^2} \qquad \beta \in \mathcal{A} \end{aligned} \tag{8.261}$$

where K is a constant. As $V_k(\ \beta)$ is a probability, (8.261) can be written as

$$V_k(\beta) = \frac{e^{-\frac{1}{\sigma_w^2}|\rho_k - \beta|^2}}{\sum_{\alpha \in \mathcal{A}} e^{-\frac{1}{\sigma_w^2}|\rho_k - \alpha|^2}} \qquad \beta \in \mathcal{A} \qquad k = 0, 1, \ldots, K-1 \tag{8.262}$$

Then (8.254) simply becomes

$$\mathtt{L}_k(\beta) = V_k(\beta) \qquad \beta \in \mathcal{A} \tag{8.263}$$

Simplified version of the MAP algorithm (Max-Log-MAP)

We introduce the Log-MAP criterion, which employs the logarithm of the variables $F_k(i)$, $B_k(i)$, $C_k(j \mid i)$, $V_k(i)$, and $\mathtt{L}_k(\beta)$. The logarithmic variables are indicated with the corresponding lower-case letters; in particular, from (8.249) we have

$$v_k(i) = \ln V_k(i) \tag{8.264}$$

and (8.254) becomes

$$\ell_k(\beta) = \ln \mathtt{L}_k(\beta) = \ln \left(\sum_{\substack{i=1 \\ [\boldsymbol{\sigma}_i]_1 = \beta}}^{N_s} e^{v_k(i)} \right) \qquad \beta \in \mathcal{A} \tag{8.265}$$

The function $\ell_k(\beta)$ is called *log-likelihood*.

The exponential emphasizes the difference between the metrics $v_k(i)$: typically a term dominates within each sum; this suggests the approximation

$$\ln \sum_{i=1}^{N_s} e^{v_k(i)} \simeq \max_{i \in \{1,\ldots,N_s\}} v_k(i) \tag{8.266}$$

Consequently (8.265) is *approximated* as

$$\tilde{\ell}_k(\beta) = \max_{\substack{i \in \{1,\ldots,N_s\} \\ [\boldsymbol{\sigma}_i]_1 = \beta}} v_k(i) \tag{8.267}$$

and the Log-MAP criterion is replaced by the Max-Log-MAP criterion

$$\hat{a}_{k+L_1} = \arg\max_{\beta \in \mathcal{A}} \tilde{\ell}_k(\beta) \tag{8.268}$$

Observation 8.12
In the particular case of absence of ISI, substitution of (8.263) in (8.265) yields

$$\ell_k(\beta) = -\frac{|\rho_k - \beta|^2}{\sigma_w^2} \tag{8.269}$$

and (8.268) becomes

$$\hat{a}_k = \arg\min_{\beta \in \mathcal{A}} |\rho_k - \beta|^2 \tag{8.270}$$

which corresponds to the minimum distance decision criterion.

Example 8.10.3 (Binary case)
In the binary case, $\beta \in \{-1, 1\}$, it is convenient to introduce the *likelihood ratio* given by (see (8.220))

$$\mathtt{L}_k = \frac{P[a_{k+L_1} = 1 \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}]}{P[a_{k+L_1} = -1 \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}]} = \frac{\mathtt{L}_k(1)}{\mathtt{L}_k(-1)} \tag{8.271}$$

Then the decision rule (8.221) becomes

$$\hat{a}_{k+L_1} = \begin{cases} 1 & \text{if } \mathtt{L}_k \geqslant 1 \\ -1 & \text{if } \mathtt{L}_k < 1 \end{cases} \tag{8.272}$$

We recall that in the absence of ISI the expression of $\mathtt{L}_k(\beta)$ is given by (8.262). The analysis is simplified by the introduction of the *log-likelihood ratio* (LLR),

$$\ell_k = \ln \mathtt{L}_k = \ell_k(1) - \ell_k(-1) \tag{8.273}$$

where $\ell_k(\beta)$ is given by (8.265). Then (8.272) becomes

$$\hat{a}_{k+L_1} = \operatorname{sgn}(\ell_k) \tag{8.274}$$

Observing (8.274), we can write

$$\ell_k = \hat{a}_{k+L_1} |\ell_k| \tag{8.275}$$

In other words, the sign of the log-likelihood ratio yields a hard-decision on the symbol being detected, while the magnitude indicates the reliability of the decision. The function ℓ_k is used as a soft-decision parameter in some detection algorithms (see page 923).

In the Max-Log-MAP formulation, the LLR is given by

$$\begin{aligned} \tilde{\ell}_k &= \tilde{\ell}_k(1) - \tilde{\ell}_k(-1) \\ &= \max_{\substack{i \in \{1,\ldots,N_s\} \\ [\boldsymbol{\sigma}_i]_1 = 1}} v_k(i) - \max_{\substack{i \in \{1,\ldots,N_s\} \\ [\boldsymbol{\sigma}_i]_1 = -1}} v_k(i) \end{aligned} \tag{8.276}$$

and

$$\hat{a}_{k+L_1} = \operatorname{sgn}(\tilde{\ell}_k) \tag{8.277}$$

Apart from non-essential constants, the Max-Log-MAP algorithm in the case of transmission of i.i.d. symbols over a channel with additive white Gaussian noise is formulated as follows.

1. *Computation of channel transition metrics*. For $k = 0, 1, \ldots, K-1$,

$$c_k(j \mid i) = -|\rho_k - u_k|^2 \qquad i, j = 1, \ldots, N_s \tag{8.278}$$

where $u_k = f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)$, assuming there is a transition between $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i$ and $\mathbf{s}_k = \boldsymbol{\sigma}_j$. For $k = K$, we let

$$c_K(j \mid i) = 0 \qquad i, j = 1, \ldots, N_s \tag{8.279}$$

again, assuming there is a transition between $\boldsymbol{\sigma}_i$ and $\boldsymbol{\sigma}_j$.

2. *Backward procedure*. For $k = K-1, K-2, \ldots, 0$,

$$\tilde{b}_k(i) = \max_{m \in \{1,\ldots,N_s\}} [\tilde{b}_{k+1}(m) + c_{k+1}(m \mid i)] \qquad i = 1, \ldots, N_s \tag{8.280}$$

If the final state is known, then

$$\tilde{b}_K(i) = \begin{cases} 0 & \boldsymbol{\sigma}_i = \boldsymbol{\sigma}_{f_0} \\ -\infty & \text{otherwise} \end{cases} \tag{8.281}$$

If the final state is unknown, we set $\tilde{b}_K(i) = 0$, $i = 1, \ldots, N_s$.

3. *Forward procedure*. For $k = 0, 1, \ldots, K-1$,

$$\tilde{f}_k(j) = \max_{\ell \in \{1,\ldots,N_s\}} [\tilde{f}_{k-1}(\ell) + c_k(j \mid \ell)] \qquad j = 1, \ldots, N_s \tag{8.282}$$

If the initial state is known, then

$$\tilde{f}_{-1}(j) = \begin{cases} 0 & \boldsymbol{\sigma}_j = \boldsymbol{\sigma}_{i_0} \\ -\infty & \text{otherwise} \end{cases} \tag{8.283}$$

If the initial state is unknown, we set $\tilde{f}_{-1}(j) = 0$, $j = 1, \ldots, N_s$.

4. *State metric*. For $k = 0, 1, \ldots, K-1$,

$$\tilde{v}_k(i) = \tilde{f}_k(i) + \tilde{b}_k(i) \qquad i = 1, \ldots, N_s \tag{8.284}$$

5. *Log-likelihood function of an isolated symbol*. For $k = 0, 1, \ldots, K-1$, the log-likelihood function is given by (8.267), with $\tilde{v}_k(i)$ in substitution of $v_k(i)$; the decision rule is given by (8.268).

In practical implementations of the algorithm, steps 3, 4, and 5 can be carried out in sequence for each value of k: this saves memory locations. To avoid overflow, for each value of k a common value can be added to all variables $\tilde{f}_k(i)$ and $\tilde{b}_k(i)$, $i = 1, \ldots, N_s$.

We observe that the two procedures, backward and forward, can be efficiently implemented by the Viterbi algorithm, using a trellis diagram run both in backward and forward directions. The simplified MAP algorithm requires about twice the complexity of the VA implementing the MLSD criterion. Memory requirements are considerably increased with respect to the VA, because the backward metrics must be stored before evaluating the state metrics. However, methods for an efficient use of the memory are proposed in [7].

Relation between Max-Log-MAP and Log-MAP

We define the following function of two variables [7]

$$\max^*(x, y) = \ln(e^x + e^y) \tag{8.285}$$

it can be verified that the following relation holds:

$$\max^*(x, y) = \max(x, y) + \ln(1 + e^{-|x-y|}) \tag{8.286}$$

We now extend the above definition to the case of three variables,

$$\max^*(x, y, z) = \ln(e^x + e^y + e^z) \tag{8.287}$$

then we have

$$\max^*(x, y, z) = \max^*(\max^*(x, y), z) \tag{8.288}$$

The extension to more variables is readily obtained by induction. So, if in the backward and forward procedures of page 676 we substitute the max function with the max* function, we obtain the exact Log-MAP formulation that relates $v_k(i) = \ln V_k(i)$ to $b_k(i) = \ln B_k(i)$ and $f_k(i) = \ln F_k(i)$, using the branch metric $c_k(j \mid i)$.

8.11 Optimum receivers for transmission over dispersive channels

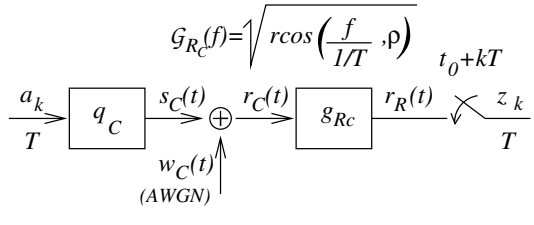
It is possible to identify two different receiver structures that supply the signal z_k given by (8.173).

1. The first, illustrated in Figure 8.25a, is considered for the low implementation complexity; it refers to the receiver of Figure 7.12, where

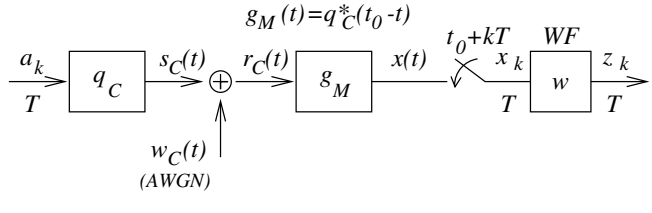
$$\mathcal{G}_{Rc}(f) = \sqrt{\text{rcos}\left(\frac{f}{1/T}, \rho\right)} \tag{8.289}$$

and $w_C(t)$ is white Gaussian noise with spectral density N_0. Recalling that $r_R(t) = s_R(t) + w_R(t)$, with

$$\mathcal{P}_{w_R}(f) = \mathcal{P}_w(f)\,|\mathcal{G}_{Rc}(f)|^2 = N_0\,\text{rcos}\left(\frac{f}{1/T}, \rho\right) \tag{8.290}$$



(a)



(b)

Figure 8.25. Two receiver structures with i.i.d. noise samples at the decision point.

it is important to verify that the noise sequence $\{w_k = w_R(t_0 + kT)\}$ has a constant spectral density equal to N_0, and the variance of the noise samples is $\sigma_w^2 = N_0/T$.

Although the filter defined by (8.289) does not necessarily yield a sufficient statistic (see Observation 8.13 on page 681), it considerably reduces the noise and this may be useful in estimating the channel impulse response. Another problem concerns the optimum timing phase, which may be difficult to determine for non-minimum phase channels.

2. An alternative, also known as the *Forney receiver*, is represented in Figure 8.20 and repeated in Figure 8.25b.

   To construct the WF, however, it is necessary to determine poles and zeros of the function $\Phi(z)$; this can be rather complicated in real time applications. A practical method is based on Observation 8.9 on page 650. From the knowledge of the autocorrelation sequence of the channel impulse response, the prediction error filter $A(z)$ is determined. The WF of Figure 8.25 is given by $W(z) = \frac{1}{f_0^2} A^*(1/z^*)$; therefore it is an FIR filter. The impulse response $\{\eta_n\}$ is given by the inverse z-transform of $\frac{1}{f_0} F(z) = 1/A(z)$. For the realization of symbol detection algorithms, a windowed version of the impulse response is considered.

   A further method, which usually requires a filter w with a smaller number of coefficients than the previous method, is based on the observation that, for channels with low noise level, the DFE solution determined by the MSE method coincides with the DFE-ZF solution; in this case the FF filter plays the role of the filter w in Figure 8.25b. We consider two cases.

   a. At the output of the MF g_M, let $\{h_n\}$ be the system impulse response with sampling period T, determined for example through the method described in the Observation 8.5 on page 641. Using (8.96), a DFE is designed with filter parameters (M_1, M_2), and consequently $\mathrm{w} = c_{opt}$ in Figure 8.25b. At the output of the filter w, the ideally minimum phase impulse response $\{\eta_n\}$ corresponds to the translated, by D sampling intervals, and windowed, with $L_1 = 0$ and $L_2 = M_2$, version of $\psi = h * c_{opt}$.

   b. If the impulse response of the system is unknown, we can resort to the FS-DFE structure of Figure 8.14. Using now an adaptive method to determine the coefficients of the DFE, at convergence we get

$$\eta_n = \psi_{n+D} \simeq \begin{cases} 1 & n = 0 \\ -b_n & n = 1, \ldots, M_2 \end{cases} \tag{8.291}$$

Actually, unless the length of $\{\eta_n\}$ is shorter than that of the impulse response q_C, it is convenient to use Ungerboeck's formulation of the MLSD [8] that utilizes only samples $\{x_k\}$ at the MF output; now, however, the metric is no longer Euclidean. The derivation of the non-Euclidean metric is the subject of the next section.

We note that, as it is not important to obtain the likelihood of an isolated symbol from the non-Euclidean metric, there are cases in which this method is not adequate. We refer in particular to the case in which decoding with soft input is performed separately from

symbol detection in the presence of ISI (see Section 11.3.2). However, joint decoding and detection are always possible using a suitable trellis (see Section 11.3.2).

Ungerboeck's formulation of the MLSD

We refer to the transmission of K symbols and to an observation interval $T_K = KT$ sufficiently large, so that the transient of filters at the beginning and at the end of the transmission has a negligible effect.

The derivation of the likelihood is based on the received signal (7.45),

$$r_C(t) = s_C(t) + w_C(t) \tag{8.292}$$

where w_C is white noise with PSD N_0, and

$$s_C(t) = \sum_{k=0}^{K-1} a_k \, q_C(t - kT) \tag{8.293}$$

For a suitable basis, we consider for (8.292) the following vector representation:

$$\mathbf{r} = \mathbf{s} + \mathbf{w}$$

Assuming that the transmitted symbol sequence $\mathbf{a} = [a_0, a_1, \ldots, a_{K-1}]$ is equal to $\boldsymbol{\alpha} = [\alpha_0, \alpha_1, \ldots, \alpha_{K-1}]$, the probability density function of $\mathbf{r}$ is given by (6.16),

$$p_{\mathbf{r}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) = K \exp\left(-\frac{1}{N_0} \|\boldsymbol{\rho} - \boldsymbol{s}\|^2\right) \tag{8.294}$$

Using (1.36), and observing $r_C(t) = \rho(t)$, we get

$$p_{\mathbf{r}|\mathbf{a}}(\boldsymbol{\rho} \mid \boldsymbol{\alpha}) = K \exp\left(-\frac{1}{N_0} \int_{T_K} |\rho(t) - s_C(t)|^2 \, dt\right) \tag{8.295}$$

Taking the logarithm in (8.295), the log-likelihood (to be maximized) is

$$\ell(\boldsymbol{\alpha}) = -\int_{T_K} \left|\rho(t) - \sum_{k=0}^{K-1} \alpha_k \, q_C(t - kT)\right|^2 dt \tag{8.296}$$

Correspondingly the detected sequence is given by

$$\hat{\mathbf{a}}_k = \arg\max_{\boldsymbol{\alpha}} \, \ell(\boldsymbol{\alpha}) \tag{8.297}$$

Expanding the squared term in (8.296), we obtain

$$\begin{aligned} \ell(\boldsymbol{\alpha}) = -\Bigg\{ & \int_{T_K} |\rho(t)|^2 \, dt - 2Re\left[\int_{T_K} \rho(t) \sum_{k=0}^{K-1} \alpha_k^* \, q_C^*(t - kT) \, dt\right] \\ & + \int_{T_K} \sum_{k_1=0}^{K-1} \sum_{k_2=0}^{K-1} \alpha_{k_1} \alpha_{k_2}^* \, q_C(t - k_1 T) \, q_C^*(t - k_2 T) \, dt \Bigg\} \end{aligned} \tag{8.298}$$

We now introduce the MF

$$g_M(t) = q_C^*(t_0 - t) \tag{8.299}$$

and the overall impulse response at the MF output

$$q(t) = (q_C * g_M)(t) = \mathtt{r}_{q_C}(t - t_0) \tag{8.300}$$

where $\mathtt{r}_{q_C}$ is the autocorrelation of q_C, whose samples are given by (see (8.23))

$$h_n = q(t_0 + nT) = \mathtt{r}_{q_C}(nT) \tag{8.301}$$

Let $x(t)$ be the MF output signal expressed as

$$x(t) = (r_C * g_M)(t) \tag{8.302}$$

with samples given by

$$x_k = x(t_0 + kT) \tag{8.303}$$

In (8.298) the first term can be ignored since it does not depend on $\boldsymbol{\alpha}$, while the other two terms are rewritten in the following form:

$$\ell(\boldsymbol{\alpha}) = -\left\{-2Re\left[\sum_{k=0}^{K-1} \alpha_k^* x_k\right] + \sum_{k_1=0}^{K-1}\sum_{k_2=0}^{K-1} \alpha_{k_1}\alpha_{k_2}^* h_{k_2-k_1}\right\} \tag{8.304}$$

Observing (8.304), we obtain the following important result.

Observation 8.13
The sequence of samples $\{x_k\}$, taken by sampling the MF output signal with sampling period equal to the symbol period T, forms a *sufficient statistic* to detect the message $\{a_k\}$ associated with the signal r_C defined in (8.292).

We express the double summation in (8.304) as the sum of three terms, the first for $k_1 = k_2$, the second for $k_1 < k_2$, and the third for $k_1 > k_2$:

$$\begin{aligned} A &= \sum_{k_1=0}^{K-1}\sum_{k_2=0}^{K-1} \alpha_{k_1}\alpha_{k_2}^* h_{k_2-k_1} \\ &= \sum_{k_1=0}^{K-1} \alpha_{k_1}\alpha_{k_1}^* h_0 + \sum_{k_1=1}^{K-1}\sum_{k_2=0}^{k_1-1} \alpha_{k_1}\alpha_{k_2}^* h_{k_2-k_1} + \sum_{k_2=1}^{K-1}\sum_{k_1=0}^{k_2-1} \alpha_{k_1}\alpha_{k_2}^* h_{k_2-k_1} \end{aligned} \tag{8.305}$$

Because the sequence $\{h_n\}$ is an autocorrelation, it enjoys the Hermitian property, i.e. $h_{-n} = h_n^*$; consequently, the third term in (8.305) is the complex conjugate of the second, and

$$A = \sum_{k=0}^{K-1} \alpha_k^*\alpha_k h_0 + 2Re\left[\sum_{k=1}^{K-1}\sum_{k_2=0}^{k-1} \alpha_k^*\alpha_{k_2} h_{k-k_2}\right] \tag{8.306}$$

By the change of indices $n = k - k_2$, assuming $\alpha_k = 0$ for $k < 0$, we get

$$\begin{aligned} A &= Re\left\{\sum_{k=0}^{K-1}\alpha_k^*\alpha_k h_0 + 2\left[\sum_{k=1}^{K-1}\sum_{n=1}^{k}\alpha_k^*\alpha_{k-n}h_n\right]\right\} \\ &= Re\left\{\sum_{k=0}^{K-1}\alpha_k^*\left[h_0\alpha_k + 2\sum_{n=1}^{k}h_n\alpha_{k-n}\right]\right\} \end{aligned} \tag{8.307}$$

In particular, if

$$|h_n| \simeq 0 \qquad \text{for } |n| > N_h \tag{8.308}$$

(8.307) is simplified in the following expression

$$A = Re\left\{\sum_{k=0}^{K-1}\alpha_k^*\left[h_0\alpha_k + 2\sum_{n=1}^{N_h}h_n\alpha_{k-n}\right]\right\} \tag{8.309}$$

Then the log-likelihood (8.304) becomes

$$\ell(\boldsymbol{\alpha}) = -\sum_{k=0}^{K-1} Re\left\{\alpha_k^*\left[-2x_k + h_0\alpha_k + 2\sum_{n=1}^{N_h}h_n\alpha_{k-n}\right]\right\} \tag{8.310}$$

To maximize $\ell(\boldsymbol{\alpha})$ or, equivalently, to minimize $-\ell(\boldsymbol{\alpha})$ with respect to $\boldsymbol{\alpha}$ we apply the Viterbi algorithm (see page 663) with the state vector defined as

$$\mathbf{s}_k = (a_k, a_{k-1}, \ldots, a_{k-N_h+1}) \tag{8.311}$$

and branch metric given by

$$Re\left\{a_k^*\left[-2x_k + h_0 a_k + 2\sum_{n=1}^{N_h}h_n a_{k-n}\right]\right\} \tag{8.312}$$

Extensions of Ungerboeck's approach to time variant radio channels are proposed in [9].

8.12 Error probability achieved by MLSD

In the Viterbi algorithm, we have an error if a sequence of states that is different from the correct sequence is chosen as maximum likelihood sequence in the trellis diagram; the probability that one or more states of the detected ML sequence are in error is interesting. The error probability is dominated by the probability that a sequence at the minimum Euclidean distance from the correct sequence is chosen as ML sequence. We note, however, that increasing the sequence length K also increases the number of different paths in the trellis diagram associated with sequences that are at the minimum distance. Therefore, by increasing K, the probability that the chosen sequence is in error usually tends to 1.

The probability that the whole sequence of states is not received correctly is rarely of interest; instead, we consider the probability that the detection of a generic symbol is in

error. For the purpose of determining the symbol error probability, the concept of *error event* is introduced. Let $\{\boldsymbol{\sigma}\} = (\boldsymbol{\sigma}_{i_0}, \ldots, \boldsymbol{\sigma}_{i_{K-1}})$ be the realization of the state sequence associated with the information sequence, and let $\{\hat{\boldsymbol{\sigma}}\}$ be the sequence chosen by the Viterbi algorithm. In a sufficiently long time interval, the paths in the trellis diagram associated with $\{\boldsymbol{\sigma}\}$ and $\{\hat{\boldsymbol{\sigma}}\}$ diverge and converge several times: every distinct separation from the correct path is called an error event.

Definition 8.1
An *error event* **e** is defined as a path in the trellis diagram that has only the initial and final states in common with the correct path; the length of an error event is equal to the number of nodes visited in the trellis before rejoining with the correct path.

Error events of length one and two are illustrated in a trellis diagram with two states, where the correct path is represented by a continuous line, in Figure 8.26a and Figure 8.26b, respectively.

Let $\mathtt{E}$ be the set of all error events beginning at instant i. Each element **e** of $\mathtt{E}$ is characterized by a correct path $\{\boldsymbol{\sigma}\}$ and a wrong path $\{\hat{\boldsymbol{\sigma}}\}$, which diverges from $\{\boldsymbol{\sigma}\}$ at instant i and converges at $\{\boldsymbol{\sigma}\}$ after a certain number of steps in the trellis diagram. We assume that the probability $P[\mathbf{e}]$ is independent of instant i: this hypothesis is verified with good approximation if the length of the trellis diagram is much greater than the length of the significant error events. An error event produces one or more errors in the detection of symbols of the input sequence. We have a *detection error* at instant k if the detection of the input at the k-th stage of the trellis diagram is not correct. We define the function [10]

$$c_m(\mathbf{e}) = \begin{cases} 1 & \text{if } \mathbf{e} \text{ causes a detection error at the instant } i+m \text{ , with } m \geqslant 0 \\ 0 & \text{otherwise} \end{cases} \tag{8.313}$$

The probability of a particular error event that starts at instant i and causes a detection error at instant k is given by $c_{k-i}(\mathbf{e})P[\mathbf{e}]$. Because the error events in $\mathtt{E}$ are disjointed, we have

$$P_e = P[\hat{a}_k \neq a_k] = \sum_{i=-\infty}^{k} \sum_{\mathbf{e}\in\mathtt{E}} c_{k-i}(\mathbf{e})\, P[\mathbf{e}] \tag{8.314}$$

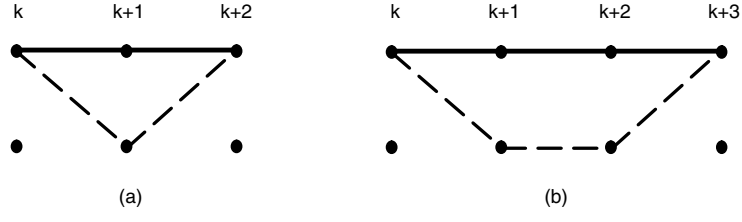


Figure 8.26. Error events of length (a) one and (b) two in a trellis diagram with two states.

Assuming that the two equations can be exchanged, we obtain

$$P_e = \sum_{\mathbf{e}\in\mathbb{E}} P[\mathbf{e}] \sum_{i=-\infty}^{k} c_{k-i}(\mathbf{e}) \tag{8.315}$$

With a change of variables it turns out

$$\sum_{i=-\infty}^{k} c_{k-i}(\mathbf{e}) = \sum_{m=0}^{\infty} c_m(\mathbf{e}) = N(\mathbf{e}) \tag{8.316}$$

which indicates the total number of detection errors caused by the error event $\mathbf{e}$. Therefore,

$$P_e = \sum_{\mathbf{e}\in\mathbb{E}} N(\mathbf{e}) P[\mathbf{e}] \tag{8.317}$$

where the dependence on the time index k vanishes. We therefore find that the detection error probability is equal to the average number of errors caused by all the possible error events initiating at a given instant i; this result is expected, because the detection error probability at a particular instant k must take into consideration all error events that initiate at previous instants and are not yet terminated.

If $\{\mathbf{s}\} = (\mathbf{s}_0, \ldots, \mathbf{s}_{K-1})$ denotes the random variable sequence of states at the transmitter and $\{\hat{s}\} = (\hat{s}_0, \ldots, \hat{s}_{K-1})$ denotes the random variable sequence of states selected by the ML receiver, the probability of an error event $\mathbf{e}$ beginning at a given instant i depends on the joint probability of the correct and incorrect path, and it can be written as

$$P[\mathbf{e}] = P[\{\hat{s}\} = \{\hat{\boldsymbol{\sigma}}\} \mid \{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] P[\{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] \tag{8.318}$$

Because it is usually difficult to find the exact expression for $P[\{\hat{s}\} = \{\hat{\boldsymbol{\sigma}}\} \mid \{\mathbf{s}\} = \{\boldsymbol{\sigma}\}]$, we resort to upper and lower limits.

Upper limit. Because detection of the sequence of states $\{\mathbf{s}\}$ is obtained by observing the sequence $\{\mathbf{u}\}$, for the signal in (8.173) with zero mean additive white Gaussian noise having variance σ_I^2 per dimension, we have the upper limit

$$P[\{\hat{s}\} = \{\hat{\boldsymbol{\sigma}}\} \mid \{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] \leq Q\left(\frac{d[\mathbf{u}(\{\boldsymbol{\sigma}\}), \mathbf{u}(\{\hat{\boldsymbol{\sigma}}\})]}{2\sigma_I}\right) \tag{8.319}$$

where $d[\mathbf{u}(\{\boldsymbol{\sigma}\}), \mathbf{u}(\{\hat{\boldsymbol{\sigma}}\})]$ is the Euclidean distance between signals $\mathbf{u}(\{\boldsymbol{\sigma}\})$ and $\mathbf{u}(\{\hat{\boldsymbol{\sigma}}\})$, given by (8.193). Substitution of the upper limit in (8.317) yields

$$P_e \leq \sum_{\mathbf{e}\in\mathbb{E}} N(\mathbf{e})\, P[\{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] Q\left(\frac{d[\mathbf{u}(\{\boldsymbol{\sigma}\}), \mathbf{u}(\{\hat{\boldsymbol{\sigma}}\})]}{2\sigma_I}\right) \tag{8.320}$$

which can be rewritten as follows, by giving prominence to the more significant terms,

$$P_e \leq \sum_{\mathbf{e}\in\mathbb{E}_{min}} N(\mathbf{e})\, P[\{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] Q\left(\frac{d_{min}}{2\sigma_I}\right) + \text{other terms} \tag{8.321}$$

where $\mathbb{E}_{min}$ is the set of error events at minimum distance d_{min} defined in (8.194), and the remaining terms are characterized by arguments of the Q function larger than $d_{min}/(2\sigma_I)$. For higher values of the signal-to-noise ratio these terms are negligible and the following approximation holds

$$P_e \leq \mathcal{K}_1 \, Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{8.322}$$

where

$$\mathcal{K}_1 = \sum_{\mathbf{e}\in\mathbb{E}_{min}} N(\mathbf{e}) \, P[\{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] \tag{8.323}$$

Lower limit. A lower limit to the error probability is obtained by considering the probability that any error event may occur rather than the probability of a particular error event. Since $N(\mathbf{e}) \geq 1$ for all the error events $\mathbf{e}$, from (8.317) we have

$$P_e \geq \sum_{\mathbf{e}\in\mathbb{E}} P[\mathbf{e}] \tag{8.324}$$

Let us consider a particular path in the trellis diagram determined by the sequence of states $\{\boldsymbol{\sigma}\}$. We set

$$d_{min}(\{\boldsymbol{\sigma}\}) = \min_{\{\tilde{\boldsymbol{\sigma}}\}} d[\boldsymbol{u}(\{\boldsymbol{\sigma}\}), \boldsymbol{u}(\{\tilde{\boldsymbol{\sigma}}\})] \tag{8.325}$$

i.e., for this path, $d_{min}(\{\boldsymbol{\sigma}\})$ is the Euclidean distance of the minimum distance error event. We have $d_{min}(\{\boldsymbol{\sigma}\}) \geq d_{min}$, where d_{min} is the minimum distance obtained considering all the possible state sequences. If $\{\boldsymbol{\sigma}\}$ is the correct state sequence, the probability of an error event is lower limited by

$$P[\mathbf{e} \mid \{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] \geq Q\left(\frac{d_{min}(\{\boldsymbol{\sigma}\})}{2\sigma_I}\right) \tag{8.326}$$

Consequently,

$$P_e \geq \sum_{\{\boldsymbol{\sigma}\}} P[\{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] Q\left(\frac{d_{min}(\{\boldsymbol{\sigma}\})}{2\sigma_I}\right) \tag{8.327}$$

If some terms are omitted in the equation, the lower limit is still valid, because the terms are non-negative. Therefore, taking into consideration only those state sequences $\{\boldsymbol{\sigma}\}$ for which $d_{min}(\{\boldsymbol{\sigma}\}) = d_{min}$, we obtain

$$P_e \geq \sum_{\{\boldsymbol{\sigma}\}\in A} P[\{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{8.328}$$

where A is the set of state sequences that admit an error event with minimum distance d_{min}, for an arbitrarily chosen initial instant of the given error event. Defining

$$\mathcal{K}_2 = \sum_{\{\boldsymbol{\sigma}\}\in A} P[\{\mathbf{s}\} = \{\boldsymbol{\sigma}\}] \tag{8.329}$$

as the probability that a path $\{\boldsymbol{\sigma}\}$ admits an error event with minimum distance, it is

$$P_e \geq \mathcal{K}_2 \, Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{8.330}$$

Combining upper and lower limits we obtain

$$\mathcal{K}_2 \, Q\left(\frac{d_{min}}{2\sigma_I}\right) \leq P_e \leq \mathcal{K}_1 \, Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{8.331}$$

For large values of the signal-to-noise ratio, therefore we have

$$P_e \simeq \mathcal{K} \, Q\left(\frac{d_{min}}{2\sigma_I}\right) \tag{8.332}$$

for some value of the constant $\mathcal{K}$ between $\mathcal{K}_1$ and $\mathcal{K}_2$.

We stress that the error probability, expressed by (8.332) and (8.195), is determined by the ratio between the minimum distance d_{min} and the standard deviation of the noise σ_I.

Here the expressions of the constants $\mathcal{K}_1$ and $\mathcal{K}_2$ are obtained by resorting to various approximations. An accurate method to calculate upper and lower limits of the error probability is proposed in [11].

Computation of the minimum distance

The application of the Viterbi algorithm to maximum likelihood sequence detection in transmission systems with ISI requires that the overall impulse response is FIR, otherwise the number of states, and hence also the complexity of the detector, becomes infinite. From (8.173), the samples at the detector input, conditioned on the event that the sequence of symbols $\{a_k\}$ is transmitted, are statistically independent Gaussian random variables with mean

$$\sum_{n=-L_1}^{L_2} \eta_n \, a_{k-n} \tag{8.333}$$

and variance σ_I^2 per dimension. The metric that the Viterbi algorithm attributes to the sequence of states corresponding to the sequence of input symbols $\{a_k\}$ is given by the squared Euclidean distance between the sequence of samples $\{z_k\}$ at the detector input and its mean value, which is known, given the sequence of symbols (see (8.189)),

$$\sum_{k=0}^{\infty} \left| z_k - \sum_{n=-L_1}^{L_2} \eta_n \, a_{k-n} \right|^2 \tag{8.334}$$

In the previous section it was demonstrated that the symbol error probability is given by (8.332). In particularly simple cases, the minimum distance can be determined by direct inspection of the trellis diagram; in practice, however, this situation is rarely verified in channels with ISI. To evaluate the minimum distance it is necessary to resort to simulations. To find the minimum distance error event with initial instant $k = 0$, we consider the

desired signal u_k under the condition that the sequence $\{a_k\}$ is transmitted, and we compute the squared Euclidean distance between this signal and the signal obtained for another sequence $\{\tilde{a}_k\}$,

$$d^2[\mathbf{u}(\{a_k\}), \mathbf{u}(\{\tilde{a}_k\})] = \sum_{k=0}^{\infty} \left| \sum_{n=-L_1}^{L_2} \eta_n \, a_{k-n} - \sum_{n=-L_1}^{L_2} \eta_n \, \tilde{a}_{k-n} \right|^2 \tag{8.335}$$

where it is assumed that the two paths identifying the state sequences are identical for $k < 0$.

It is possible to avoid computing the minimum distance for each sequence $\{a_k\}$ if we exploit the linearity of the ISI. Defining

$$\epsilon_k = a_k - \tilde{a}_k \tag{8.336}$$

we have

$$d^2(\{\epsilon_k\}) = d^2[\mathbf{u}(\{a_k\}), \mathbf{u}(\{\tilde{a}_k\})] = \sum_{k=0}^{\infty} \left| \sum_{n=-L_1}^{L_2} \eta_n \, \epsilon_{k-n} \right|^2 \tag{8.337}$$

The minimum among the squared Euclidean distances relative to all error events that initiate at $k = 0$ is

$$d^2_{min} = \min_{\{\epsilon_k\}: \epsilon_k = 0,\, k < L_1, \epsilon_{L_1} \neq 0} d^2(\{\epsilon_k\}) \tag{8.338}$$

It is convenient to solve this minimization without referring to the symbol sequences. In particular, we define the state $\mathbf{s}_k = (\epsilon_{k+L_1}, \epsilon_{k+L_1-1}, \ldots, \epsilon_{k-L_2+1})$, and a trellis diagram that describes the development of this state. Adopting the branch metric

$$\left| \sum_{n=-L_1}^{L_2} \eta_n \, \epsilon_{k-n} \right|^2 \tag{8.339}$$

the minimization problem is equivalent to determining the path in a trellis diagram that has minimum metric (8.337) and differs from the path that joins states corresponding to correct decisions: the resulting metric is d^2_{min}. We note, however, that the cardinality of ϵ_k is larger than M, and this implies that the complexity of this trellis diagram can be much larger than that of the original trellis diagram. In the PAM case, the cardinality of ϵ_k is equal to $2M - 1$.

In practice, as the terms of the series in (8.337) are non-negative, if we truncate the series after a finite number of terms we obtain a result that is smaller than or equal to the effective value. Therefore a lower limit to the minimum distance is given by

$$d^2_{min} \geq \min_{\{\epsilon_k\}: \epsilon_k = 0,\, k < L_1, \epsilon_{L_1} \neq 0} \sum_{k=0}^{K-1} \left| \sum_{n=-L_1}^{L_2} \eta_n \, \epsilon_{k-n} \right|^2 \tag{8.340}$$

Example 8.12.1

We consider the *partial response* system class IV (PR-IV), also known as *modified duobinary* (see Appendix 7.A), illustrated in Figure 8.27. The transfer function of the discrete-time

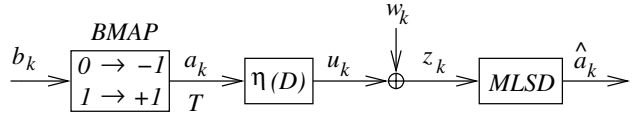


Figure 8.27. PR-IV (modified duobinary) transmission system.

b_k	...	0	1	1	0	1	0	0	...
a_k	...	−1	+1	+1	−1	+1	−1	−1	...
$u_k = a_k - a_{k-2}$	...		+2	−2	0	0	−2		...

Figure 8.28. Input and output sequences for an ideal PR-IV system.

overall system is given by $\eta(D) = 1 - D^2$. For an ideal noiseless system, the input sequence $\{u_k\}$ to the detector is formed by random variables taking values in the set $\{-2, 0, +2\}$, as shown in Figure 8.28.

Assuming that the sequence of noise samples $\{w_k\}$ is composed of real-valued, statistically independent, Gaussian random variables with mean zero and variance σ_I^2, and observing that u_k for k even (odd) depends only on symbols with even (odd) indices, the MLSD receiver for a PR-IV system is usually implemented by considering two interlaced dicode independent channels, each having a transfer function given by $1 - D$. As seen in Figure 8.29, for detection of the two interlaced input symbol sequences, two trellis diagrams are used. The state at instant k is given by the symbol $s_k = a_k$, where k is

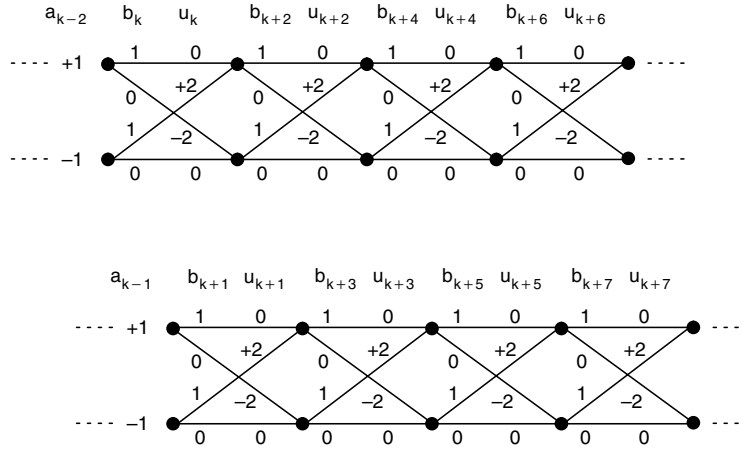


Figure 8.29. Trellis diagrams for detection of interlaced sequences.

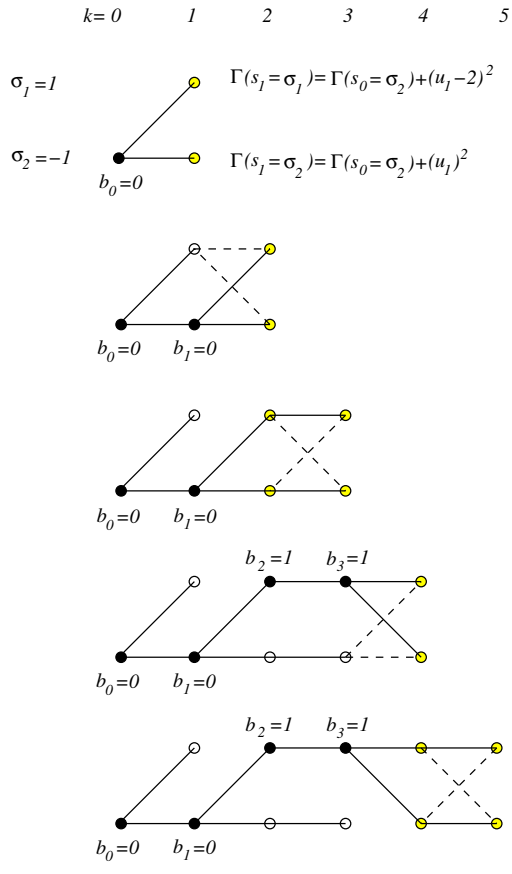


Figure 8.30. Survivor sequences at successive iterations of the Viterbi algorithm for a dicode channel.

even-valued in one of the two diagrams and odd-valued in the other. Each branch of the diagram is marked with a label that represents the binary input symbol b_k or, equivalently, the value of the dicode signal u_k. For a particular realization of the output signal of a dicode channel, the two survivor sequences at successive iterations of the Viterbi algorithm are represented in Figure 8.30.

It is seen that the minimum squared Euclidean distance between two separate paths in the trellis diagram is given by $d^2_{min} = 2^2 + 2^2 = 8$. However, we note that for the same initial instant, there are an infinite number of error events with minimum distance from the effective path: this fact is evident in the trellis diagram of Figure 8.31a, where the state $\mathbf{s}_k = (\epsilon_k) \in \{-2, 0, +2\}$ characterizes the development of the error event, and the labels indicate the branch metrics associated with an error event. It is seen that at every instant an error event may be extended along a path, for which the metric is equal to zero, parallel

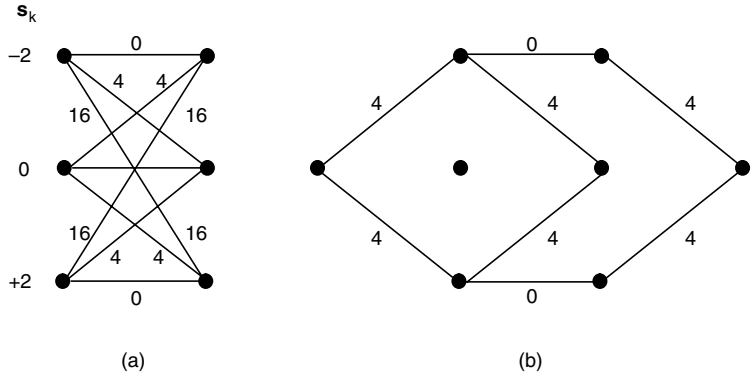


Figure 8.31. Examples of (a) trellis diagram to compute the minimum distance for a dicode channel and (b) four error events with minimum distance.

to the path corresponding to the zero sequence. Paths of this type correspond to a sequence of errors having the same polarity. Four error events with minimum distance are shown in Figure 8.31b.

The error probability is given by $\mathcal{K}Q\left(\frac{\sqrt{8}}{2\sigma_I}\right)$, where $\mathcal{K}_2 \leq \mathcal{K} \leq \mathcal{K}_1$. The constant $\mathcal{K}_2$ can be immediately determined by noting that every effective path admits at every instant at least one error event with minimum distance: consequently $\mathcal{K}_2 = 1$. To find $\mathcal{K}_1$, we consider the contribution of an error event with m consecutive errors. For this event to occur, it is required that m consecutive input symbols have the same polarity, which happens with probability 2^{-m}. Since such an error event determines m symbol errors and two error events with identical characteristics can be identified in the trellis diagram, we have

$$\mathcal{K}_1 = 2\sum_{m=1}^{\infty} m\, 2^{-m} = 4 \tag{8.341}$$

Besides the error events associated with the minimum distance $d_{min} = \sqrt{8}$, the occurrence of long sequences of identical output symbols from a dicode channel raises two problems.

- The system becomes catastrophic in the sense that in the trellis diagram used by the detector, valid sequences of arbitrary length are found having squared Euclidean distance from the effective path equal to 4; therefore, a MLSD receiver with finite memory will make additional errors with probability proportional to $Q\left(\frac{\sqrt{4}}{2\sigma_I}\right)$ if the channel produces sequences $\{u_k = 0\}$ of length larger than the memory of the detector.

- The occurrence of these sequences is detrimental for the control of receive filter gain and sampling instants: the problem is solved by suitable coding of the input binary sequence, that sets a limit on the number of consecutive identical symbols that are allowed at the channel input.

8.13 Reduced state sequence detection

For transmission over channels characterized by strong ISI, maximum likelihood detection implemented by the Viterbi algorithm typically gives better performance than a decision-feedback equalizer. On the other hand, if we indicate with $N = L_1 + L_2$ the length of the channel memory and with M the cardinality of the alphabet of transmitted symbols, the implementation complexity of the VA is of the order of M^N, which can be too large for practical applications if N and/or M assume large values.

To find receiver structures that are characterized by performance similar to that of a MLSD receiver, but lower complexity, two directions may be taken:

- decrease N;
- decrease the number of paths considered in the trellis;

The first direction leads to the application of pre-processing techniques, e.g., LE or DFE [12], to reduce the length of the channel impulse response. However, they only partially solve the problem: for bandwidth efficient modulation systems that utilize a large set of signals, the minimization of N is often insufficient to significantly reduce the complexity.

The second direction leads to further study of the MLSD method. An interesting algorithm is the $\mathcal{M}$-algorithm [13, 14]: it operates in the same way as the MLSD, i.e. using the full trellis diagram, but at every step it takes into account only $\mathcal{M} \leq M^N$ states, that is, those associated with paths with smaller metrics. The performance of the $\mathcal{M}$-algorithm is close to that of the MLSD even for small values of $\mathcal{M}$, however, it works only if the channel impulse response is minimum phase: otherwise, it is likely that among the paths that are eliminated in the trellis diagram the optimum path is also included.

The *reduced state sequence estimator* (RSSE)[4] [15, 16] is capable of yielding performance very close to that of MLSD, with significantly reduced complexity even though it retains the fundamental structure of MLSD. It can be used for modulation with a large symbol alphabet $\mathcal{A}$ and/or for channels with a long impulse response. The basic idea consists in using a trellis diagram with a reduced number of states, obtained by combining the states of the ML trellis diagram in a manner suggested by the principle of *partitioning* the set $\mathcal{A}$ (see Chapter 12), and possibly including the decision-feedback method in the computation of the branch metrics. In this way the RSSE guarantees a performance/complexity trade-off that can vary from that characterizing a DFE zero-forcing to that characterizing MLSD.

Reduced state trellis diagram

We consider the transmission system depicted in Figure 8.25b. Contrary to an MLSD receiver, the performance of an RSSE receiver may be poor if the overall channel impulse response is not minimum phase: to underline this fact, we slightly change the notation adopted in Section 8.10. We indicate with $\{f_1, f_2, \ldots, f_N\}$ the coefficients of the impulse

[4] We will maintain the name RSSE, although the algorithm is applied to perform a *detection* rather than an *estimation*.

response that determine the ISI and assume, without loss of generality, the desired sample is $f_0 = 1$. Hence the observed signal is given by

$$z_k = \sum_{n=0}^{N} f_n a_{k-n} + w_k = a_k + \langle \mathbf{s}_{k-1}, \mathbf{f} \rangle + w_k \tag{8.342}$$

where the state at instant $k-1$ is given by

$$\mathbf{s}_{k-1}^T = [a_{k-1}, a_{k-2}, \ldots, a_{k-N}] \tag{8.343}$$

and

$$\mathbf{f}^T = [f_1, f_2, \ldots, f_N] \tag{8.344}$$

Observation 8.14
In the transmission of sequences of blocks of data, the RSSE yields best performance by imposing the final state, for example, using the knowledge of a training sequence. Therefore the formulation of this section is suited for the case of a training sequence placed at the end of the data block. In the case the training sequence is placed at the beginning of a data block, it is better to process the signals in backward mode as described in Observation 8.6 on page 642.

The RSSE maintains the fundamental structure of MLSD unaltered, corresponding to the search in the trellis diagram of the path with minimum cost.

To reduce the number of states to be considered, we introduce for every component a_{k-n}, $n = 1, \ldots, N$, of the vector $\mathbf{s}_{k-1}$ defined in (8.343) a suitable partition $\Omega(n)$ of the two-dimensional set $\mathcal{A}$ of possible values of a_{k-n}: a partition is composed of J_n subsets, with J_n an integer between 1 and M. The index of the subset of the partition $\Omega(n)$ to which the symbol a_{k-n} belongs is indicated by c_n, an integer value between 0 and $J_n - 1$. Ungerboeck's partitioning of the symbol set associated with a 16-QAM system is illustrated in Figure 8.32. The partitions must satisfy the following two conditions (see also Chapter 12, or, for a more general partitioning method, [17]):

1. the numbers J_n are non-increasing, that is $J_1 \geq J_2 \geq \cdots \geq J_N$;
2. the partition $\Omega(n)$ is obtained by subdividing the subsets that make up the partition $\Omega(n+1)$, for every n between 1 and $N-1$.

Therefore we define as *reduced state at instant* $k-1$ the vector $\mathbf{t}_{k-1}$ that has as n-th element the index c_n of the subset of the partition $\Omega(n)$ to which the n-th element of $\mathbf{s}_{k-1}$ belongs, for $n = 1, 2, \ldots, N$, that is

$$\mathbf{t}_{k-1}^T = [c_1, c_2, \ldots, c_N] \qquad c_n \in \{0, 1, \ldots, J_n - 1\} \tag{8.345}$$

and we write

$$\mathbf{s}_{k-1} = \mathbf{s}(\mathbf{t}_{k-1}) \tag{8.346}$$

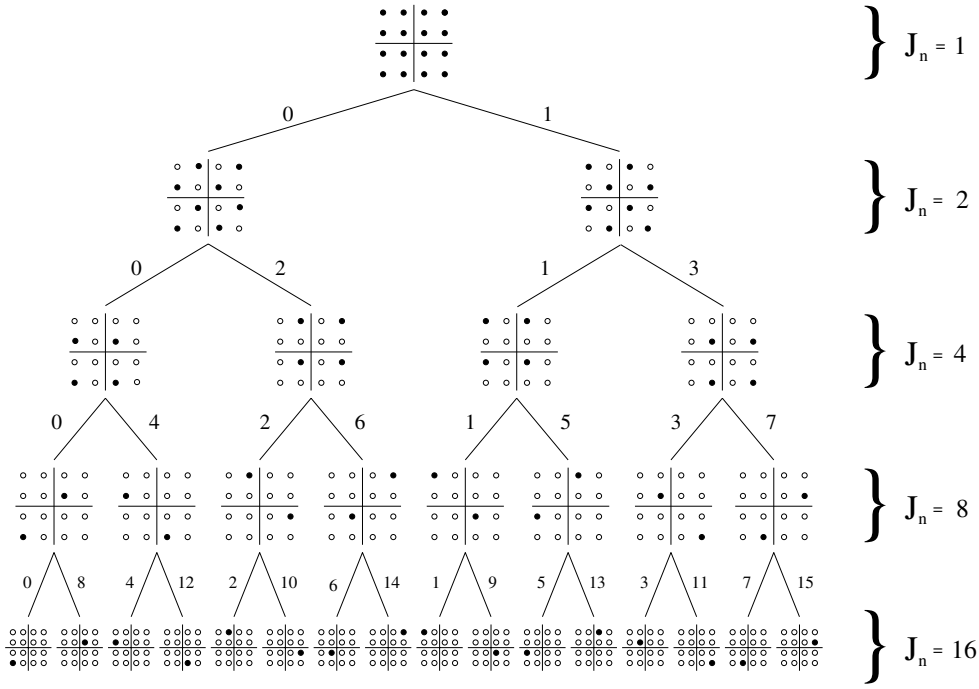


Figure 8.32. Ungerboeck's partitioning of the symbol set associated with a 16-QAM system. The various subsets are identified by the value of $c_n \in \{0, 1, \ldots, J_n - 1\}$.

It is useful to stress that the reduced state $\mathbf{t}_{k-1}$ does not uniquely identify a state $\mathbf{s}_{k-1}$, but all the states $\mathbf{s}_{k-1}$ that include as n-th element one of the symbols belonging to the subset c_n of partition $\Omega(n)$.

The conditions imposed on the partitions guarantee that, given a reduced state at instant $k-1$, $\mathbf{t}_{k-1}$, and the subset j of partition $\Omega(1)$ to which the symbol a_k belongs, the reduced state at instant k, $\mathbf{t}_k$, can be uniquely determined. In fact, observing (8.345), we have

$$\mathbf{t}_k^T = [c'_1, c'_2, \ldots, c'_N] \tag{8.347}$$

where $c'_1 = j$, c'_2 is the index of the subset of the partition $\Omega(2)$ to which belongs the subset with index c_1 of the partition $\Omega(1)$, c'_3 is the index of the subset of the partition $\Omega(3)$ to which belongs the subset with index c_2 of the partition $\Omega(2)$, and so forth. In this way the reduced states $\mathbf{t}_{k-1}$ define a proper *reduced state trellis diagram*, that represents all the possible sequences $\{a_k\}$.

As the symbol c_n can only assume one of the integer values between 0 and $J_n - 1$, the total number of possible reduced states of the trellis diagram of the RSSE is given by the product $N_s = J_1 J_2 \ldots J_N$, with $J_n \leq M$, for $n = 1, 2, \ldots, N$.

We know that in the VA, for uncoded transmission of i.i.d. symbols, there are M possible transitions from a state, one for each of the values that a_k can assume. In the reduced state trellis diagram M transitions are still possible from a state, however, to only J_1 distinct

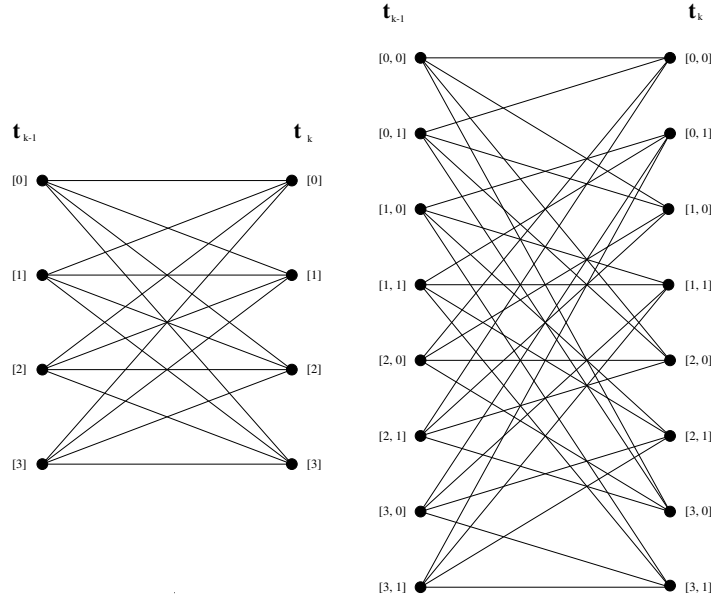
states, thus giving origin to parallel transitions.[5] In fact, if $J_1 < M$, J_1 sets of branches depart from every state $\mathbf{t}_{k-1}$, each set consisting of as many parallel transitions as there are symbols belonging to the subset of $\Omega(1)$ associated with the reduced state.

Therefore partitions must be obtained such that two effects are guaranteed: 1) minimum performance degradation with respect to MLSD, and 2) easy search of the optimum path among the various parallel transitions. The method that is usually adopted is Ungerboeck's set partitioning method, which, for every partition Ω, maximizes the minimum distance Δ among the symbols belonging to the same subset. For QAM systems and J_n a power of 2, the maximum distance Δ_n relative to partition $\Omega(n)$ is obtained through a tree diagram with binary partitioning (see Chapter 12). An example of partitioning of the symbol set associated with a 16-QAM system is illustrated in Figure 8.32.

In Figure 8.33 two examples of reduced state trellis diagram are shown, both referring to the partition of Figure 8.32.

RSSE algorithm

As in MLSD, for each transition that originates from a state $\mathbf{t}_{k-1}$ the RSSE computes the branch metric according to the expression



(a) $N = 1$, $J_1 = 4$. (b) $N = 2$, $J_1 = 4$, $J_2 = 2$.

Figure 8.33. Reduced state trellis diagrams.

[5] Parallel transitions are present when two or more branches connect the same pair of states in the trellis diagram.

$$|z_k - a_k - \langle \mathbf{s}_{k-1}, \mathbf{f} \rangle|^2 = |z_k - a_k - \langle \mathbf{s}(\mathbf{t}_{k-1}), \mathbf{f} \rangle|^2 \tag{8.348}$$

However, whereas in MLSD a survivor sequence may be described in terms of the *sequence of states* that led to a certain state, in the RSSE, if $J_1 < M$, it is convenient to memorize a *survivor sequence* as a *sequence of symbols*. As a matter of fact there is no one-to-one correspondence between state sequences and symbol sequences. Therefore backward tracing the optimum path in the trellis diagram in terms of the sequence of states does not univocally establish the optimum sequence of symbols.

Moreover, if $J_1 < M$, and therefore there are parallel transitions in the trellis diagram, for every branch the RSSE selects the symbol a_k in the subset of the partition $\Omega(1)$ that yields the minimum metric.[6] Thus the RSSE already makes a decision in selecting one of the parallel transitions, using also the past decisions memorized in the survivor sequence associated with the considered state.

At every iteration these decisions reduce the number of possible extensions of the N_s states from $N_s M$ to $N_s J_1$. This number is further reduced to N_s by selecting, for each state $\mathbf{t}_k$, the path with the minimum metric among the J_1 entering paths, operation which requires $N_s(J_1 - 1)$ comparisons. As in the case of the VA, "final" decisions are taken with a certain delay by tracing the history of the path with lowest metric.

We note that if $J_n = 1$, $n = 1, \ldots, N$, then the RSSE becomes a DFE-ZF, and if $J_n = M$, $n = 1, \ldots, N$, the RSSE performs full MLSD. Therefore the choice of $\{J_n\}$ determines a trade-off between performance and computational complexity.

The error probability for an RSSE is rather difficult to evaluate because of the presence of the decision-feedback mechanism. For the analysis of RSSE performance we refer the reader to [18].

Further simplification: DFSE

There are many applications in which the complexity of MLSD is mainly due to the length of the channel impulse response. In these cases a method to reduce the number of states consists of cancelling part of the ISI by a DFE; better results are obtained by incorporating the decision-feedback mechanism in the VA, that is using for each state a different feedback symbol sequence given by the survivor sequence.

The idea of cancelling the residual ISI based on the survivor sequence is a particular application of a general principle that is applied when the branch metric is affected by some uncertainty that can be eliminated or reduced by (possibly adaptive) estimation techniques of the type *data-aided*. Typical examples are the non-perfect knowledge of some channel characteristics, as the carrier phase, the timing phase, or the impulse response: all these cases can be solved in part by a general approach called *per survivor processing* (PSP) [19]. It represents an effective alternative to classical methods to estimate channel parameters, as the effects of error propagation are significantly reduced. Other interesting aspects characterizing this class of algorithms are the following.

[6] We note that if the points of the subsets of the partition $\Omega(1)$ are on a rectangular grid, as in the example of Figure 8.32, the value of the symbol a_k that minimizes the metric is determined through simple "quantization rules", without explicitly evaluating the branch metric for every symbol of the subset. Hence, for every state, only J_1 explicit computations of the branch metric are needed.

a. The estimator associated with the survivor sequence uses symbols that can be considered decisions with no delay and high reliability, making the PSP a suitable approach for channels that are fast time-varying.

b. *Blind* techniques, i.e. without knowledge of the training sequence, may be adopted as part of the PSP to estimate the various parameters.

A variant of the RSSE, belonging to the class of PSP algorithms, is the *decision feedback sequence estimator* (DFSE) [15] which considers as reduced state the vector $\mathbf{s}'_{k-1}$ formed simply by truncating the ML state vector at a length $N_1 \leq N$

$$\mathbf{s}'^{T}_{k-1} = [a_{k-1}, a_{k-2}, \ldots, a_{k-N_1}] \tag{8.349}$$

This is the same as considering $J_n = M$, for $n = 1, \ldots, N_1$, and $J_n = 1$, for $n = N_1 + 1, \ldots, N$.

We discuss the main points of this algorithm. We express the received sequence $\{z_k\}$, always under the hypothesis of a minimum-phase overall impulse response with $f_0 = 1$, as

$$z_k = a_k + \langle \mathbf{s}'_{k-1}, \mathbf{f}' \rangle + \langle \mathbf{s}''_{k-1}, \mathbf{f}'' \rangle + w_k \tag{8.350}$$

where $\mathbf{f}'$ and $\mathbf{f}''$, $\mathbf{s}'_{k-1}$, and $\mathbf{s}''_{k-1}$ are defined as follows:

$$\mathbf{f}^T = [\mathbf{f}'^T \mid \mathbf{f}''^T] = [f_1, \ldots, f_{N_1} \mid f_{N_1+1}, \ldots, f_N] \tag{8.351}$$

$$\mathbf{s}^T_{k-1} = [\mathbf{s}'^T_{k-1} \mid \mathbf{s}''^T_{k-1}] = [a_{k-1}, \ldots, a_{k-N_1} \mid a_{k-(N_1+1)}, \ldots, a_{k-N}] \tag{8.352}$$

The trellis diagram is built by assuming the reduced state $\mathbf{s}'_{k-1}$. The term $\langle \mathbf{s}''_{k-1}, \mathbf{f}'' \rangle$ represents the residual ISI that is estimated by considering as $\mathbf{s}''_{k-1}$ the symbols that are memorized in the survivor sequence associated with each state. We write

$$\hat{\mathbf{s}}''_{k-1} = \mathbf{s}''(\mathbf{s}'_{k-1}) \tag{8.353}$$

In fact, with respect to z_k, it is as if we have cancelled the term $\langle \mathbf{s}''_{k-1}, \mathbf{f}'' \rangle$ by a FB filter associated with each state $\mathbf{s}'_{k-1}$. With respect to the optimum path, the feedback sequence $\mathbf{s}''_{k-1}$ is expected to be very reliable.

The branch metric of the DFSE is computed as follows:

$$|z_k - a_k - \langle \mathbf{s}'_{k-1}, \mathbf{f}' \rangle - \langle \mathbf{s}''(\mathbf{s}'_{k-1}), \mathbf{f}'' \rangle|^2 \tag{8.354}$$

We note that the reduced state $\mathbf{s}'_{k-1}$ may be further reduced by adopting the RSSE technique (8.346).

The primary difference between an MLSD receiver and the DFSE is that in the trellis diagram used by the DFSE two paths may merge earlier, as it is sufficient that they share the

more recent N_1 symbols, rather than N as in MLSD. This increases the error probability; however, the performance of the DFSE is better than that achieved by a classical DFE.

8.14 Passband equalizers

For QAM signals, we analyze alternatives to the baseband equalizers considered at the beginning of this chapter. We refer to the QAM transmitter scheme of Figure 7.5 and we consider the model of Figure 8.34, where the transmission channel has impulse response g_{Ch} and introduces additive white noise $w(t)$. The received passband signal is given by

$$r(t) = Re\left[\sum_{k=-\infty}^{+\infty} a_k q_{Ch}(t - kT) e^{j(2\pi f_0 t + \varphi)}\right] + w(t) \tag{8.355}$$

where, from (7.42) or equivalently from Figure 7.11, q_{Ch} is a *baseband equivalent pulse* with frequency response

$$\mathcal{Q}_{Ch}(f) = \mathcal{H}_{Tx}(f)\,\tfrac{1}{2}\,\mathcal{G}_{Ch}^{(bb)}(f) = \mathcal{H}_{Tx}(f)\,\mathcal{G}_{Ch}(f + f_0)\,1(f + f_0) \tag{8.356}$$

In this model the phase φ in (8.355) implies that $\arg \mathcal{Q}_{Ch}(0) = 0$, hence it includes also the phase offset introduced by the channel frequency response at $f = f_0$, in addition to the carrier phase offset between the transmit and receive carriers. Because of this impairment, which as a first approximation is equivalent to a rotation of the symbol constellation, a suitable receiver structure needs to be developed [10].

As φ may be time-varying, it can be decomposed as the sum of three terms

$$\varphi(t) = \Delta\varphi + 2\pi\,\Delta f\,t + \psi(t) \tag{8.357}$$

where $\Delta\varphi$ is a fixed phase offset, Δf is a fixed frequency offset, and $\psi(t)$ is a random or quasi-periodic term (see the definition of phase noise in (4.271)).

For example, over telephone channels typically $|\psi(t)| \leq \pi/20$. Moreover, the highest frequency of the spectral components of $\psi(t)$ is usually lower than $0.1/T$: in other words, if $\psi(t)$ were a sinusoidal signal, it would have a period larger than $10T$.

Therefore $\varphi(t)$ may be regarded as a constant, or at least as *slowly time varying*, at least for a time interval equal to the duration of the overall system impulse response.

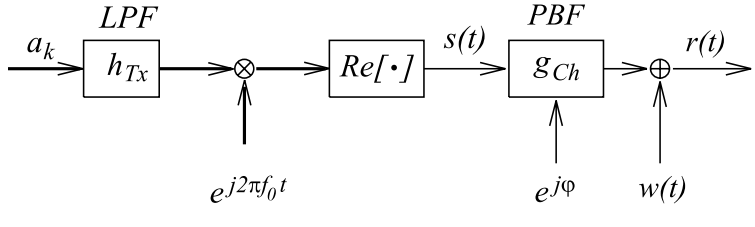


Figure 8.34. Passband modulation scheme with phase offset introduced by the channel.

8.14.1 Passband receiver structure

Filtering and equalization of the signal r are performed in the passband. First, a filter extracts the positive frequency components of r, acting also as a matched filter. It is a complex-valued passband filter, as illustrated in Figure 8.35.

In particular, from the theory of the optimum receiver we have

$$g_M^{(pb)}(t) = g_M(t)e^{j2\pi f_0 t} \tag{8.358}$$

where

$$g_M(t) = q_{Ch}^*(t_0 - t) \tag{8.359}$$

with q_{Ch} defined in (8.356).

Then, with reference to Figure 8.36, we have

$$g_{M,I}^{(pb)}(t) = Re\{g_M^{(pb)}(t)\} \quad \text{and} \quad g_{M,Q}^{(pb)}(t) = Im\{g_M^{(pb)}(t)\} \tag{8.360}$$

If g_M in (8.359) is real-valued, then $g_{M,I}^{(pb)}$ and $g_{M,Q}^{(pb)}$ are related by the Hilbert transform (1.163), that is

$$g_{M,Q}^{(pb)}(t) = \mathcal{H}^{(h)}[g_{M,I}^{(pb)}(t)] \tag{8.361}$$

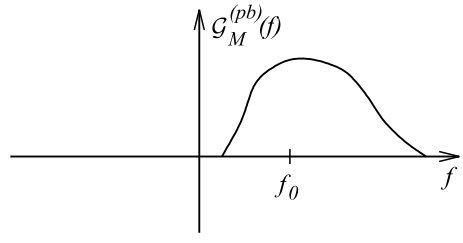


Figure 8.35. Frequency response of a passband matched filter.

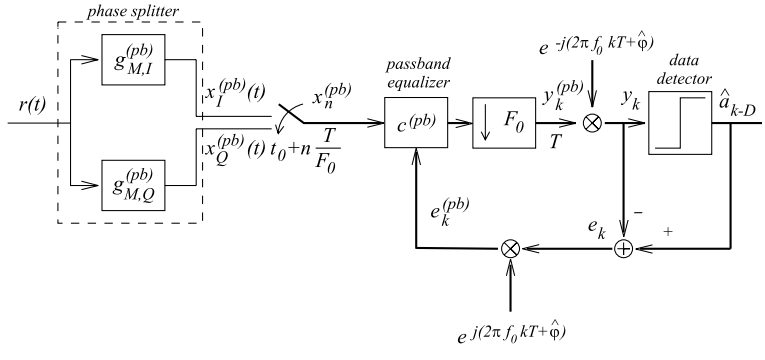


Figure 8.36. QAM passband receiver.

After the passband matched filter, the signal is oversampled with sampling period T/F_0; oversampling is suggested by the following two reasons:

1. If q_{Ch} is unknown and $g_M^{(pb)}$ is a simple passband filter, matched filtering is carried out by the filter $c^{(pb)}$, hence the need for oversampling.
2. If the timing phase t_0 is not accurate, it is convenient to use an FSE.

Let $q(t)$ be the overall baseband equivalent impulse response of the system at the sampler input,

$$q(t) = \mathcal{F}^{-1}[\mathcal{Q}(f)] \tag{8.362}$$

where

$$\mathcal{Q}(f) = \mathcal{Q}_{Ch}(f)\,\mathcal{G}_M(f) = \mathcal{H}_{Tx}(f)\,\mathcal{G}_{Ch}(f+f_0)\,\mathcal{G}_M(f)\,1(f+f_0) \tag{8.363}$$

The sampled passband signal is given by

$$\begin{aligned} x_n^{(pb)} &= x_I^{(pb)}\left(t_0 + n\frac{T}{F_0}\right) + jx_Q^{(pb)}\left(t_0 + n\frac{T}{F_0}\right) \\ &= \sum_{k=-\infty}^{+\infty} a_k h_{n-kF_0} e^{j\left(2\pi f_0 n\frac{T}{F_0}+\varphi\right)} + \tilde{w}_n \end{aligned} \tag{8.364}$$

where[7]

$$h_n = q\left(t_0 + n\frac{T}{F_0}\right) \tag{8.365}$$

In (8.364) $\tilde{w}_n$ denotes the noise component

$$\tilde{w}_n = w_R\left(t_0 + n\frac{T}{F_0}\right) \tag{8.366}$$

where $w_R(t) = w * g_M^{(pb)}(t)$.

For a passband equalizer with N coefficients, the output signal with sampling period equal to T is given by

$$y_k^{(pb)} = \sum_{i=0}^{N-1} c_i^{(pb)} x_{kF_0-i}^{(pb)} = \mathbf{x}_{kF_0}^{(pb)T}\mathbf{c}^{(pb)} \tag{8.367}$$

with the usual meaning of the two vectors $\mathbf{x}_n^{(pb)}$ and $\mathbf{c}^{(pb)}$.

Ideally it should result

$$y_k^{(pb)} = a_{k-D}e^{j(2\pi f_0 kT+\varphi)} \tag{8.368}$$

where the phase offset φ needs to be estimated.

[7] In (8.364) φ takes also into account the phase $2\pi f_0 t_0$.

In Figure 8.36 the signal $y_k^{(pb)}$ is shifted to baseband by multiplication with the function $e^{-j(2\pi f_0 kT+\hat{\varphi})}$, where $\hat{\varphi}$ is an estimate of φ. Then the data detector follows.

At this point some observations can be made: by demodulating the received signal, that is by multiplying it with the function $e^{-j2\pi f_0 t}$, before the equalizer or the receive filter we obtain a scheme equivalent to that of Figure 8.3, with a baseband equalizer. As we will see at the end of this section, the only advantage of a passband equalizer is that the computational complexity of the receiver is reduced; in any case, it is desirable to compensate for the presence of the phase offset, that is to multiply the received signal by $e^{-j\hat{\varphi}}$, as near as possible to the decision point, so that the delay in the loop for the update of the phase offset estimate is small.

Joint optimization of equalizer coefficients and carrier phase offset

To simplify the notation, the analysis is carried out for $F_0 = 1$.
As usual, for an error signal defined as

$$e_k = a_{k-D} - y_k \tag{8.369}$$

where y_k is given by (8.367), we desire to minimize the following cost function:

$$\begin{aligned} J &= E[|e_k|^2] \\ &= E[|a_{k-D} - \mathbf{x}_k^{(pb)T}\mathbf{c}^{(pb)}e^{-j(2\pi f_0 kT+\hat{\varphi})}|^2] \\ &= E[|a_{k-D}e^{j(2\pi f_0 kT+\hat{\varphi})} - \mathbf{x}_k^{(pb)T}\mathbf{c}^{(pb)}|^2] \end{aligned} \tag{8.370}$$

Equation (8.370) expresses the classical Wiener problem for a desired signal expressed as

$$a_{k-D}e^{j(2\pi f_0 kT+\hat{\varphi})} \tag{8.371}$$

and input $\mathbf{x}_k^{(pb)}$. Assuming φ is known, the Wiener–Hopf solution is given by

$$\mathbf{c}_{opt}^{(pb)} = \mathbf{R}^{-1}\mathbf{p} \tag{8.372}$$

where

$$\mathbf{R} = E[\mathbf{x}_k^{(pb)*}\mathbf{x}_k^{(pb)T}] \tag{8.373}$$

has elements for $\ell, m = 0, 1, \ldots, M-1$, given by

$$\begin{aligned} [\mathbf{R}]_{\ell,m} &= E[x_{k-m}^{(pb)}x_{k-\ell}^{(pb)*}] = \mathrm{r}_{x^{(pb)}}((\ell-m)T) \\ &= \sigma_a^2 \sum_{i=-\infty}^{+\infty} h_i h_{i-(\ell-m)}^* e^{j2\pi f_0(\ell-m)T} + \mathrm{r}_{w_R}((\ell-m)T) \end{aligned} \tag{8.374}$$

and

$$\mathbf{p} = E[a_{k-D}e^{j(2\pi f_0 kT+\hat{\varphi})}\mathbf{x}_k^{(pb)*}] \tag{8.375}$$

has elements for $\ell = 0, 1, \ldots, M-1$, given by:

$$[\mathbf{p}]_\ell = [\mathbf{p}']_\ell e^{-j(\varphi-\hat{\varphi})} \tag{8.376}$$

where

$$[\mathbf{p}']_\ell = \sigma_a^2 h^*_{D-\ell} e^{j2\pi f_0 \ell T} \tag{8.377}$$

From (8.376), the optimum solution (8.372) is expressed as:

$$\mathbf{c}^{(pb)}_{opt} = \mathbf{R}^{-1}\mathbf{p}' e^{-j(\varphi-\hat{\varphi})} \tag{8.378}$$

where $\mathbf{R}^{-1}$ and $\mathbf{p}'$ do not depend on the phase offset φ.

From (8.378) it can be verified that if φ is a constant the equalizer automatically compensates for the phase offset introduced by the channel, and the output signal $y_k^{(pb)}$ remains unchanged.

A difficulty appears if φ varies (slowly) in time and the equalizer attempts to track it. In fact, to avoid the output signal being affected by convergence errors, typically an equalizer in the steady state must not vary its coefficients by more than 1% within a symbol interval. Therefore another algorithm is needed to estimate φ.

Adaptive method

The adaptive LMS algorithm is used for an instantaneous squared error defined as

$$|e_k|^2 = |a_{k-D} - \mathbf{x}_k^{(pb)T}\mathbf{c}^{(pb)} e^{-j(2\pi f_0 kT+\hat{\varphi})}|^2 \tag{8.379}$$

The gradient of the function in (8.379) with respect to $\mathbf{c}^{(pb)}$ is equal to

$$\begin{aligned}\nabla_{\mathbf{c}^{(pb)}}|e_k|^2 &= -2e_k(\mathbf{x}_k^{(pb)} e^{-j(2\pi f_0 kT+\hat{\varphi}_k)})^* \\ &= -2e_k e^{j(2\pi f_0 kT+\hat{\varphi}_k)}\mathbf{x}_k^{(pb)*} \\ &= -2e_k^{(pb)}\mathbf{x}_k^{(pb)*}\end{aligned} \tag{8.380}$$

where

$$e_k^{(pb)} = e_k e^{j(2\pi f_0 kT+\hat{\varphi}_k)} \tag{8.381}$$

The law for coefficient adaptation is given by

$$\mathbf{c}_{k+1}^{(pb)} = \mathbf{c}_k^{(pb)} + \mu e_k^{(pb)}\mathbf{x}_k^{(pb)*} \tag{8.382}$$

We now compute the gradient with respect to $\hat{\varphi}_k$. Let

$$\theta = 2\pi f_0 kT + \hat{\varphi}_k \tag{8.383}$$

then

$$\begin{aligned} |e_k|^2 &= (a_{k-D} - y_k)(a_{k-D} - y_k)^* \\ &= (a_{k-D} - y_k^{(pb)} e^{-j\theta})(a_{k-D}^* - y_k^{(pb)*} e^{j\theta}) \end{aligned} \tag{8.384}$$

Therefore we obtain

$$\begin{aligned} \nabla_{\hat{\varphi}} |e_k|^2 = \frac{\partial}{\partial \theta} |e_k|^2 &= j y_k^{(pb)} e^{-j\theta} e_k^* - e_k j y_k^{(pb)*} e^{j\theta} \\ &= 2Im[e_k (y_k^{(pb)} e^{-j\theta})^*] \\ &= 2Im[e_k y_k^*] \end{aligned} \tag{8.385}$$

As $e_k = a_{k-D} - y_k$, (8.385) may be rewritten as

$$\nabla_{\hat{\varphi}} |e_k|^2 = 2Im[a_{k-D} y_k^*] \tag{8.386}$$

We note that $Im[a_{k-D} y_k^*]$ is related to the "sine" of the phase difference between a_{k-D} and y_k, therefore the algorithm has reached convergence only if the phase of y_k coincides (on average) with that of a_{k-D}.

The law for updating the phase offset estimate is given by

$$\hat{\varphi}_{k+1} = \hat{\varphi}_k - \mu_\varphi Im[e_k y_k^*] \tag{8.387}$$

or

$$\hat{\varphi}_{k+1} = \hat{\varphi}_k - \mu_\varphi Im[a_{k-D} y_k^*] \tag{8.388}$$

The adaptation gain is typically normalized as

$$\mu_\varphi = \frac{\tilde{\mu}_\varphi}{|a_{k-D}|\,|y_k|} \tag{8.389}$$

In general, μ_φ is chosen larger than μ, so that the variations of φ are tracked by the carrier phase offset estimator, and not by the equalizer.

In the ideal case, we have

$$y_k^{(pb)} = a_{k-D} e^{j(2\pi f_0 kT + \varphi)} \tag{8.390}$$

and

$$y_k = a_{k-D} e^{j(\varphi - \hat{\varphi}_k)} \tag{8.391}$$

therefore the adaptive algorithm becomes:

$$\begin{aligned} \hat{\varphi}_{k+1} &= \hat{\varphi}_k - \tilde{\mu}_\varphi Im[e^{-j(\varphi - \hat{\varphi}_k)}] \\ &= \hat{\varphi}_k + \tilde{\mu}_\varphi \sin(\varphi - \hat{\varphi}_k) \end{aligned} \tag{8.392}$$

which is the equation of a first-order *phase-locked-loop* (PLL) (see Section 14.7). For a constant phase offset, at convergence we get $\hat{\varphi} = \varphi$.

8.14.2 Efficient implementations of voiceband modems

As an example, we consider the previous QAM schemes in the case of transmission over a *telephone channel*, with passband in the range 300–3400 Hz, for which a carrier f_0 in the range 1700–1800 Hz and a symbol rate $1/T$ of 2400 Baud are adopted. We note that for a transmit pulse h_{Tx} of the square root raised cosine type with roll-off $\rho = 0.125$ the transmission band goes from $f_0 - 1350$ Hz to $f_0 + 1350$ Hz.

Using the configuration of Figure 8.4 with a sampling rate of 9600 Hz, equivalent to a sampling period $T_c = T/4$, the complex-valued scheme of a passband receiver, employing an FSE with sampling period equal to $T/2$ and a discrete-time phase-splitter, is illustrated in Figure 8.37. We note that, after the phase splitter filter, the frequency support of the received signal is halved, therefore we can sample with sampling period $T/2$.

The *loop filter* updates the value of $\hat{\varphi}_k$ according to (8.388). The *voltage controlled oscillator* (VCO) determines $\theta_k = 2\pi f_0 kT + \hat{\varphi}_k$, and computes the complex exponential $e^{j\theta_k}$. The algorithm for coefficient updating is given by (8.382).

Let us consider some simplifications in the structure of Figure 8.37. From (8.360) let $g_{M,I}^{(pb)}$ and $g_{M,Q}^{(pb)}$, be, respectively, the in-phase and quadrature components of $g_M^{(pb)}$.

Let $c_I^{(pb)}$ and $c_Q^{(pb)}$ be, respectively, the in-phase and quadrature components of the filter $c^{(pb)}$. To determine $y_k^{(pb)}$, from the structure of Figure 8.37 we obtain the structure of Figure 8.38, where real-valued signals are considered. Overall we have six discrete-time convolutions. Combining the phase splitter and the equalizer of Figure 8.38, a single filter $c^{(pb)}$ is used that operates with sampling period of the input signal equal to $T/4$, and sampling period of the output signal equal to T, as depicted in Figure 8.39.

The LMS algorithm for coefficient adaptation is given by

$$\mathbf{c}_{k+1}^{(pb)} = \mathbf{c}_k^{(pb)} + \mu e_k^{(pb)} \mathbf{x}_{4k} \tag{8.393}$$

where $c^{(pb)}$ is a complex-valued filter, and the passband input signal x is real-valued. The overall complexity of filtering and adaptation is lower as compared to the previous realization, although convergence is slower. A receiver structure with a baseband adaptive filter is depicted in Figure 8.40.

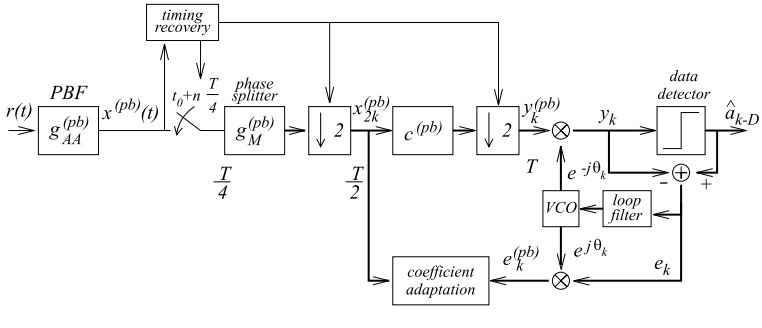


Figure 8.37. QAM passband receiver for transmission over telephone channels.

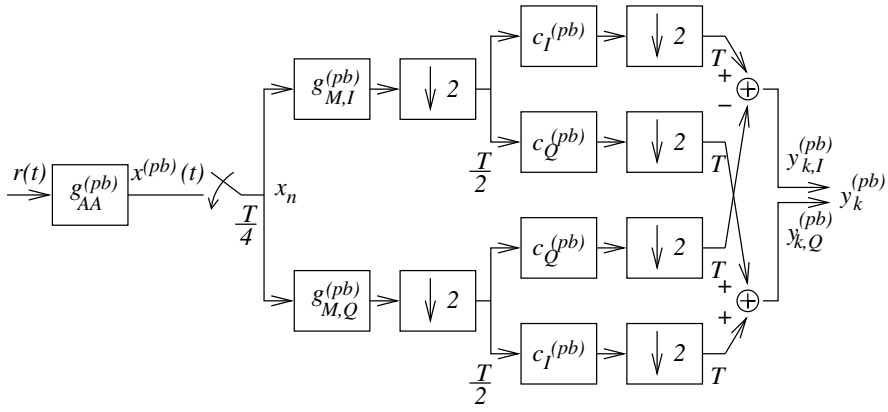


Figure 8.38. Implementation of the scheme of Figure 8.37 using real-valued signals and filters.

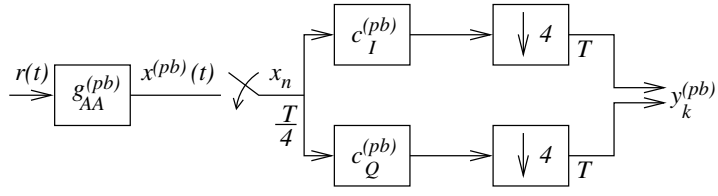


Figure 8.39. Efficient implementation of the receiver of Figure 8.37 combining phase splitter and equalizer.

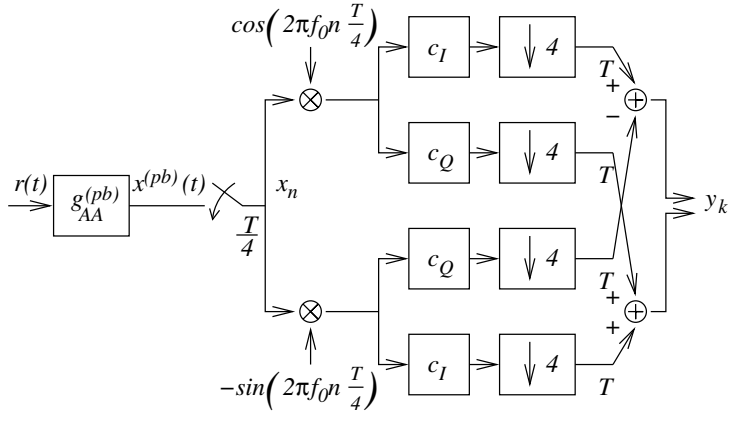


Figure 8.40. Efficient implementation of the QAM receiver using a baseband filter.

8.15 LE for voiceband modems

A fast method, in the sense that it requires a relatively short training interval, to determine the coefficients of a LE in the scheme of Figure 8.36 consists of using a periodic PN training sequence of period LT, equal to the time span of the equalizer filter delay line. In this case we speak of *cyclic equalization*, because the received signal, even though it is distorted, in the absence of noise is periodic of period LT [20].

We consider equalizers with N coefficients and sampling period of the input signal equal to T/F_0, with $F_0 > 1$, because of the robustness they offer with respect to the choice of the timing phase. As it must be $N(T/F_0) = LT$, we have

$$N = F_0 L \tag{8.394}$$

With reference to Figure 8.36 and (8.364), we consider the case in which after the phase splitter, the signal $x_n^{(pb)}$ is shifted to baseband using a nominal carrier frequency f_0. From (8.364) and (8.357), we define

$$\begin{aligned} \bar{x}_n &= x_n^{(pb)} e^{-j2\pi f_0 n \frac{T}{F_0}} \\ &= \sum_{\ell=-\infty}^{+\infty} a_\ell h_{n-\ell F_0} e^{j\left(2\pi \Delta f n \frac{T}{F_0} + \Delta\varphi\right)} + \bar{w}_n \end{aligned} \tag{8.395}$$

where Δf and $\Delta\varphi$ represent, respectively, the carrier frequency offset and phase offset, both unknown quantities. In (8.395), the noise $\bar{w}_n$ is given by

$$\bar{w}_n = \tilde{w}_n e^{-j2\pi f_0 n \frac{T}{F_0}} \tag{8.396}$$

where $\tilde{w}_n$ is defined in (8.366).

The scheme of Figure 8.36 is redrawn as in Figure 8.41.

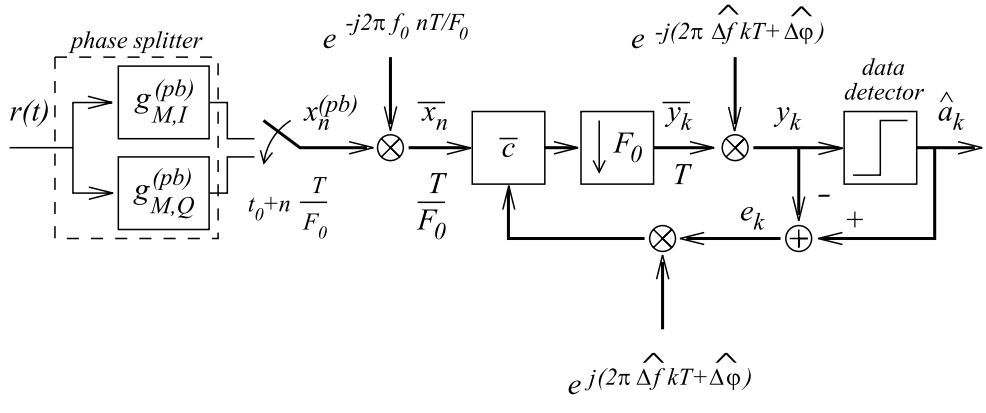


Figure 8.41. Receiver with cyclic equalizer for voiceband modems.

From (8.395), defining the *training signal*

$$v_n = \sum_{\ell=-\infty}^{+\infty} a_\ell h_{n-\ell F_0} \tag{8.397}$$

at the instant the equalizer outputs the sample $\bar{y}_k$, the samples stored in the delay line of the filter $\bar{c}$ are given by

$$\bar{x}_{kF_0-i} = v_{kF_0-i} e^{j\left(2\pi \Delta f (kF_0-i)\frac{T}{F_0}+\Delta\varphi\right)} + \bar{w}_{kF_0-i} \qquad i = 0, 1, \ldots, N-1 \tag{8.398}$$

Choosing as training sequence $\{a_\ell\}$ the repetition of a PN sequence with period L, the sequence $\{v_n\}$ is periodic with period $N = F_0 L$; in particular,

$$v_{kF_0-N-i} = v_{kF_0-i} \tag{8.399}$$

Consequently, from (8.395), neglecting the noise $\bar{w}_n$, the sample that leaves the equalizer delay line, $\bar{x}_{kF_0-N}$, differs from the sample that enters, $\bar{x}_{kF_0}$, by a phase rotation equal to $2\pi \Delta f N(T/F_0) = 2\pi \Delta f LT$. On the other hand, noise samples that are separate by an interval equal to LT are uncorrelated.

We consider now two problems: detection of the beginning of the transmission and efficient computation of the equalizer coefficients.

Detection of the training sequence

The presence of the cyclic training signal can be detected by considering the following metric:

$$\mathsf{M} = \frac{1}{\hat{\mathrm{M}}_{\bar{x}}} \frac{1}{N_w} \sum_{n=0}^{N_w-1} \left| \bar{x}_{kF_0-nF_0} - \bar{x}_{kF_0-N-nF_0} e^{j2\pi \widehat{\Delta f} LT} \right|^2 \tag{8.400}$$

where N_w is the number of observations utilized and $\hat{\mathrm{M}}_{\bar{x}}$ is an estimate of the statistical power of $\bar{x}_n$.

We underline that in (8.400), as Δf is unknown, we use an estimate of the value $\widehat{\Delta f}$, which can be derived by minimizing M with respect to $\widehat{\Delta f}$. This choice yields

$$\widehat{\Delta f}_{opt} = \frac{1}{2\pi LT} \arg\left(\sum_{n=0}^{N_w-1} \bar{x}_{kF_0-nF_0} \bar{x}^*_{kF_0-N-nF_0} \right) \tag{8.401}$$

Because $-\pi < \arg < \pi$, $|\widehat{\Delta f}_{opt}|$ is smaller than $\frac{1}{2LT}$; this fact sets a limit on Δf. For example, for a filter with $L = 32$ and $1/T = 2400$ Baud it must be $|\Delta f| < 37.5$ Hz.

For $\widehat{\Delta f}_{opt}$ given by (8.400), the value of M is computed. If it falls below a certain threshold then we detect the presence of a full period of the training signal in the equalizer filter delay line, and we trigger the procedure for computing the equalizer coefficients.

A statistical analysis leads to the following relations [21]

$$E\,[\,\mathsf{M}\mid \text{training signal present}\,] = \frac{2\sigma_{\bar{w}}^2}{\mathtt{M}_{\bar{x}}}\left(1-\frac{1}{2N_w}\right) \tag{8.402}$$

and

$$E\,[\,\mathsf{M}\mid \text{training signal absent}\,] = \frac{2\sigma_{\bar{w}}^2}{\mathtt{M}_{\bar{x}}}\left(1-\sqrt{\frac{2}{\pi N_w}}\right) \tag{8.403}$$

Considering that in the presence of the training signal we have $\mathtt{M}_{\bar{x}} \gg \sigma_{\bar{w}}^2$, whereas in the absence of the training signal $\mathtt{M}_{\bar{x}} \simeq \sigma_{\bar{w}}^2$, the metric in (8.402) is much lower than that in (8.403). In [21] it is observed that a value of $N_w = 8$ is adequate to reach a false alarm probability, i.e. the probability of detecting the training sequence in the presence of noise only, of 10^{-4} and a probability of missed detection, i.e. the probability of not detecting the training sequence, of 10^{-6}, even in the presence of severe distortion introduced by the telephone channel.

Computations of the coefficients of a cyclic equalizer

As the training sequence $\{a_k\}$ has a period LT, the training signal $\{v_n\}$ has spectral lines spaced of $1/(LT)$ Hz.

Ideally, in the absence of noise, the equalizer compensates exactly for the channel distortion at L frequencies if the N coefficients $\{c_i\}$ are such that

$$\sum_{i=0}^{N-1} c_i\, v_{kF_0-i} = a_{k_{mod\,L}} \qquad \forall k \tag{8.404}$$

An estimate $\hat{v}_n$ of the sample v_n is obtained by first compensating for the frequency offset in $\bar{x}_n$, and successively averaging the received signal over N_v periods

$$\begin{aligned}\hat{v}_{kF_0-i} &= \frac{1}{N_v}\sum_{m=0}^{N_v-1} \bar{x}_{kF_0-i-mN}\, e^{-j2\pi\widehat{\Delta f}_{opt}(kF_0-i-mN)\frac{T}{F_0}} \\ &= e^{-j2\pi\widehat{\Delta f}_{opt}\left(k-\frac{i}{F_0}\right)T}\,\frac{1}{N_v}\sum_{m=0}^{N_v-1} \bar{x}_{kF_0-i-mN}\, e^{j2\pi\widehat{\Delta f}_{opt}mLT}\end{aligned} \tag{8.405}$$

In a practical implementation it is convenient to use an equalizer with coefficients given by

$$\bar{c}_i = c_i e^{j2\pi\widehat{\Delta f}_{opt} i\frac{T}{F_0}} \qquad i = 0, 1, \ldots, N-1 \tag{8.406}$$

Substituting $\hat{v}$ for v and interpreting (8.404) as a circular convolution, we have

$$\sum_{i=0}^{N-1} c_i\, \hat{v}_{kF_0-i} = a_k \qquad k = 0, 1, \ldots, L-1 \tag{8.407}$$

We now take the DFT of both members in the previous equation. Let

$$\mathcal{A}_p = \sum_{k=0}^{L-1} a_k e^{-j2\pi \frac{kp}{L}} \qquad p = 0, 1, \ldots, L-1 \tag{8.408}$$

$$\mathcal{C}_q = \sum_{i=0}^{N-1} c_i e^{-j2\pi \frac{iq}{N}} \quad q = 0, 1, \ldots, N-1 \tag{8.409}$$

$$\hat{\mathcal{V}}_q = \sum_{i=0}^{N-1} \hat{v}_i e^{-j2\pi \frac{iq}{N}} \quad q = 0, 1, \ldots, N-1 \tag{8.410}$$

then, for $N = F_0 L$, (8.407) becomes

$$\mathcal{A}_p = \frac{1}{F_0} \sum_{r=0}^{F_0-1} \mathcal{C}_{p+rL} \hat{\mathcal{V}}_{p+rL} \quad p = 0, 1, \ldots, L-1 \tag{8.411}$$

The above relation represents a system of L linear equations and $N = F_0 L$ unknowns $\{\mathcal{C}_q\}$. Among the infinite number of solutions we choose the one that minimizes the filter energy

$$\sum_{i=0}^{N-1} |c_i|^2 = \frac{1}{N} \sum_{q=0}^{N-1} |\mathcal{C}_q|^2 \tag{8.412}$$

For the minimization of (8.412), with the constraint (8.411), the method of the Lagrange multipliers may be used. We consider here the particular case $F_0 = 2$. For the general case we refer to [21].

For $F_0 = 2$, we have $N = 2L$, and the solution is given by

$$\mathcal{C}_q = \frac{2\mathcal{A}_{q_{mod\,L}}\, \hat{\mathcal{V}}_q^*}{|\hat{\mathcal{V}}_{q_{mod\,L}}|^2 + |\hat{\mathcal{V}}_{(L+q)_{mod\,L}}|^2} \qquad q = 0, 1, \ldots, N-1 \tag{8.413}$$

The filter coefficients $\{c_i\}$ are obtained by inverse DFT:

$$c_i = \frac{1}{N} \sum_{q=0}^{N-1} \mathcal{C}_q e^{j2\pi \frac{iq}{N}} \qquad i = 0, 1, \ldots, N-1 \tag{8.414}$$

Summarizing, the algorithm for the computation of the coefficients of an FSE with sampling period of the input signal equal to $T/2$ includes the following steps:

1. record N_v sequences of $2L$ samples from the equalizer delay line;
2. remove the phase rotation as indicated by (8.405);
3. perform a $2L$-point DFT to obtain $\{\mathcal{V}_q\}$ ($\{\mathcal{A}_p\}$ may be pre-computed);
4. compute $\{\mathcal{C}_q\}$ as indicated by (8.413);
5. perform a $2L$-point inverse DFT to obtain $\{c_i\}$;
6. adjust the coefficients as indicated by (8.406).

Transition from training to data mode

To equalize an aperiodic signal, it is necessary to select the equalizer coefficients so that they implement a linear convolution rather than a circular convolution: this is obtained by positioning the coefficients to better reflect the channel delay characteristics. In practice, the coefficients $\{\bar{c}_i\}$ are cyclically shifted of S positions so that the coefficients with larger values are positioned near the center of the filter. Note that this introduces a delay of S samples in the timing phase of the downsampler at the equalizer output.

When the computation of the coefficients is completed, the equalizer yields output samples $\{\bar{y}_k\}$ every T seconds. To compensate the frequency and phase offsets, the samples are then multiplied by the term $e^{-j\theta_k}$, where

$$\theta_k = 2\pi(\widehat{\Delta f})_k kT + (\widehat{\Delta\varphi})_k \tag{8.415}$$

Initially, the phase synchronization is obtained by setting

$$(\Delta\varphi)_0 = S2\pi \frac{T}{F_0} \widehat{\Delta f}_{opt} \tag{8.416}$$

to compensate for the phase introduced by the shift of the equalizer coefficients.

After transition to data mode, the various parameters are updated by adaptive algorithms, similar to those derived in Section 8.14.1. In particular, to update the filter coefficients, from (8.382), we have

$$\bar{\mathbf{c}}_{k+1} = (1 - \mu\alpha)\, \bar{\mathbf{c}}_k + \mu \bar{e}_k\, \bar{\mathbf{x}}_k^* \tag{8.417}$$

where $0 < \alpha < 1$ is a parameter that allows the LMS algorithm to track small variations of the channel characteristics (see (3.125)). To update the phase estimate, from (8.387) we have

$$\theta_{k+1} = \theta_k - \mu_\varphi Im[e_k y_k^*] + 2\pi T \widehat{\Delta f} \tag{8.418}$$

where $\widehat{\Delta f}$, initialized with $\widehat{\Delta f}_{opt}$ given by (8.401), is updated by a second order PLL (see Section 14.7).

Example of application: a simple modem

A 2400 Baud modem implemented by a *digital signal processor* (DSP) is described in [22]: this modem uses transmit and receive filters with a time span of the impulse response equal to $6T$, an FSE with $N = 64$ coefficients, and sampling period of the input signal equal to $T/2$. The equalizer design is performed in the frequency domain by the procedure discussed in the previous section. Because of its good spectral characteristics, a CAZAC sequence of length $L = 32$ symbols is used, and the number of observations to obtain the estimate in (8.401) is equal to $N_w = 8$. The minimum start-up time is equal to $(L + N_w)T = 16.6$ ms; in any case, considering also the various transients, it is typically of the order of 20 ms.

8.16 LE and DFE in the frequency domain with data frames using cyclic prefix

We consider now an equalization method, which is attractive for the low complexity that it requires for determining the filter coefficients and for performing the filtering operations.

From the model of Figure 8.7, we obtain the discrete-time model of Figure 8.42, where the sampling period of the equalizer input signal is equal to $T/2$, and the polyphase representations of the sequences x, h, and c are employed; in particular we have the following relations.

- Overall impulse response at the receive filter output, with sampling period equal to $T/2$,

$$\{h_i\} \qquad i = 0, 1, \ldots, 2N_h - 1 \tag{8.419}$$

- Polyphase components of the overall impulse response,

$$h_m^{(\ell)} = h_{2m+\ell} \qquad m = 0, 1, \ldots, N_h - 1 \qquad \ell = 0, 1 \tag{8.420}$$

- Impulse response of the equalizer with sampling period of the input signal equal to $T/2$,

$$\{c_i\} \qquad i = 0, 1, \ldots, N - 1 \qquad N \text{ even} \tag{8.421}$$

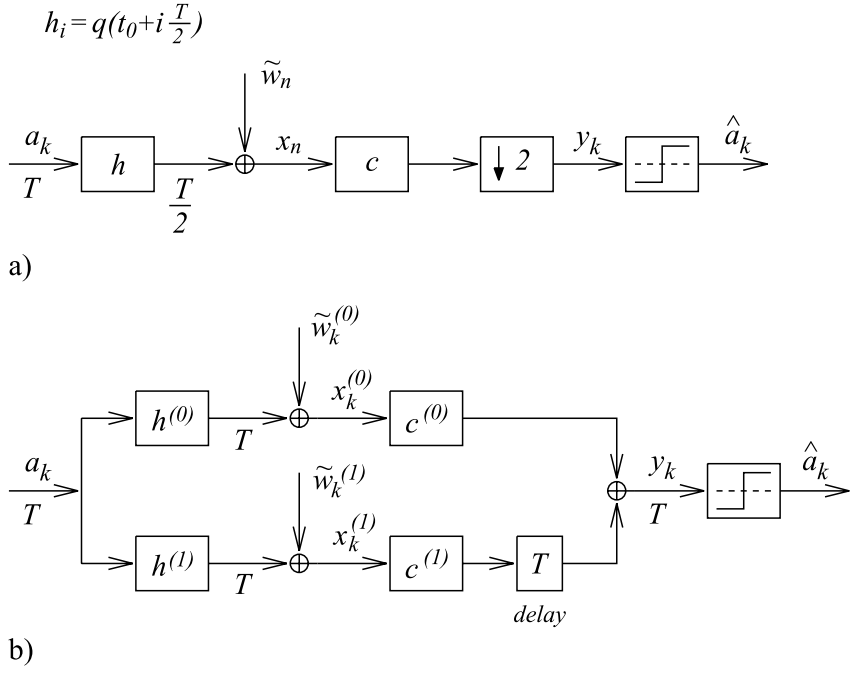


Figure 8.42. (a) Discrete-time model; (b) overall model employing the polyphase representation.

- Polyphase components of the equalizer impulse response

$$c_m^{(\ell)} = c_{2m+\ell} \qquad m = 0, 1, \ldots, \frac{N}{2} - 1 \qquad \ell = 0, 1 \tag{8.422}$$

- Sequence of samples $\{x_n\}$ at the receive filter output with polyphase components $\{x_k^{(0)} = x_{2k}\}$ and $\{x_k^{(1)} = x_{2k+1}\}$.
- White noise sequence $\{\tilde{w}_n\}$ with PSD N_0, and polyphase components $\{\tilde{w}_k^{(0)}\}$ and $\{\tilde{w}_k^{(1)}\}$.

Let $\{a_k\}$, $k = 0, 1, \ldots, \mathcal{M} - 1$, be the transmitted sequence of symbols. To simplify the implementation of the equalizer, we consider the transmission of an extended sequence of symbols $\{a_k^{(px)}\}$, obtained by partially repeating $\{a_k\}$ [23]

$$a_k^{(px)} = \begin{cases} a_k & k = 0, 1, \ldots, \mathcal{M} - 1 \\ a_{\mathcal{M}+k} & k = -1, \ldots, -N_{px} \end{cases} \tag{8.423}$$

In (8.423) N_{px} is the length of the *cyclic prefix*, which is related to N_h, length of the channel impulse response in number of symbol periods T, so that it results

$$N_{px} \geq N_h - 1 \tag{8.424}$$

We note that (8.423) assumes a transmission data frame such that, between blocks of data, there is a guard period within which the data are partially repeated. For a given bandwidth of the transmission channel, or rather for a given symbol rate, the system introduces an overhead of N_{px} symbols every $\mathcal{M}$ information symbols.

From the received sequences $\{x_k^{(\ell)}\}$, for $k = -N_{px}, \ldots, -1, 0, 1, \ldots, \mathcal{M} - 1$, $\ell = 0, 1$, we omit the first N_{px} samples, and we consider the sequence

$$z_k^{(\ell)} = \begin{cases} x_k^{(\ell)} & k = 0, 1, \ldots, \mathcal{M} - 1 \\ 0 & \text{elsewhere} \end{cases} \tag{8.425}$$

We also introduce the following vectors with $\mathcal{M}$ components and the corresponding DFTs:

$$\mathbf{a} = [a_0, \ldots, a_{\mathcal{M}-1}]^T \tag{8.426}$$

$$\boldsymbol{\mathcal{A}} = [\mathcal{A}_0, \ldots, \mathcal{A}_{\mathcal{M}-1}]^T = \text{DFT}[\mathbf{a}] \tag{8.427}$$

For $\ell = 0, 1$,

$$\mathbf{h}^{(\ell)} = [h_0^{(\ell)}, \ldots, h_{N_h-1}^{(\ell)}, 0, \ldots, 0]^T \tag{8.428}$$

$$\boldsymbol{\mathcal{H}}^{(\ell)} = [\mathcal{H}_0^{(\ell)}, \ldots, \mathcal{H}_{\mathcal{M}-1}^{(\ell)}]^T = \text{DFT}[\mathbf{h}^{(\ell)}] \tag{8.429}$$

$$\mathbf{c}^{(\ell)} = [c_0^{(\ell)}, \ldots, c_{(N/2)-1}^{(\ell)}, 0, \ldots, 0]^T \tag{8.430}$$

$$\boldsymbol{\mathcal{C}}^{(\ell)} = \text{DFT}[\mathbf{c}^{(\ell)}] \tag{8.431}$$

$$\mathbf{z}^{(\ell)} = [z_0^{(\ell)}, \ldots, z_{\mathcal{M}-1}^{(\ell)}]^T \tag{8.432}$$

$$\boldsymbol{\mathcal{Z}}^{(\ell)} = \text{DFT}[\mathbf{z}^{(\ell)}] \tag{8.433}$$

$$\mathbf{y} = [y_0, \ldots, y_{\mathcal{M}-1}]^T \tag{8.434}$$

$$\mathcal{Y} = \mathrm{DFT}[\mathbf{y}] \tag{8.435}$$

It is easy to verify that if (8.424) holds, the same conditions as in (1.116) are verified, therefore we have the relation

$$\mathcal{Z}_m^{(\ell)} = \mathcal{A}_m \mathcal{H}_m^{(\ell)} \qquad m = 0, 1, \ldots, \mathcal{M}-1 \tag{8.436}$$

Moreover, as seen in Figure 8.42b, it results

$$\mathcal{Y}_m = \mathcal{Z}_m^{(0)} \mathcal{C}_m^{(0)} + \mathcal{Z}_m^{(1)} \bar{\mathcal{C}}_m^{(1)} \qquad m = 0, 1, \ldots, \mathcal{M}-1 \tag{8.437}$$

with

$$\bar{\mathcal{C}}_m^{(1)} = \mathcal{C}_m^{(1)}\, e^{-j2\pi \frac{m}{\mathcal{M}}} \tag{8.438}$$

Finally, we have

$$\mathbf{y} = \mathrm{IDFT}[\mathcal{Y}] \tag{8.439}$$

The receiver structure with a linear equalizer in the frequency domain is illustrated in Figure 8.43; the convolution between x and c is substituted by three $\mathcal{M}$-point DFTs.

The attractiveness of this structure resides in the simplicity of the determination of the equalizer coefficients to be used in (8.437). A first method, described in Section 8.15, consists of adopting a training sequence of suitable length. Then from the DFTs of the various signals we determine the DFT of the sequence c (see (8.413)). As an alternative, proposed in [24], we describe the MSE method (8.12) that for $f = q\frac{1}{2\mathcal{M}T/2}$, $q = 0, 1, \ldots, 2\mathcal{M}-1$, yields the DFT of the optimum FSE with sampling period of the input signal equal to $T/2$. In this case, we assume as known the impulse response $\{h_i\}$ and also its $2\mathcal{M}$-point DFT,

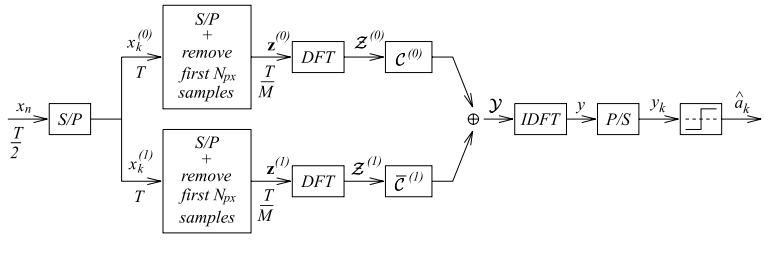


Figure 8.43. Structure of a linear equalizer in the frequency domain with data frames using cyclic prefix.

that we will indicate as $\{\mathcal{H}_q\}$, $q = 0, 1, \ldots, 2\mathcal{M} - 1$. The DFT of the optimum equalizer is given by

$$\mathcal{C}_q = \mathcal{G}_{Rc}(f)|_{f=q\frac{1}{2\mathcal{M}T/2}} = \frac{\sigma_a^2 \mathcal{H}_q^*}{N_0 + \sigma_a^2 \frac{1}{2}[|\mathcal{H}_q|^2 + |\mathcal{H}_{q+\mathcal{M}}|^2]} \qquad q = 0, 1, \ldots, 2\mathcal{M} - 1 \tag{8.440}$$

Recalling the properties of the DFT and (8.422) we have, for $m = 0, 1, \ldots, \mathcal{M} - 1$,

$$\mathcal{C}_m^{(0)} = \tfrac{1}{2}(\mathcal{C}_m + \mathcal{C}_{m+\mathcal{M}}) \tag{8.441}$$

and

$$\mathcal{C}_m^{(1)} = \tfrac{1}{2}\left(e^{j2\pi\frac{m}{2\mathcal{M}}}\mathcal{C}_m + e^{j\frac{2\pi}{2\mathcal{M}}(m+\mathcal{M})}\mathcal{C}_{m+\mathcal{M}}\right) \tag{8.442}$$

$$= \tfrac{1}{2}e^{j\pi\frac{m}{\mathcal{M}}}(\mathcal{C}_m - \mathcal{C}_{m+\mathcal{M}}) \tag{8.443}$$

or, from (8.438),

$$\bar{\mathcal{C}}_m^{(1)} = \frac{e^{-j\pi\frac{m}{\mathcal{M}}}}{2}(\mathcal{C}_m - \mathcal{C}_{m+\mathcal{M}}) \tag{8.444}$$

We note that, exploiting the data frame structure with cyclic prefix, the implementation of Figure 8.43 uses DFTs with a number of samples lower than that required by the general frequency domain method illustrated in Figure 3.22.

A frequency domain DFE, which utilizes a known data sequence as guard interval, rather than a prefix, has been proposed in [25]. In general, its performance is much better than that of the above LE configuration. In [25] design methods with a reduced complexity are also proposed for the direct design of the FF filter in the frequency domain.

8.17 Numerical results obtained by simulations

Using Monte Carlo simulations (see Appendix 7.E) we give a comparison, in terms of P_{bit} as a function of the signal-to-noise Γ, of the various equalization and data detection methods described in the previous sections. We refer to the system model of Figure 8.5 with an overall impulse response $\{h_n\}$ having five coefficients, as given in Table 1.4 on page 26, and AWGN noise $\tilde{w}$. Recalling the definition of the signal-to-noise $\Gamma = \sigma_a^2 \mathrm{r}_h(0)/\sigma_{\tilde{w}}^2$, we examine four cases.

QPSK transmission over a minimum phase channel

We examine the following receivers, where the delay D is optimized by applying the MSE criterion.

We anticipate that the ZF equalizer is designed by the DFT method (see Section 8.7). We also introduce the abbreviation DFE-VA to indicate the method consisting of the FF filter of a DFE, followed by MLSD implemented by the VA, with M^{M_2} states determined by the impulse response $\eta_n = \psi_{n+D}$ for $n = 0, \ldots, M_2$: this technique is commonly used to shorten the overall channel impulse response and thus simplify the VA (see case (2a) on page 679).

1. ZF with $N = 7$ and $D = 0$;
2. LE with $N = 7$ and $D = 0$;
3. DFE with $M_1 = 7$, $M_2 = 4$, and $D = 6$;
4. VA with $4^4 = 256$ states and path memory depth equal to 15, i.e. approximately three times the length of $\{h_n\}$;
5. DFSE with $J_1 = J_2 = 4$, $J_3 = J_4 = 1$ (16 states);
6. DFE-VA with $M_1 = 7$, $M_2 = 2$, and $D = 0$ (VA with $4^{M_2} = 16$ states).

The performance of the various receivers is illustrated in Figure 8.44; for comparison, the performance achieved by transmission over an ideal AWGN channel is also given.

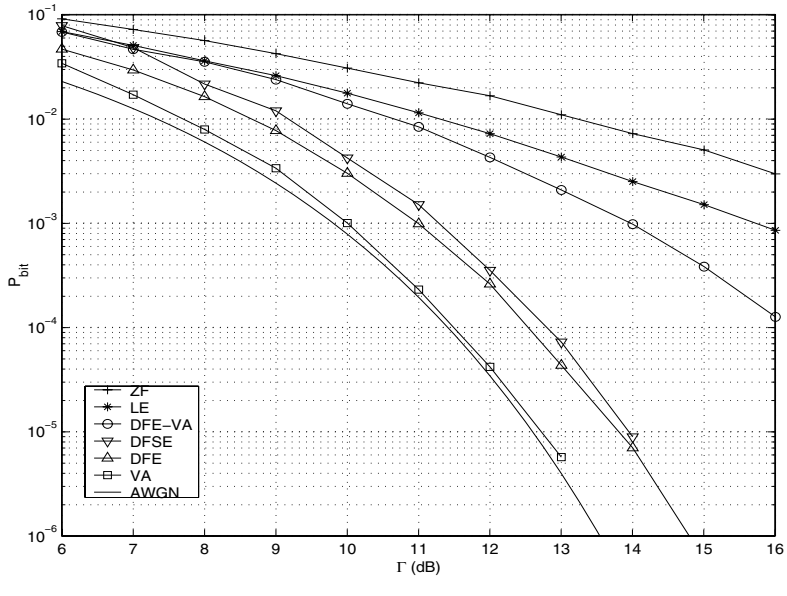


Figure 8.44. Bit error probability, P_{bit}, as a function of Γ for QPSK transmission over a minimum phase channel, using various equalization and data detection methods.

QPSK transmission over a non-minimum phase channel

We examine the following receivers.

1. ZF with $N = 7$ and $D = 4$;
2. LE with $N = 7$ and $D = 4$;
3. DFE with $M_1 = 7$, $M_2 = 4$, and $D = 6$;
4. VA with $4^4 = 256$ states and path memory depth equal to 15;
5. DFSE with $J_1 = J_2 = 4$, $J_3 = J_4 = 1$ (16 states);
6. DFE-VA with $M_1 = 7$, $M_2 = 2$, and $D = 4$ (VA with 16 states).

The performance of the various receivers is illustrated in Figure 8.45.

Comparing the error probability curves shown in Figure 8.44 and 8.45, we note that the performance of the VA is better than that of the other methods; moreover, performance of the VA is almost independent of the phase of the overall channel impulse response; however, if the channel is minimum phase even a simple DFE or DFSE can give performance close to the optimum. We also note that in these simulations, the DFE-VA gives poor performance because the value of M_2 is too small, hence the DFE is unable to equalize the channel.

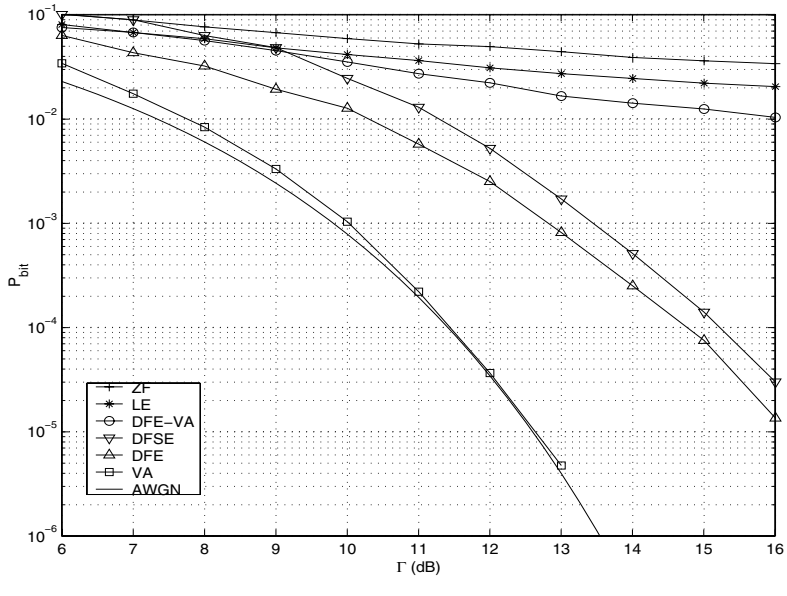


Figure 8.45. Bit error probability, P_{bit}, as a function of Γ for QPSK transmission over a non-minimum phase channel, using various equalization and data detection methods.

8-PSK transmission over a minimum phase channel

We examine the following receivers.

1. DFE with $M_1 = 6$, $M_2 = 4$, and $D = 0$;
2. DFSE with $J_1 = J_2 = 8$, $J_3 = J_4 = 1$ (64 states);
3. DFE-VA with $M_1 = 7$, $M_2 = 2$, and $D = 0$ (VA with $8^{M_2} = 64$ states);
4. RSSE with $J_1 = 8$, $J_2 = 4$, $J_3 = 2$, $J_4 = 1$ (64 states).

The performance of the various receivers is illustrated in Figure 8.46.

8-PSK transmission over a non-minimum phase channel

We examine the following receivers.

1. DFSE with $J_1 = J_2 = 8$, $J_3 = J_4 = 1$ (64 states);
2. DFE with $M_1 = 12$, $M_2 = 4$, and $D = 11$;
3. RSSE with $J_1 = 8$, $J_2 = 4$, $J_3 = 2$, $J_4 = 1$ (64 states).

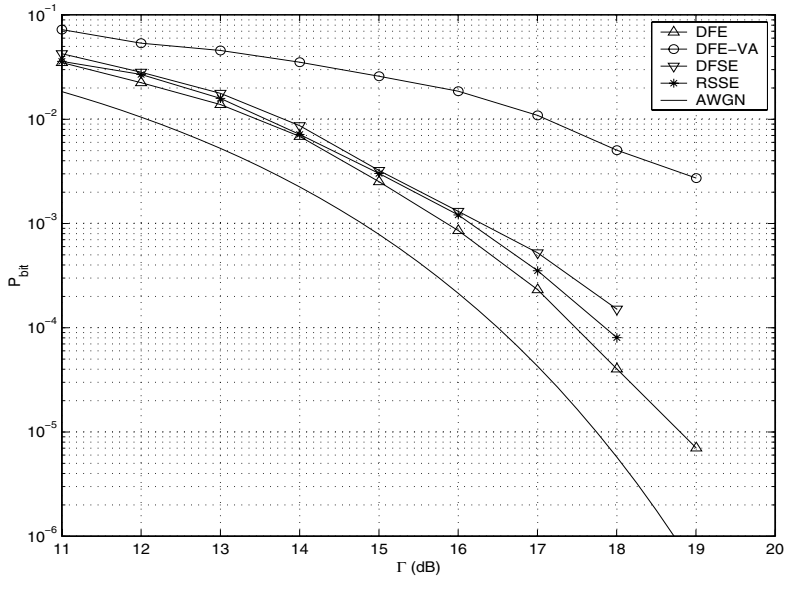


Figure 8.46. Bit probability error, P_{bit}, as a function of Γ for 8-PSK transmission over a minimum phase channel, using various equalization and data detection methods.

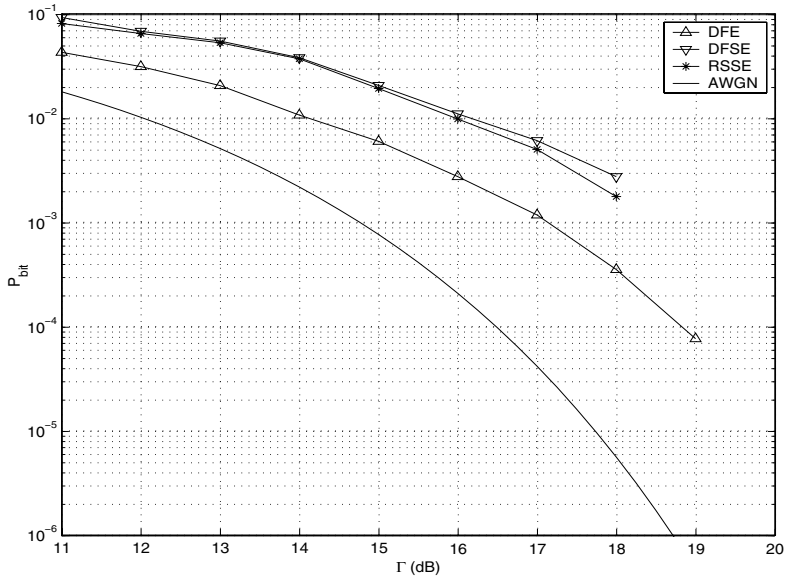


Figure 8.47. Bit probability error rate, P_{bit}, as a function of Γ for 8-PSK transmission over a non-minimum phase channel, using various equalization and data detection methods.

In these simulations, the error probability for a DFE-VA is in the range between 0.08 and 0.2 and it is not shown; the performance of the various receivers is illustrated in Figure 8.47.

By comparison of the results of Figure 8.46 and Figure 8.47 we observe that the RSSE and DFSE may be regarded as valid approximations of the VA as long as the overall channel impulse response is minimum phase.

8.18 Diversity combining techniques

The various equalization and data detection methods described in this chapter can be extended to systems in which the transmission of the information message takes place over several channels. The task of the receiver is to suitably combine the various received signals and to form a signal at the decision point with a better signal-to-noise ratio, or bit error probability, as compared to the case of transmission over a single channel.

These structures are widely used in radio systems that are subject to channel fading (see Section 4.6). Diversity is realized by using more antennas at the receiver and/or at the transmitter. The more uncorrelated are the channels, the higher is the diversity gain. In indoor and urban environments this requires a spacing between antennas of at least $\lambda/2$, whereas in rural environments the spacing may be a multiple of λ. In spread spectrum systems (see Chapter 10) the diversity is generated by the multipath channel itself and we speak of *multipath diversity*. In general, using suitable techniques, at the receiver it is possible to compose a signal in which the effect of fading is reduced.

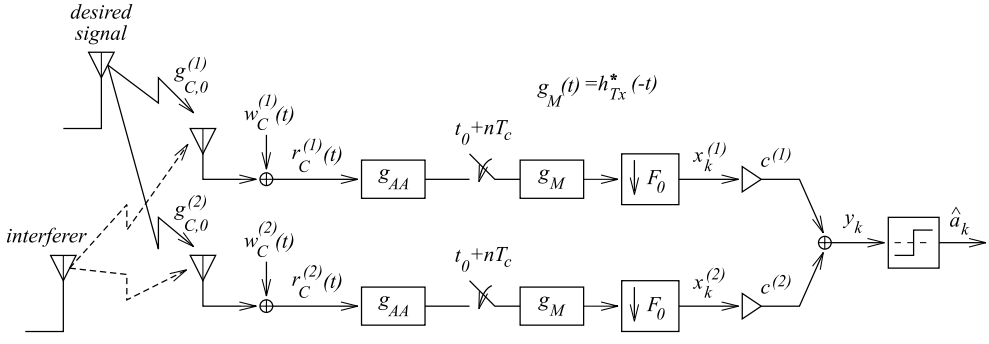


Figure 8.48. Receiver with two receive antennas for flat fading radio channels.

Antenna arrays

Let us consider the scheme of Figure 8.4. We generalize it to the case of two receive antennas: thus, we obtain the scheme of Figure 8.48, where the receive filter for each branch is the matched filter to the received pulse. For a *non-dispersive transmission channel*, assuming absence of ISI, for a single transmit antenna and $N_{A_{Rc}}$ receive antennas, at the i-th antenna branch the sampled signal, with sampling period equal to T, is given by

$$x_k^{(i)} = A\, a_k\, g_{C,0}^{(i)} + \tilde{w}_k^{(i)} + \tilde{v}_k^{(i)} \tag{8.445}$$

where a_k is the desired symbol, A is the amplitude of the desired pulse at the output of the downsampler (in the scheme of Figure 8.48 $A = E_h$, the energy of the transmitted pulse h_{Tx}), $g_{C,0}^{(i)}$ is a complex coefficient representing the flat fading channel with $E[|g_{C,0}^{(i)}|^2] = 1$, and $\{\tilde{w}_k^{(i)}\}$, $i = 1, \ldots, N_{A_{Rc}}$, are uncorrelated sequences of i.i.d. noise samples, having variance $\sigma^2_{\tilde{w}^{(i)}} = N_0 E_h$. The sample $\tilde{v}_k^{(i)}$ represents the interference on the i-th branch due to undesired signals using the same carrier as the desired signal. If N_I is the number of interfering signals, that are assumed synchronous with the desired signal, we have

$$\tilde{v}_k^{(i)} = \sum_{j=1}^{N_I} A^{(j)}\, a_k^{(j)}\, g_{C,0}^{(i,j)} \tag{8.446}$$

With regard to the j-th interfering signal, $a_k^{(j)}$ is the generic symbol of the transmitted message, $A^{(j)}$ is the amplitude of the interfering pulse at the output of the downsampler, and $g_{C,0}^{(i,j)}$ represents the flat fading channel between the j-th interfering antenna and the i-th receiving antenna.

We also assume that there is no Doppler spread. Therefore the channels are time invariant for the duration of the transmission. Finally, we recall the reference signal-to-noise ratio at the decision point for an ideal AWGN channel, $\gamma_{MF} = \frac{2E_h}{N_0}$, given by (7.113).

The signal at the decision point is usually a linear combination of samples given by (8.445), that is

$$y_k = \sum_{i=1}^{N_{A_{Rc}}} c^{(i)}\, x_k^{(i)} \tag{8.447}$$

where $\{c^{(i)}\}$, $i = 1, \ldots, N_{A_{Rc}}$, are suitable coefficients. In general, however, the combination can be made either directly on the received signals $\{r_C^{(i)}(t)\}$, $i = 1, \ldots, N_{A_{Rc}}$, thus realizing a *pre-detection combiner*, or on signals after the matched filter as in (8.447), realizing in this case a *post-detection combiner*.

We emphasize that the scheme of Figure 8.48 represents the simplest case, with only two antennas, of a structure, or array, that in general has $N_{A_{Rc}}$ antennas. Depending on the placement of elements, arrays are classified as: 1) *linear* if the elements are aligned on a straight line; 2) *circular* if the elements are placed on a circle; 3) *planar* if the elements are placed on a grid. More complex structures also exist.

Combining techniques

For the selection of the coefficients in (8.447) we consider two switching techniques and three combining techniques. To simplify the analysis we assume $\tilde{v}_k^{(i)}$ absent or included in $\tilde{w}_k^{(i)}$.

1. Selective combining. Only one of the received signals is selected. Let i_{SC} be the branch corresponding to the received signal with highest statistical power,

$$i_{SC} = \arg \max_{i \in \{1,\ldots,N_{A_{Rc}}\}} \mathtt{M}_{r_C^{(i)}} \tag{8.448}$$

where the different powers are estimated using a training sequence. In some cases, in place of the statistical power the bit error probability or the receive signal-to-noise ratio[8] is used as the decision parameter. Based on the decision (8.448), the receiver selects the antenna i_{SC} and consequently extracts the signal $x_k^{(i_{SC})}$ aligned in phase.

With reference to (8.447), this method is equivalent to selecting

$$c^{(i)} = \begin{cases} g_{C,0}^{(i)*} & i = i_{SC} \\ 0 & i \neq i_{SC} \end{cases} \tag{8.449}$$

and

$$\begin{aligned} y_k &= g_{C,0}^{(i_{SC})*}\, x_k^{(i_{SC})} \\ &= \left|g_{C,0}^{(i_{SC})}\right|^2 A\, a_k + g_{C,0}^{(i_{SC})*}\, \tilde{w}_k^{(i_{SC})} \end{aligned} \tag{8.450}$$

At the decision point, we have

$$\gamma_{SC} = \gamma_{MF} \left|g_{C,0}^{(i_{SC})}\right|^2 \tag{8.451}$$

[8] This parameter can be estimated by the estimate of the channel impulse response (see Appendix 3.B)

A variant of this technique consists in selecting only two or three signals; next, their combination takes place.

2. Switched combining. Another antenna is selected only when the statistical power of the signal, or equivalently the bit error probability, or the receive signal-to-noise ratio, drops below a given threshold. Once a new antenna is selected, the signal is processed as in the previous case.

3. Equal gain combining (EGC). In this case the signals are only aligned in phase; therefore we have

$$c^{(i)} = e^{-j \arg(g_{C,0}^{(i)})} = e^{j \arg(g_{C,0}^{(i)*})} \qquad i = 1, \ldots, N_{A_{Rc}} \tag{8.452}$$

It results in

$$y_k = A\, a_k \left(\sum_{i=1}^{N_{A_{Rc}}} \left| g_{C,0}^{(i)} \right| \right) + \sum_{i=1}^{N_{A_{Rc}}} c^{(i)}\, \tilde{w}_k^{(i)} \tag{8.453}$$

which yields

$$\gamma_{EGC} = \gamma_{MF}\, \frac{\left(\sum_{i=1}^{N_{A_{Rc}}} \left| g_{C,0}^{(i)} \right| \right)^2}{N_{A_{Rc}}} \tag{8.454}$$

This technique is often used in receivers of DPSK signals (see Section 6.5); in this case the combining is obtained by summing the differentially demodulated signals on the various branches.

4. Maximal ratio combining (MRC). We assume *absence of interferers*. The MRC criterion consists in maximizing the signal-to-noise ratio at the decision point. Substituting (8.445) in (8.447) and taking the expectation with respect to the message and the various noise signals, we get

$$\gamma = 2E_h^2\, \frac{\left| \sum_{i=1}^{N_{A_{Rc}}} c^{(i)}\, g_{C,0}^{(i)} \right|^2}{\sum_{i=1}^{N_{A_{Rc}}} \left| c^{(i)} \right|^2 \sigma_{\tilde{w}^{(i)}}^2} \tag{8.455}$$

Using the Schwarz inequality (see page 4), the signal-to-noise in (8.455) is maximized for

$$c^{(i)} = \mathcal{K}\, \frac{g_{C,0}^{(i)*}}{\sigma_{\tilde{w}^{(i)}}} \qquad i = 1, \ldots, N_{A_{Rc}} \tag{8.456}$$

where $\mathcal{K}$ is a constant.

Because the noise signals of the various branches have the same variance, the choice

$$c^{(i)} = \mathcal{K}\, g_{C,0}^{(i)*} \tag{8.457}$$

yields the maximum of γ, given by

$$\gamma_{MRC} = \gamma_{MF} \left(\sum_{i=1}^{N_{A_{Rc}}} \left| g_{C,0}^{(i)} \right|^2 \right) \tag{8.458}$$

Introducing the signal-to-noise ratio of the i-th branch

$$\gamma^{(i)} = \gamma_{MF} \left| g_{C,0}^{(i)} \right|^2 \tag{8.459}$$

(8.458) can be written as

$$\gamma_{MRC} = \sum_{i=1}^{N_{A_{Rc}}} \gamma^{(i)} \tag{8.460}$$

that is, the total signal-to-noise ratio is the sum of the signal-to-noise ratios of the individual branches.

It is interesting to note that the choice (8.457) is also the solution of the maximum likelihood criterion associated with signals (8.445).

In the case of uncorrelated channels with a Rayleigh statistic, the performance, in terms of bit error probability of the various combining techniques can be obtained analytically by applying the technique described in Section 6.12 [26, 27, 28].

5. Optimum combining (OC). The MRC criterion performs well in situations where the desired signal power is much greater than the power of the interfering signals; otherwise the effect of interference must be considered. In the OC criterion the ratio between the power of the desired signal and the power of the *interference plus noise* (SINR) is maximized.

In practice the coefficients $\{c^{(i)}\}$, $i = 1, \ldots, N_{A_{Rc}}$, are determined by the Wiener formulation, where y_k is the sample at the decision point and a_k is the desired symbol. Defining

$$\begin{aligned} \mathbf{x}_k &= [x_k^{(1)}, \ldots, x_k^{(N_{A_{Rc}})}]^T \\ \mathbf{c} &= [c^{(1)}, \ldots, c^{(N_{A_{Rc}})}]^T \end{aligned} \tag{8.461}$$

we have

$$y_k = \mathbf{x}_k^T\, \mathbf{c} \tag{8.462}$$

Recalling the expression of the autocorrelation matrix

$$\mathbf{R} = E[\mathbf{x}_k^*\, \mathbf{x}_k^T] \tag{8.463}$$

and the cross-correlation vector

$$\mathbf{p} = E[a_k\, \mathbf{x}_k^*] \tag{8.464}$$

the optimum solution is given by

$$\mathbf{c} = \mathbf{R}^{-1}\,\mathbf{p} \tag{8.465}$$

In many applications, starting from estimates of $\mathbf{R}$ and $\mathbf{p}$ obtained by equations similar to (2.130) and (2.131) (see also the covariance method of page 148), the estimate of $\mathbf{R}^{-1}$ is obtained directly with a computational complexity $O(N_{A_{Rc}}^3)$. This method is termed *direct matrix inversion* (DMI).

As an alternative, or next to a first solution through the DMI method, the LMS or RLS iterative methods may be used (see Chapter 3), that present the advantage of tracking the system variations.

A graphic method that illustrates the effectiveness of the MRC and OC techniques is the array radiation diagram [29], which shows a main lobe in the direction of the desired user and zeros along the direction of the interferers: in fact a receiver with $N_{A_{Rc}}$ antennas and N_I interferers has the same performance of a receiver with $(N_{A_{Rc}} - N_I)$ antennas and no interferers [30, 31]. However, if the interferers are known, cancellation methods give better performance.

Equalization and diversity

Techniques of the previous section can be extended to transmission over multipath channels using, in place of a single coefficient $c^{(i)}$ per antenna, a filter with N coefficients. Introducing the vector whose elements are the coefficients of the $N_{A_{Rc}}$ filters, that is

$$\mathbf{c} = [c_0^{(1)}, \ldots, c_{N-1}^{(1)}, c_0^{(2)}, \ldots, c_{N-1}^{(2)}, \ldots, c_{N-1}^{(N_{A_{Rc}})}]^T \tag{8.466}$$

and the input vector

$$\mathbf{x}_k = [x_k^{(1)}, \ldots, x_{k-N+1}^{(1)}, x_k^{(2)}, \ldots, x_{k-N+1}^{(2)}, \ldots, x_{k-N+1}^{(N_{A_{Rc}})}]^T \tag{8.467}$$

the solution is of the type (8.465).

Also in this case a DFE structure, or better an MLSD receiver, yields improved performance at the cost of a greater complexity. For an in-depth study we refer to the bibliography [32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49].

We point out that the optimum structure of a *combiner/linear equalizer* is given by a bank of filters, each matched to the overall receive pulse at the antenna output, whose outputs are then summed; the signal thus obtained is equalized by an LE or a DFE [50].

We also observe that in mobile radio systems, the use of multiple antennas generally occurs at the base-station, as the mobile is limited both in power consumption and dimension.

Diversity in transmission

We now discuss some diversity methods in which $N_{A_{Tx}}$ transmit antennas and one receive antenna are employed [51, 52, 53].

First, we distinguish close-loop methods, with feedback from receiver, from open-loop methods, without feedback. Obviously in the first case there is a return channel to communicate the selected parameters to the transmitter; the drawback of these methods is that, besides requiring a certain capacity, the parameters of the transmission channel are known with a certain delay.

1. Selection transmission diversity. As illustrated in Figure 8.49, the idea is to select in transmission the antenna that yields the best value of the *received* power, or bit error rate, or signal-to-noise ratio. To identify this antenna, a training sequence must be transmitted by each antenna; if the training sequences are orthogonal, they can be simultaneously transmitted. The receiver then communicates to the transmitter the antenna to be employed in the next *slot* or *frame* (see Section 6.13.2).

2. Transmit array. The signals sent over the various antennas are multiplied by coefficients $\{c^{(i)}\}$, $i = 1, \ldots, N_{A_{Tx}}$, computed at the receiver (see Figure 8.50). At the receiver the criterion can be the MRC or the OC.

In order not to increase the average transmitted power, the coefficients must be scaled so that

$$\sum_{i=1}^{N_{A_{Tx}}} \left| c^{(i)} \right|^2 = 1 \tag{8.468}$$

In [52] a method is presented to determine the coefficients which minimize the receiver bit error probability under a constraint on the total transmitted power or on the power transmitted by each antenna.

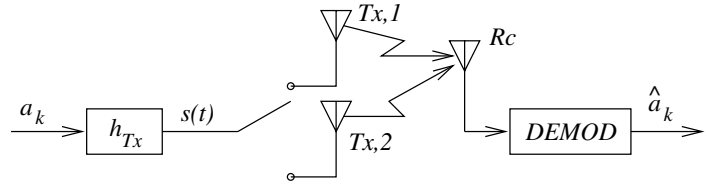


Figure 8.49. Selection transmission diversity.

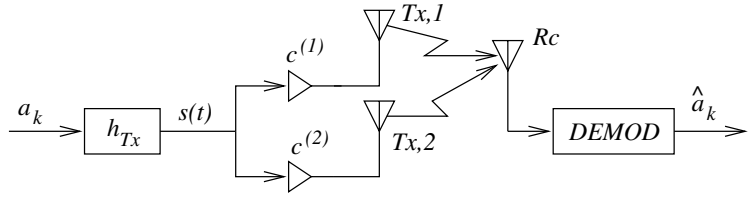


Figure 8.50. Transmit array.

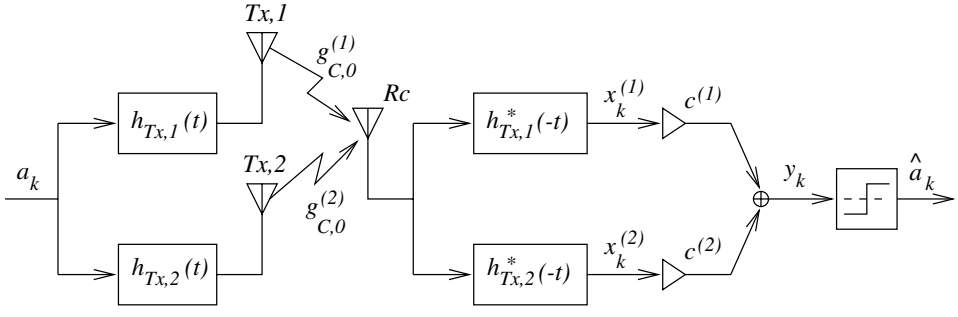


Figure 8.51. Orthogonal transmission diversity.

3. Orthogonal transmission diversity. The symbol a_k is modulated with two orthogonal pulses, $h_{Tx,1}(t)$ and $h_{Tx,2}(t)$, as illustrated in Figure 8.51.

The receiver employs two matched filters, that output the signals

$$x_k^{(i)} = A\, a_k\, g_{C,0}^{(i)} + \tilde{w}_k^{(i)} \qquad i = 1, 2 \tag{8.469}$$

Combining the various outputs using the coefficients

$$c^{(i)} = g_{C,0}^{(i)*} \tag{8.470}$$

we get

$$\begin{aligned} y_k &= c^{(1)}\, x_k^{(1)} + c^{(2)}\, x_k^{(2)} \\ &= A\, a_k (|g_{C,0}^{(1)}|^2 + |g_{C,0}^{(2)}|^2) + g_{C,0}^{(1)*}\, \tilde{w}_k^{(1)} + g_{C,0}^{(2)*}\, \tilde{w}_k^{(2)} \end{aligned} \tag{8.471}$$

If the noise signals $\tilde{w}_k^{(i)}$, $i = 1, \ldots, N_{A_{Tx}}$, are uncorrelated, and if each antenna can transmit a signal with maximum power, this scheme has the same performance of an MRC scheme, in terms of γ; for the same total transmitted power, instead, it loses $10 \log_{10} N_{A_{Tx}}$ dB. Another drawback lies in the use of at least two orthogonal signals per user. A variant of the system is proposed in [51].

4. Delay diversity. Let $s(t)$ be the modulated signal, as shown in Figure 8.52. The i-th antenna transmits the delayed signal

$$s(t - iT) \qquad i = 0, \ldots, N_{A_{Tx}} - 1 \tag{8.472}$$

At the receiver, the message is detected from the resulting signal with ISI by MLSD [54].

5. Time switched transmit diversity. As illustrated in Figure 8.53, the transmitter selects in turn the antenna on which to transmit a symbol sequence. The method is much simpler than the previous one, however, it requires the use of channel coding and interleaving (see Chapter 11) to recover the errors introduced by some channels.

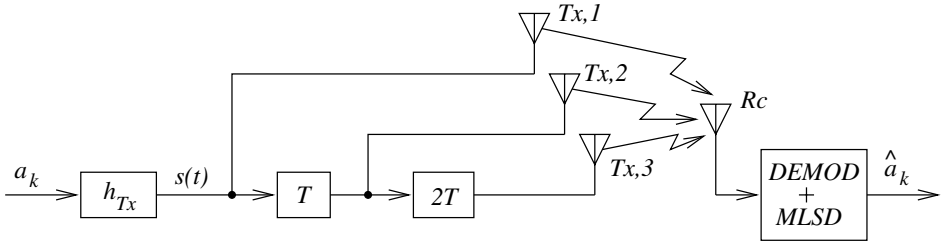


Figure 8.52. Delay diversity.

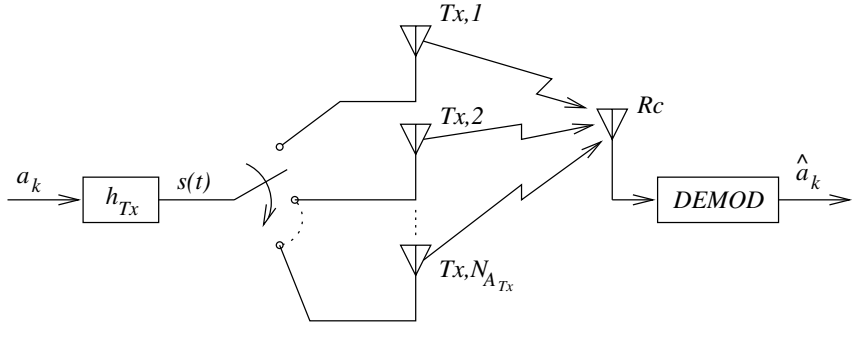


Figure 8.53. Time switched transmit diversity.

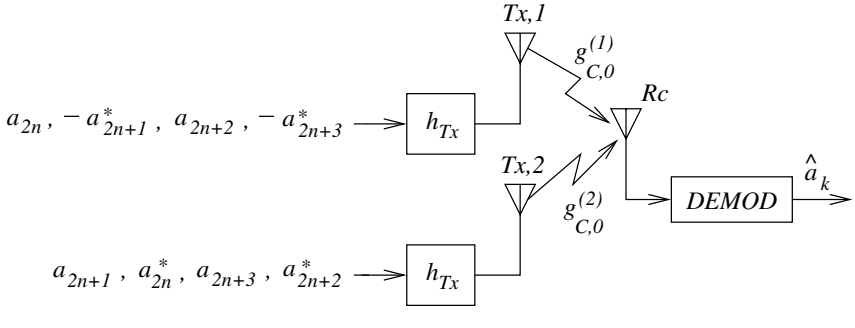


Figure 8.54. Space-time transmit diversity.

6. Space-time transmit diversity. The basic scheme is illustrated in Figure 8.54 [53]. The message $\{a_k\}$ is transmitted over two antennas: over antenna 1 the transmitted signal is modulated by the data sequence

$$a_{2n} \qquad -a^*_{2n+1}, \ldots \tag{8.473}$$

and over antenna 2 the transmitted data sequence is modulated by

$$a_{2n+1} \qquad a^*_{2n}, \ldots \tag{8.474}$$

For a receiver with matched filter to the pulse h_{Tx}, at the decision point we have the signal

$$x_k = \begin{cases} A(a_{2n}\, g_{C,0}^{(1)} + a_{2n+1}\, g_{C,0}^{(2)}) + \tilde{w}_{2n} & \text{for } k = 2n \\ A(-a_{2n+1}^{*}\, g_{C,0}^{(1)} + a_{2n}^{*}\, g_{C,0}^{(2)}) + \tilde{w}_{2n+1} & \text{for } k = 2n+1 \end{cases} \tag{8.475}$$

Assuming the channels are known, we consider the combination of the samples

$$y_{2n} = g_{C,0}^{(1)*}\, x_{2n} + g_{C,0}^{(2)}\, x_{2n+1}^{*} \tag{8.476}$$

and

$$y_{2n+1} = g_{C,0}^{(2)*}\, x_{2n} - g_{C,0}^{(1)}\, x_{2n+1}^{*} \tag{8.477}$$

It is easy to verify that

$$y_k = \begin{cases} a_{2n}(|g_{C,0}^{(1)}|^2 + |g_{C,0}^{(2)}|^2) + g_{C,0}^{(1)*}\, \tilde{w}_{2n} + g_{C,0}^{(2)}\, \tilde{w}_{2n+1}^{*} & \text{for } k = 2n \\ a_{2n+1}(|g_{C,0}^{(1)}|^2 + |g_{C,0}^{(2)}|^2) + g_{C,0}^{(2)*}\, \tilde{w}_{2n} - g_{C,0}^{(1)}\, \tilde{w}_{2n+1}^{*} & \text{for } k = 2n+1 \end{cases} \tag{8.478}$$

A comparison with the MRC technique leads to the same considerations made on (8.471).

We conclude this section by mentioning diversity techniques with more than one transmit and receive antenna, called *space-time coding* techniques, whereby the message is coded by using suitable channel codes [55, 56, 57].

Bibliography

[1] G. Ungerboeck, "Fractionally tap-spacing and consequences for clock recovery in data modems", *IEEE Trans. on Communications*, Aug. 1976.

[2] R. D. Gitlin and S. B. Weinstein, "Fractionally-spaced equalization: an improved digital transversal equalizer", *Bell System Technical Journal*, vol. 60, Feb. 1981.

[3] J. E. Mazo and J. Salz, "Probability of error quadratic detector", *Bell System Technical Journal*, vol. 44, Nov. 1965.

[4] J. G. Proakis, *Digital communications*. New York: McGraw-Hill, 3rd ed., 1995.

[5] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", *IEEE Proceedings*, vol. 77, pp. 257–285, Feb. 1989.

[6] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate", *IEEE Trans. on Information Theory*, vol. 20, pp. 284–287, Mar. 1974.

[7] A. J. Viterbi, "An intuitive justification and simplified implementation of the MAP decoder for convolutional codes", *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 260–264, Feb. 1998.

[8] G. Ungerboeck, "Adaptive maximum likelihood receiver for carrier modulated data transmission systems", *IEEE Trans. on Communications*, vol. 22, pp. 624–635, May 1974.

[9] G. E. Bottomley and S. Chennakeshu, "Unification of MLSE receivers and extension to time-varying channels", *IEEE Trans. on Communications*, vol. 46, pp. 464–472, Apr. 1998.

[10] D. G. Messerschmitt and E. A. Lee, *Digital communication*. Boston, MA: Kluwer Academic Publishers, 2nd ed., 1994.

[11] K. M. Chugg and A. Anastasopoulos, "On symbol error probability bounds for ISI-like channels", *IEEE Trans. on Communications*, vol. 49, pp. 1704–1709, Oct. 2001.

[12] W. U. Lee and F. S. Hill, "A maximum-likelihood sequence estimator with decision feedback equalization", *IEEE Trans. on Communications*, vol. 25, pp. 971–979, Sept. 1977.

[13] G. J. Foschini, "A reduced-state variant of maximum-likelihood sequence detection attaining optimum performance for high signal-to-noise ratio performance", *IEEE Trans. on Information Theory*, vol. 24, pp. 505–509, Sept. 1977.

[14] J. B. Anderson and S. Mohan, "Sequential coding algorithms: a survey and cost analysis", *IEEE Trans. on Communications*, vol. 32, pp. 169–176, Feb. 1984.

[15] M. V. Eyuboglu and S. U. H. Qureshi, "Reduced-state sequence estimator with set partitioning and decision feedback", *IEEE Trans. on Communications*, vol. 36, pp. 13–20, Jan. 1988.

[16] M. V. Eyuboglu, "Reduced-state sequence estimator for coded modulation on intersymbol interference channels", *IEEE Trans. on Communications*, vol. 7, pp. 989–995, Aug. 1989.

[17] R. E. Kamel and Y. Bar-Ness, "Reduced-complexity sequence estimation using state partitioning", *IEEE Trans. on Communications*, vol. 44, pp. 1057–1063, Sept. 1996.

[18] W. Sheen and G. L. Stuber, "Error probability for reduced-state sequence estimation", *IEEE Journal on Selected Areas in Communications*, vol. 10, pp. 571–578, April 1992.

[19] R. Raheli, A. Polydoros, and C.-K. Tzou, "Per-survivor processing: a general approach to MLSE in uncertain environments", *IEEE Trans. on Communications*, vol. 43, pp. 354–364, Feb.–Apr. 1995.

[20] K. H. Müller and D. A. Spaulding, "Cyclic equalization: a new rapidly converging equalization technique for synchronous data communication", *Bell System Technical Journal*, vol. 54, pp. 369–406, Feb. 1975.

[21] P. R. Chevillat, D. Maiwald, and G. Ungerboeck, "Rapid training of a voiceband data-modem receiver employing an equalizer with fractional-T spaced coefficients", *IEEE Trans. on Communications*, vol. COM–35, pp. 869–876, Sept. 1987.

[22] G. Ungerboeck, D. Maiwald, H. P. Kaeser, and P. R. Chevillat, "The SP16 signal processor", in *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 16.2.1–16.2.4, Mar. 1984.

[23] H. Sari, G. Karam, and I. Jeanclaude, "An analysis of orthogonal frequency-division multiplexing for mobile radio applications", in *Proc. 1994 IEEE Vehicular Technology Conference*, vol. 3, Stockholm, Sweden, pp. 1635–1639, June 8–10 1994.

[24] M. Huemer, L. Reindl, A. Springer, and R. Weigel, "Frequency domain equalization of linear polyphase channels", in *Proc. 2000 IEEE Vehicular Technology Conference*, Tokyo, Japan, May 2000.

[25] N. Benvenuto and S. Tomasin, "On the comparison between OFDM and single carrier modulation with a DFE using a frequency domain feedforward filter", *IEEE Trans. on Communications*, 2002.

[26] T. S. Rappaport, *Wireless communications: principles and practice*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[27] G. L. Stuber, *Principles of mobile communication*. Norwell, MA: Kluwer Academic Publishers, 1996.

[28] T. Eng, N. Kong, and L. B. Milstein, "Comparison of diversity combining techniques for Rayleigh-fading channels", *IEEE Trans. on Communications*, vol. 44, pp. 1117–1129, Sept. 1996.

[29] C. G. Someda, *Electromagnetic waves*. London: Chapman & Hall, 1998.

[30] J. H. Winters, "Optimum combining in digital mobile radio with cochannel interference", *IEEE Journal on Selected Areas in Communications*, vol. 2, pp. 528–539, July 1984.

[31] J. H. Winters, J. Salz, and R. D. Gitlin, "The impact of antenna diversity on the capacity of wireless communication systems", *IEEE Trans. on Communications*, vol. 42, pp. 1740–1751, Feb.–Apr. 1994.

[32] W. C. Jakes, *Microwave mobile communications*. New York: IEEE Press, 1993.

[33] B. Glance and L. J. Greenstein, "Frequency-selective fading in digital mobile radio with diversity combining", *IEEE Trans. on Communications*, vol. 31, pp. 1085–1094, Sept. 1983.

[34] W. K. Kennedy, L. J. Greenstein, and M. V. Clark, "Optimum linear diversity receivers for mobile communications", *IEEE Trans. on Vehicular Technology*, vol. 43, pp. 47–56, Feb. 1994.

[35] M. V. Clark, L. J. Greenstein, W. K. Kennedy, and M. Shafi, "MMSE diversity combining for wide-band digital cellular radio", *IEEE Trans. on Communications*, vol. 40, pp. 1128–1135, June 1992.

[36] R. T. Compton, *Adaptive antennas: concepts and performance*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[37] D. D. Falconer, M. Abdulrahman, N. W. K. Lo, B. R. Petersen, and A. U. H. Sheikh, "Advances in equalization and diversity for portable wireless systems", *Digital Signal Processing*, vol. 3, pp. 148–162, 1993.

[38] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas", *Bell System Technical Journal*, vol. 1, pp. 41–59, Autumn 1996.

[39] L. C. Godara, "Application of antenna arrays to mobile communications—Part I: Performance improvement, feasibility and system considerations", *IEEE Proceedings*, vol. 85, pp. 1031–1060, July 1997.

[40] L. C. Godara, "Application of antenna arrays to mobile communications—Part II: Beam-forming and direction-of-arrival considerations", *IEEE Proceedings*, vol. 85, pp. 1195–1245, Aug. 1997.

[41] I. P. Kirsteins and D. W. Tufts, "Adaptive detection using low rank approximation to a data matrix", *IEEE Trans. on Aerospace and Electronic Systems*, vol. 30, pp. 55–67, Jan. 1994.

[42] R. Kohno, H. Imai, M. Hatori, and S. Pasupathy, "Combination of an adaptive array antenna and a canceller of interference for direct-sequence spread-spectrum multiple-access system", *IEEE Journal on Selected Areas in Communications*, vol. 8, pp. 675–682, May 1990.

[43] W. C. Y. Lee, "Effects on correlation between two mobile radio base-station antennas", *IEEE Trans. on Communications*, vol. 21, pp. 1214–1224, Nov. 1973.

[44] R. A. Monzingo and T. W. Miller, *Introduction to adaptive arrays*. New York: John Wiley & Sons, 1980.

[45] A. J. Paulraj and C. B. Papadias, "Space-time processing for wireless communications", *IEEE Signal Processing Magazine*, vol. 14, pp. 49–83, Nov. 1997.

[46] J. Salz and J. H. Winters, "Effect of fading correlation on adaptive arrays in digital wireless communications", *IEEE Trans. on Vehicular Technology*, vol. 43, pp. 1049–1057, Nov. 1994.

[47] G. Tsoulos, M. Beach, and J. McGeehan, "Wireless personal communications for the 21st century: European technological advances in adaptive antennas", *IEEE Communications Magazine*, vol. 35, pp. 102–109, Sept. 1997.

[48] B. Widrow, P. E. Mantey, L. J. Griffiths, and B. B. Goode, "Adaptive antenna systems", *IEEE Proceedings*, vol. 55, pp. 2143–2159, Dec. 1967.

[49] J. H. Winters, "Smart antennas for wireless systems", *IEEE Personal Communications Magazine*, vol. 5, pp. 23–27, Feb. 1998.

[50] P. Balaban and J. Salz, "Dual diversity combining and equalization in digital cellular mobile radio", *IEEE Trans. on Vehicular Technology*, vol. 40, pp. 342–354, May 1991.

[51] K. Rohani, M. Harrison, and K. Kuchi, "A comparison of base station transmit diversity methods for third generation cellular standards", in *Proc. 1999 IEEE Vehicular Technology Conference*, Houston, USA, pp. 351–355, May 16–20 1999.

[52] J. K. Cavers, "Optimized use of diversity modes in transmitter diversity systems", in *Proc. 1999 IEEE Vehicular Technology Conference*, Houston, USA, pp. 1768–1773, May 16–20 1999.

[53] S. A. Alamouti, "A simple transmit diversity technique for wireless communications", *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1451–1458, Oct. 1998.

[54] J. H. Winters, "The diversity gain of transmit diversity in wireless systems with Rayleigh fading", *IEEE Trans. on Vehicular Technology*, vol. 47, pp. 119–123, Feb. 1998.

[55] V. Tarokh, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction", *IEEE Trans. on Information Theory*, vol. 44, pp. 744–765, Mar. 1998.

[56] A. Naguib, V. Tarokh, N. Seshadri, and A. R. Calderbank, "A space-time coding modem for high data rate wireless communications", *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 1459–1478, Oct. 1998.

[57] V. Tarokh, A. Naguib, N. Seshadri, and A. R. Calderbank, "Combined array processing and space-time coding", *IEEE Trans. on Information Theory*, vol. 45, pp. 1121–1128, May 1999.

[58] L. E. Franks, *Signal theory*. Stroudsburg, PA: Dowden and Culver, revised ed., 1981.

[59] S. Benedetto and E. Biglieri, *Principles of digital transmission with wireless applications*. New York: Kluwer Academic Publishers, 1999.

[60] N. Al-Dhahir and J. M. Cioffi, "MMSE decision-feedback equalizers: finite-length results", *IEEE Trans. on Information Theory*, vol. 41, pp. 961–975, July 1995.

Appendix 8.A *Calculus of variations and receiver optimization*

The first part of this appendix introduces the calculus of variations [58]. This technique will be applied to solve two optimization problems as described in the second part of the appendix.

8.A.1 Calculus of variations

The minimization of a functional[9] with respect to a continuous-time signal differs from the classical problem of the Wiener filter because the signal cannot be represented by a finite number of coefficients.

In this section, using the definition of a complete basis for continuous-time signals, we introduce the gradient of linear and quadratic functionals.

Linear functional

Definition 8.2
Let $x(t)$, $t \in \Re$, be a real-valued signal. We define *linear functional* of x the following integral:

$$\mathcal{L}_x = \int_{-\infty}^{+\infty} g(t)x(t)\,dt \tag{8.479}$$

where $g(t)$ is also a real-valued signal.

By (1.20) we represent the signal x by referring to a complete orthornormal basis $\{\phi_i(t)\}$, $i \in \mathcal{I}$, with $\mathcal{I}$ finite or numerable set, that is

$$x(t) = \sum_{i\in\mathcal{I}} x_i\phi_i(t) \tag{8.480}$$

where

$$x_i = \langle x(t), \phi_i(t)\rangle = \int_{-\infty}^{+\infty} x(t)\phi_i(t)\,dt \qquad i \in \mathcal{I} \tag{8.481}$$

are the signal components with respect to the basis.

Substitution of (8.480) in (8.479) yields

$$\begin{aligned}\mathcal{L}_x &= \int_{-\infty}^{+\infty} g(t)\left[\sum_{i\in\mathcal{I}} x_i\phi_i(t)\right]dt \\ &= \sum_{i\in\mathcal{I}} x_i\left[\int_{-\infty}^{+\infty} g(t)\phi_i(t)\,dt\right]\end{aligned} \tag{8.482}$$

[9] A functional is a map that assigns to a signal $x(t)$, $t \in \Re$, a complex number.

where we assume the order between the summation and integration can be exchanged. Let

$$g_i = \langle g(t), \phi_i(t) \rangle = \int_{-\infty}^{+\infty} g(t)\phi_i(t)\,dt \qquad i \in \mathcal{I} \tag{8.483}$$

If we introduce the vectors associated with the signals

$$x(t)t \in \Re \longleftrightarrow \mathbf{x} = [\ldots, x_{-1}, x_0, x_1, \ldots]^T \tag{8.484}$$

$$g(t)t \in \Re \longleftrightarrow \mathbf{g} = [\ldots, g_{-1}, g_0, g_1, \ldots]^T \tag{8.485}$$

(8.482) becomes

$$\mathcal{L}_x = \mathbf{x}^T\mathbf{g} \tag{8.486}$$

Observing (2.24), the gradient of (8.486) is

$$\nabla_{\mathbf{x}}\mathcal{L}_x = \mathbf{g} \tag{8.487}$$

whose i-th component is equal to

$$[\nabla_{\mathbf{x}}\mathcal{L}_x]_i = g_i \tag{8.488}$$

We now introduce the gradient as a continuous-time signal,

$$\begin{aligned} \nabla_x\mathcal{L}_x &= \sum_{i\in\mathcal{I}} [\nabla_{\mathbf{x}}\mathcal{L}_x]_i \phi_i(t) \\ &= \sum_{i\in\mathcal{I}} g_i\phi_i(t) \\ &= g(t) \end{aligned} \tag{8.489}$$

using (8.485).

As we will see later, it is convenient to express the functional in the frequency domain; applying the Parseval theorem, we get

$$\mathcal{L}_x = \int_{-\infty}^{+\infty} g(t)x(t)\,dt = \int_{-\infty}^{+\infty} \mathcal{G}(f)\mathcal{X}^*(f)\,df = \mathcal{L}_{\mathcal{X}} \tag{8.490}$$

therefore using (8.489) in the frequency domain we have

$$\nabla_{\mathcal{X}}\mathcal{L}_{\mathcal{X}} = \mathcal{G}(f) \tag{8.491}$$

Quadratic functional

Definition 8.3
Let $x(t)$, $t \in \Re$, be a real-valued signal. We define *quadratic functional* of x the integral

$$\mathcal{Q}_x = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x(t)A(t,\tau)x(\tau)\,d\tau\,dt \tag{8.492}$$

where $A(t,\tau)$ is a real-valued signal, called *time operator*.

As in the previous case, if we represent the signal x by a complete orthornormal basis we obtain

$$\mathcal{Q}_x = \mathbf{x}^T \mathbf{C} \mathbf{x} \tag{8.493}$$

where $\mathbf{C}$ is the matrix with coefficients

$$c_{i,j} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \phi_i(t) A(t,\tau) \phi_j(\tau)\, d\tau\, dt \qquad i, j \in \mathcal{I} \tag{8.494}$$

Defining

$$A_i(\tau) = \langle A(t,\tau), \phi_i(t) \rangle = \int_{-\infty}^{+\infty} A(t,\tau) \phi_i(t)\, dt \qquad j \in \mathcal{I} \tag{8.495}$$

then the operator $A(t,\tau)$ can be rewritten as

$$A(t,\tau) = \sum_{i \in \mathcal{I}} A_i(\tau) \phi_i(t) \tag{8.496}$$

Analogously, defining

$$A'_j(t) = \langle A(t,\tau), \phi_j(\tau) \rangle = \int_{-\infty}^{+\infty} A(t,\tau) \phi_j(\tau)\, d\tau \qquad j \in \mathcal{I} \tag{8.497}$$

then

$$A(t,\tau) = \sum_{j \in \mathcal{I}} A'_j(t) \phi_j(\tau) \tag{8.498}$$

The gradient of (8.493) is not equal to that computed in (2.30): in fact, in general, the matrix $\mathbf{C}$ is not symmetric and the vector $\mathbf{x}$ is real-valued. Hence, we obtain

$$\nabla_{\mathbf{x}} \mathcal{Q}_x = \mathbf{C}\mathbf{x} + (\mathbf{x}^T \mathbf{C})^T = \mathbf{C}\mathbf{x} + \mathbf{C}^T \mathbf{x} = (\mathbf{C} + \mathbf{C}^T)\mathbf{x} \tag{8.499}$$

In particular the i-th component of the gradient is given by

$$[\nabla_{\mathbf{x}} \mathcal{Q}_x]_i = \sum_{j \in \mathcal{I}} (c_{i,j} + c_{j,i}) x_j \tag{8.500}$$

In conclusion, the gradient, as a continuous-time signal, is expressed as

$$\begin{aligned} \nabla_x \mathcal{Q}_x &= \sum_{i \in \mathcal{I}} [\nabla_{\mathbf{x}} \mathcal{Q}_x]_i \phi_i(t) \\ &= \sum_{i \in \mathcal{I}} \left[\sum_{j \in \mathcal{I}} (c_{i,j} + c_{j,i}) x_j \right] \phi_i(t) \end{aligned} \tag{8.501}$$

Substitution of (8.494) in (8.501) yields

$$\nabla_x \mathcal{Q}_x = \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{I}} \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} [\phi_i(\eta)A(\eta,\tau)\phi_j(\tau) + \phi_j(\eta)A(\eta,\tau)\phi_i(\tau)\,d\tau\,d\eta]x_j\phi_i(t)$$

$$= \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{I}} \left[\int_{-\infty}^{+\infty}\left(\int_{-\infty}^{+\infty}\phi_i(\eta)[A(\eta,\tau)+A(\tau,\eta)]\,d\eta\right)\phi_j(\tau)\,d\tau\right]x_j\phi_i(t) \tag{8.502}$$

Substituting (8.495) and (8.498) in the above equation, changing variables, and using (8.496) and (8.498), we obtain

$$\begin{aligned}\nabla_x \mathcal{Q}_x &= \sum_{i\in\mathcal{I}}\sum_{j\in\mathcal{I}} \left[\int_{-\infty}^{+\infty}[A_i(\tau)+A_i'(\tau)]\phi_j(\tau)\,d\tau\right]x_j\phi_i(t)\\ &= \int_{-\infty}^{+\infty}\left[\sum_{i\in\mathcal{I}}[A_i(\tau)+A_i'(\tau)]\phi_i(t)\right]\left[\sum_{j\in\mathcal{I}}x_j\phi_j(\tau)\right]d\tau \\ &= \int_{-\infty}^{+\infty}[A(t,\tau)+A(\tau,t)]x(\tau)\,d\tau\end{aligned} \tag{8.503}$$

Also in this case we express the quadratic functional in the frequency domain by applying the Parseval theorem:

$$\mathcal{Q}_x = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\mathcal{X}(f)\mathcal{B}(f,\nu)\mathcal{X}^*(\nu)\,df\,d\nu = \mathcal{Q}_{\mathcal{X}} \tag{8.504}$$

where $\mathcal{B}(f,\nu)$, called *frequency operator*, is the two-dimensional Fourier transform of $A(t,\tau)$,

$$\mathcal{B}(f,\nu) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}A(t,\tau)e^{-j2\pi ft}e^{+j2\pi\nu\tau}\,dt\,d\tau \tag{8.505}$$

From (8.503) the gradient of the quadratic cost function becomes

$$\nabla_{\mathcal{X}}\mathcal{Q}_{\mathcal{X}} = \int_{-\infty}^{+\infty}[\mathcal{B}(f,\nu)+\mathcal{B}^*(\nu,f)]\mathcal{X}(\nu)\,d\nu \tag{8.506}$$

To conclude, we give an example of a quadratic functional that will be useful in the next subsection.

Example 8.A.1
We consider the following quadratic functional in the frequency domain:

$$\mathcal{Q}_{\mathcal{X}} = \int_{-\infty}^{+\infty}\mathcal{P}(f)|\mathcal{X}(f)|^2\,df \tag{8.507}$$

Defining $\mathtt{r}(\tau) = \mathcal{F}^{-1}[\mathcal{P}(f)]$, it can be verified that the corresponding time and frequency operators are

$$A(t, \tau) = \mathtt{r}(t - \tau) \tag{8.508}$$

$$\mathcal{B}(f, \nu) = \mathcal{P}(f)\delta(f - \nu) \tag{8.509}$$

Therefore from (8.506) the gradient of the functional (8.507) is given by

$$\nabla_{\mathcal{X}} \mathcal{Q}_{\mathcal{X}} = 2\mathcal{P}(f)\mathcal{X}(f) \tag{8.510}$$

8.A.2 Receiver optimization

Using the results of the previous section, we are now able to prove equation (8.11). Referring to the scheme of Figure 7.12 reproduced in Figure 8.55 with $q_C(t) = h_{Tx} * g_C(t)$, we need to solve the following problem.

Given $\mathcal{H}_{Tx}(f)$ and $\mathcal{G}_C(f)$, that is

$$\mathcal{Q}_C(f) = \mathcal{H}_{Tx}(f)\mathcal{G}_C(f) \tag{8.511}$$

we desire to determine $\mathcal{G}_{Rc}(f)$ that minimizes the cost function (8.10)

$$J = E[|y_k - a_k|^2] \tag{8.512}$$

The following assumptions are made.

1. The sequence $\{a_k\}$ is WSS, with zero mean and autocorrelation $\mathtt{r}_a(n) = E[a_k a^*_{k-n}]$.
2. The noise $w_C(t)$ is WSS with zero mean and spectral density $\mathcal{P}_{w_C}(f)$.
3. $\{a_n\}$ and $w_C(t)$ are statistically independent.

From (7.65), the signal at the sampler output is given by

$$y_k = \sum_{i=-\infty}^{+\infty} a_i h_{k-i} + w_{R,k} \tag{8.513}$$

where, from (7.67),

$$h_i = q_R(t_0 + iT) \tag{8.514}$$

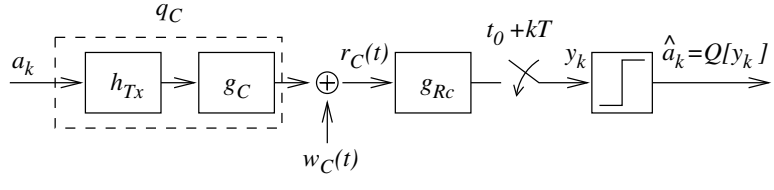


Figure 8.55. Baseband transmission system.

with Fourier transform

$$\mathcal{F}[h_i] = \sum_{i=-\infty}^{+\infty} h_i e^{-j2\pi f i T} = \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \mathcal{H}\left(f - \frac{\ell}{T}\right) \tag{8.515}$$

using the relations

$$\mathcal{H}(f) = \mathcal{Q}_R(f) e^{j2\pi f t_0} \tag{8.516}$$

and, from (7.48),

$$\mathcal{Q}_R(f) = \mathcal{Q}_C(f) \mathcal{G}_{Rc}(f) \tag{8.517}$$

Moreover, the variance of $\{w_{R,k}\}$ is given by (7.75).

Substitution of (8.513) in the expression of J yields

$$\begin{aligned} J &= E\left[\left|\sum_{i=-\infty}^{+\infty} a_i h_{k-i} + w_{R,k} - a_k\right|^2\right] \\ &= \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} E[a_i a_j^* h_{k-i} h_{k-j}^*] + \sigma_{w_R}^2 + \sigma_a^2 + 2 \sum_{i=\infty}^{+\infty} E[a_i h_{k-i} w_{R,k}^*] \\ &\quad - 2 \sum_{i=-\infty}^{+\infty} E[a_i h_{k-i} a_k^*] - 2E[w_{R,k} a_k^*] \end{aligned} \tag{8.518}$$

From the above assumptions, the fourth and the sixth term in the last expression are identically zero; therefore

$$J = \sigma_{w_R}^2 + \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} \mathrm{r}_a(i-j) h_{k-i} h_{k-j}^* - 2 \sum_{i=-\infty}^{+\infty} \mathrm{r}_a(i-k) h_{k-i} + \sigma_a^2 \tag{8.519}$$

With the change of indices $p = k - i$ and $q = k - j$, and being J real, we get

$$\begin{aligned} J &= \sigma_{w_R}^2 + \sum_{p=-\infty}^{+\infty} \sum_{q=-\infty}^{+\infty} \mathrm{r}_a(q-p) h_p h_q^* - 2 \sum_{p=-\infty}^{+\infty} \mathrm{r}_a(-p) h_p + \sigma_a^2 \\ &= \sigma_{w_R}^2 + \sum_{p=-\infty}^{+\infty} \sum_{q=-\infty}^{+\infty} \mathrm{r}_a(p-q) h_p^* h_q - 2 \sum_{p=-\infty}^{+\infty} \mathrm{r}_a(p) h_p^* + \sigma_a^2 \end{aligned} \tag{8.520}$$

It is convenient to express the above terms in the frequency domain, using the following relations:

- power spectral density of the message

$$\mathcal{P}_a(f) = T \sum_{n=-\infty}^{+\infty} \mathrm{r}_a(n) e^{-j2\pi f n T} \tag{8.521}$$

- coefficients of the sampled impulse response,

$$h_i = \int_{-\infty}^{+\infty} \mathcal{H}(f) e^{j2\pi f i T}\, df \tag{8.522}$$

where $\mathcal{H}(f) = \mathcal{F}[h(t)]$.

Now we rewrite the three terms on the right-hand side of (8.520).

1. Recalling that the noise is filtered only by the receive filter, we have

$$\sigma_{w_R}^2 = \int_{-\infty}^{+\infty} \mathcal{P}_{w_C}(f) |\mathcal{G}_{Rc}(f)|^2\, df = \int_{-\infty}^{+\infty} \mathcal{P}_{w_C}(f) \frac{|\mathcal{H}(f)|^2}{|\mathcal{Q}_C(f)|^2}\, df \tag{8.523}$$

2. From (8.521) and (8.522), using (8.515), it turns out

$$\begin{aligned}
&\sum_{p=-\infty}^{+\infty} \sum_{q=-\infty}^{+\infty} \mathrm{r}_a(p-q) h_p^* h_q \\
&\quad = \sum_{p=-\infty}^{+\infty} \sum_{q=-\infty}^{+\infty} \mathrm{r}_a(p-q) \left[\int_{-\infty}^{+\infty} \mathcal{H}^*(f) e^{-j2\pi f p T}\, df \right] h_q \\
&\quad = \int_{-\infty}^{+\infty} \mathcal{H}^*(f) \sum_{q=-\infty}^{+\infty} \left[\sum_{p=-\infty}^{+\infty} \mathrm{r}_a(p-q) e^{-j2\pi f p T} \right] h_q\, df \\
&\quad = \frac{1}{T} \int_{-\infty}^{+\infty} \mathcal{H}^*(f) \sum_{q=-\infty}^{+\infty} \mathcal{P}_a(f) e^{-j2\pi f q T} h_q\, df \\
&\quad = \frac{1}{T^2} \int_{-\infty}^{+\infty} \mathcal{H}^*(f) \mathcal{P}_a(f) \sum_{\ell=-\infty}^{+\infty} \mathcal{H}\left(f - \frac{\ell}{T} \right) df
\end{aligned} \tag{8.524}$$

3. Using again (8.522), we have

$$\begin{aligned}
-2 \sum_{p=-\infty}^{+\infty} \mathrm{r}_a(p) h_p^* &= -2 \sum_{p=-\infty}^{+\infty} \mathrm{r}_a(p) \left[\int_{-\infty}^{+\infty} \mathcal{H}^*(f) e^{-j2\pi p T}\, df \right] \\
&= -\frac{2}{T} \int_{-\infty}^{+\infty} \mathcal{H}^*(f) \mathcal{P}_a(f)\, df
\end{aligned} \tag{8.525}$$

Substitution of (8.523), (8.524), and (8.525) in (8.520) yields

$$\begin{aligned}
J = &\int_{-\infty}^{+\infty} \mathcal{P}_{w_C}(f) \frac{|\mathcal{H}(f)|^2}{|\mathcal{Q}_C(f)|^2}\, df \\
&+ \frac{1}{T^2} \int_{-\infty}^{+\infty} \mathcal{H}^*(f) \mathcal{P}_a(f) \sum_{\ell=-\infty}^{+\infty} \mathcal{H}\left(f - \frac{\ell}{T} \right) df \\
&- \frac{2}{T} \int_{-\infty}^{+\infty} \mathcal{H}^*(f) \mathcal{P}_a(f)\, df \\
&+ \sigma_a^2
\end{aligned} \tag{8.526}$$

We now consider each term of (8.526).

a) The first term is a quadratic cost function. From (8.510) the gradient with respect to $\mathcal{H}(f)$ is given by

$$2\frac{\mathcal{P}_{w_C}(f)}{|\mathcal{Q}_C(f)|^2}\mathcal{H}(f) \tag{8.527}$$

b) The second term is also a quadratic functional, with the difference with respect to the previous term that a transformation given by a periodic repetition is performed on $\mathcal{H}(f)$; it can be proven that the gradient with respect to $\mathcal{H}(f)$ is

$$\frac{2}{T^2}\mathcal{P}_a(f)\sum_{\ell=-\infty}^{+\infty}\mathcal{H}\left(f-\frac{\ell}{T}\right) \tag{8.528}$$

c) The third term is a linear functional; from (8.491) the gradient is

$$-\frac{2}{T}\mathcal{P}_a(f) \tag{8.529}$$

d) The fourth term is a constant; therefore the gradient vanishes.

Considering that, if $\mathcal{Q}_C(f)$ is known, we obtain

$$\min_{\mathcal{G}_{Rc}(f)} J = \min_{\mathcal{H}(f)=\mathcal{Q}_C(f)\mathcal{G}_{Rc}(f)e^{j2\pi f t_0}} J \tag{8.530}$$

the optimum value of $\mathcal{H}(f)$ is obtained by setting

$$\nabla_{\mathcal{H}} J = 0 \tag{8.531}$$

Extending the variational analysis to any h, we obtain

$$\nabla_{\mathcal{H}} J = 2\frac{\mathcal{P}_{w_C}(f)}{|\mathcal{Q}_C(f)|^2}\mathcal{H}(f) + \frac{2}{T^2}\mathcal{P}_a(f)\sum_{\ell=-\infty}^{+\infty}\mathcal{H}\left(f-\frac{\ell}{T}\right) - \frac{2}{T}\mathcal{P}_a(f) = 0 \tag{8.532}$$

Now we introduce the signal $\mathcal{G}(f)$ that coincides with the first term of (8.532),

$$\mathcal{G}(f) = \frac{\mathcal{P}_{w_C}(f)}{|\mathcal{Q}_C(f)|^2}\mathcal{H}(f) \tag{8.533}$$

As both the second and the third term on the left-hand side of (8.532) are periodic functions of period $1/T$, then also $\mathcal{G}(f)$ is a periodic function of period $1/T$. Substituting therefore (8.533) in (8.532), it must be

$$\mathcal{G}(f) + \frac{1}{T^2}\mathcal{P}_a(f)\sum_{\ell=-\infty}^{+\infty}\mathcal{G}\left(f-\frac{\ell}{T}\right)\frac{\left|\mathcal{Q}_C\left(f-\frac{\ell}{T}\right)\right|^2}{\mathcal{P}_{w_C}\left(f-\frac{\ell}{T}\right)} - \frac{1}{T}\mathcal{P}_a(f) = 0 \tag{8.534}$$

Considering that the function $\mathcal{G}(f)$ is periodic, it can be brought out of the equation; after some steps we get the optimum solution for $\mathcal{G}(f)$:

$$\mathcal{G}(f) = \frac{\mathcal{P}_a(f)}{T + \mathcal{P}_a(f)\frac{1}{T}\sum_{\ell=-\infty}^{+\infty} \frac{\left|\mathcal{Q}_C\left(f - \frac{\ell}{T}\right)\right|^2}{\mathcal{P}_{w_C}\left(f - \frac{\ell}{T}\right)}} \tag{8.535}$$

Hence, using (8.533),

$$\begin{aligned}\mathcal{H}(f) &= \mathcal{G}(f)\frac{|\mathcal{Q}_C(f)|^2}{\mathcal{P}_{w_C}(f)} \\ &= \frac{|\mathcal{Q}_C(f)|^2}{\mathcal{P}_{w_C}(f)} \frac{\mathcal{P}_a(f)}{T + \mathcal{P}_a(f)\frac{1}{T}\sum_{\ell=-\infty}^{+\infty} \frac{\left|\mathcal{Q}_C\left(f - \frac{\ell}{T}\right)\right|^2}{\mathcal{P}_{w_C}\left(f - \frac{\ell}{T}\right)}}\end{aligned} \tag{8.536}$$

Finally, using (8.517), the expression of the optimum receive filter is given by

$$\mathcal{G}_{Rc}(f) = \frac{\mathcal{Q}_C^*(f)}{\mathcal{P}_{w_C}(f)} e^{-j2\pi f t_0} \frac{\mathcal{P}_a(f)}{T + \mathcal{P}_a(f)\frac{1}{T}\sum_{\ell=-\infty}^{+\infty} \frac{\left|\mathcal{Q}_C\left(f - \frac{\ell}{T}\right)\right|^2}{\mathcal{P}_{w_C}\left(f - \frac{\ell}{T}\right)}} \tag{8.537}$$

Hence, (8.11) is proven.

8.A.3 Joint optimization of transmitter and receiver

Referring to Figure 8.55, we remove the assumption that $q_C(t) = h_{Tx} * g_C(t)$ is known and reformulate the problem in the following terms.

Given a transmission system with channel frequency response $\mathcal{G}_C(f)$, we wish to determine the frequency responses of the transmit filter $\mathcal{H}_{Tx}(f)$ (or equivalently $\mathcal{Q}_C(f) = \mathcal{H}_{Tx}(f)\mathcal{G}_C(f)$) and of the receive filter $\mathcal{G}_{Rc}(f)$ that minimize the functional J given by (8.512).

In this formulation we get a solution of interest only by setting a constraint on the statistical power of the transmitted signal,

$$\mathtt{M}_{Tx} = \int_{-\infty}^{+\infty} \left|\frac{1}{T}\mathcal{H}_{Tx}(f)\right|^2 \mathcal{P}_a(f)\, df = V \tag{8.538}$$

However, even for the particular case of i.i.d. input symbols, for which

$$\mathtt{M}_{Tx} = \frac{\sigma_a^2}{T}\int_{-\infty}^{+\infty} |\mathcal{H}_{Tx}(f)|^2\, df \tag{8.539}$$

the problem cannot be solved in closed form. Here we give the procedure to determine the solution [58, 59].

1. For each frequency $f \in [-\frac{1}{2T}, \frac{1}{2T}]$, determine the integer ℓ_{max} such that

$$\ell_{max}(f) = \arg\max_{\ell \in \mathcal{Z}} \frac{\left| \mathcal{G}_C \left(f - \frac{\ell}{T} \right) \right|^2}{\mathcal{P}_{w_C} \left(f - \frac{\ell}{T} \right)} \tag{8.540}$$

   Moreover, form the corresponding frequency set

$$\mathcal{B} = \left\{ f + \frac{\ell_{max}(f)}{T} : f \in \left[-\frac{1}{2T}, \frac{1}{2T} \right] \right\} \tag{8.541}$$

2. Choose a positive real number λ, then determine the subset of $\mathcal{B}$ defined as

$$\mathcal{B}_\lambda = \left\{ f \in \mathcal{B} : \frac{|\mathcal{G}_C(f)|^2}{\mathcal{P}_{w_C}(f)} > \frac{1}{\lambda} \right\} \tag{8.542}$$

3. Compute the magnitude of $\mathcal{H}_{Tx}(f)$:

$$|\mathcal{H}_{Tx}(f)|^2 = \begin{cases} \dfrac{T\sqrt{\lambda \mathcal{P}_{w_C}(f)}}{\sigma_a^2 |\mathcal{G}_C(f)|} - \dfrac{T\mathcal{P}_{w_C}(f)}{\sigma_a^2 |\mathcal{G}_C(f)|^2} & f \in \mathcal{B}_\lambda \\ 0 & f \notin \mathcal{B}_\lambda \end{cases} \tag{8.543}$$

   $|\mathcal{G}_{Rc}(f)|$ is determined using (8.537); the phase of $\mathcal{H}_{Tx}(f)$ and of $\mathcal{G}_{Rc}(f)$ must be such that $\mathcal{Q}_R(f) = \mathcal{H}_{Tx}(f)\mathcal{G}_C(f)\mathcal{G}_{Rc}(f)$ is a positive real-valued signal.

4. Substituting (8.543) in (8.539), compute $\mathtt{M}_{Tx}$. Verify whether $\mathtt{M}_{Tx} = V$, otherwise steps 2–4 must be repeated for a different value of λ until $\mathtt{M}_{Tx} = V$.

As observed in [58], the physical interpretation of (8.543) is that $\mathcal{H}_{Tx}(f)$ has non-zero components in relation to frequencies at which the channel does not attenuate too much with respect to the noise level. However, this ratio is defined by $1/\lambda$, where λ is a Lagrange multiplier used in the optimization of problem (8.512) under the constraint (8.539).

Appendix 8.B *DFE design: matrix formulations*

8.B.1 Method based on correlation sequences

We discuss a DFE design method that requires knowledge of the correlation sequences of the equalizer input signal and of the detected message, as well as their cross-correlation sequence [60].

With reference to Figure 8.13, we introduce the vectors whose elements are given by the coefficients of the feedforward and feedback filters,

$$\begin{aligned}\mathbf{c} &= [c_0, c_1, \ldots, c_{M_1-1}]^T \\ \mathbf{b} &= [b_1, b_2, \ldots, b_{M_2}]^T \\ \tilde{\mathbf{b}} &= [-1, \mathbf{b}^T]^T\end{aligned} \tag{8.544}$$

and the vectors

$$\begin{aligned}\mathbf{x}_k &= [x_k, x_{k-1}, \ldots, x_{k-M_1+1}]^T \\ \hat{\mathbf{a}}_k &= [\hat{a}_{k-D}, \ldots, \hat{a}_{k-D-M_2}]^T \\ \mathbf{a}_k &= [a_{k-D}, \ldots, a_{k-D-M_2}]^T \\ \hat{\mathbf{a}}'_k &= [\hat{a}_{k-D-1}, \ldots, \hat{a}_{k-D-M_2}]^T \\ \mathbf{a}'_k &= [a_{k-D-1}, \ldots, a_{k-D-M_2}]^T\end{aligned} \tag{8.545}$$

We note the presence of a delay D due to the feedforward filter. From (8.91), the sample at the decision point, y_k, can be expressed using the following vector notation:

$$y_k = \mathbf{c}^T \mathbf{x}_k + \mathbf{b}^T \hat{\mathbf{a}}'_k \tag{8.546}$$

Using (8.546), also the estimation error can be expressed in vector form as

$$\begin{aligned}e_k &= y_k - a_{k-D} \\ &= \mathbf{c}^T \mathbf{x}_k + \mathbf{b}^T \hat{\mathbf{a}}'_k - a_{k-D}\end{aligned} \tag{8.547}$$

Assuming that the detection of past symbols is correct, we have $\hat{\mathbf{a}}'_k = \mathbf{a}'_k$. Then, introducing the vector $\tilde{\mathbf{b}}$, (8.547) can be written as

$$\begin{aligned}e_k &= \mathbf{c}^T \mathbf{x}_k + [-1, \mathbf{b}^T]\mathbf{a}_k \\ &= \mathbf{c}^T \mathbf{x}_k + \tilde{\mathbf{b}}^T \mathbf{a}_k\end{aligned} \tag{8.548}$$

The filter coefficients $\mathbf{c}$ and $\tilde{\mathbf{b}}$ are determined by minimizing the cost function

$$J = E[|y_k - a_{k-D}|^2] \tag{8.549}$$

Introducing the correlation matrices

$$\begin{aligned}\mathbf{R}_{\mathbf{a}} &= E[\mathbf{a}_k^* \mathbf{a}_k^T] \\ \mathbf{R}_{\mathbf{x}} &= E[\mathbf{x}_k^* \mathbf{x}_k^T] \\ \mathbf{R}_{\mathbf{ax}} &= E[\mathbf{a}_k^* \mathbf{x}_k^T] \\ \mathbf{R}_{\mathbf{xa}} &= E[\mathbf{x}_k^* \mathbf{a}_k^T]\end{aligned} \tag{8.550}$$

the cost function J can be written as

$$\begin{aligned}J &= E[e_k^* e_k] \\ &= E[(\mathbf{c}^T \mathbf{x}_k + \tilde{\mathbf{b}}^T \mathbf{a}_k)^* (\mathbf{c}^T \mathbf{x}_k + \tilde{\mathbf{b}}^T \mathbf{a}_k)^T] \\ &= \mathbf{c}^H \mathbf{R}_{\mathbf{x}} \mathbf{c} + \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{a}} \tilde{\mathbf{b}} + \mathbf{c}^H \mathbf{R}_{\mathbf{xa}} \tilde{\mathbf{b}} + \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{ax}} \mathbf{c}\end{aligned} \tag{8.551}$$

On the other hand, minimization of the MSE implies orthogonality of the error e_k with respect to the input $\mathbf{x}_k$. Consequently, using the expression (8.548) for e_k it must be

$$E[e_k^* \mathbf{x}_k^T] = \mathbf{c}^H \mathbf{R}_{\mathbf{x}} + \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{ax}} = \mathbf{0} \tag{8.552}$$

or

$$\tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{ax}} = -\mathbf{c}^H \mathbf{R}_{\mathbf{x}} \tag{8.553}$$

Assuming $\mathbf{R}_{\mathbf{x}}$ is strictly positive definite and therefore invertible, we obtain

$$\mathbf{c}_{opt}^H = -\tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{ax}} \mathbf{R}_{\mathbf{x}}^{-1} \tag{8.554}$$

This relation, inserted in (8.551), yields

$$\begin{aligned}J &= -\tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{ax}} \mathbf{c} + \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{a}} \tilde{\mathbf{b}} - \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{ax}} \mathbf{R}_{\mathbf{x}}^{-1} \mathbf{R}_{\mathbf{xa}} \tilde{\mathbf{b}} + \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{ax}} \mathbf{c} \\ &= \tilde{\mathbf{b}}^H (\mathbf{R}_{\mathbf{a}} - \mathbf{R}_{\mathbf{ax}} \mathbf{R}_{\mathbf{x}}^{-1} \mathbf{R}_{\mathbf{xa}}) \tilde{\mathbf{b}}\end{aligned} \tag{8.555}$$

We recognize that

$$\mathbf{R}_{\mathbf{a}|\mathbf{x}} = \mathbf{R}_{\mathbf{a}} - \mathbf{R}_{\mathbf{ax}} \mathbf{R}_{\mathbf{x}}^{-1} \mathbf{R}_{\mathbf{xa}} \tag{8.556}$$

is the correlation matrix of the estimation error vector (see Appendix 2.A).

The problem thus reduces to finding the vector $\tilde{\mathbf{b}}$ that minimizes the quadratic form

$$J = \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{a}|\mathbf{x}} \tilde{\mathbf{b}} \tag{8.557}$$

Defining $\mathbf{e}_1 = [1, \mathbf{0}_{1 \times M_2}]^T$ and remembering that $\tilde{b}_1$, the first component of $\tilde{\mathbf{b}}$, must be equal to -1, the solution is obtained by the minimization of the quadratic function

$$J = \tilde{\mathbf{b}}^H \mathbf{R}_{\mathbf{a}|\mathbf{x}} \tilde{\mathbf{b}} \tag{8.558}$$

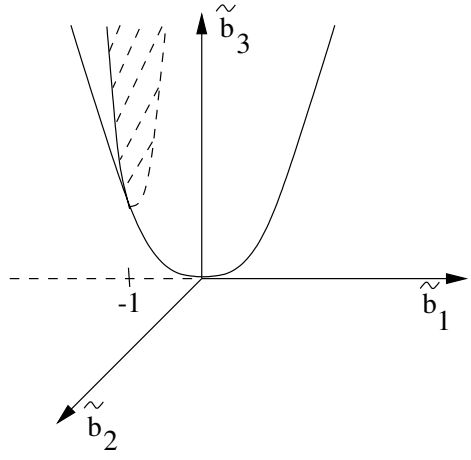


Figure 8.56. Constrained cost function in the case $M_2 = 2$.

with the constraint

$$\mathbf{v} = \mathbf{e}_1 \tilde{b}_1 + \mathbf{e}_1 = \mathbf{0} \tag{8.559}$$

Figure 8.56 illustrates the problem in the case $M_2 = 2$. The vector $\tilde{\mathbf{b}}$ that minimizes J subject to the constraint $\mathbf{v} = \mathbf{0}$ is obtained by the method of Lagrange multipliers,

$$\begin{cases} \nabla_{\tilde{\mathbf{b}}} J + \lambda \nabla_{\tilde{\mathbf{b}}} \mathbf{v} = \mathbf{0} \\ \mathbf{v} = \mathbf{0} \end{cases} \tag{8.560}$$

As $\nabla_{\tilde{\mathbf{b}}} J = 2\mathbf{R}_{\mathbf{a}|\mathbf{x}} \tilde{\mathbf{b}}$ and $\nabla_{\tilde{\mathbf{b}}} \mathbf{v} = \mathbf{e}_1$, (8.560) yields

$$\begin{cases} 2\mathbf{R}_{\mathbf{a}|\mathbf{x}} \tilde{\mathbf{b}} = -\lambda \mathbf{e}_1 \\ \mathbf{v} = \mathbf{0} \end{cases} \tag{8.561}$$

whose solution is given by

$$\tilde{\mathbf{b}}_{opt} = -\frac{1}{\lambda} (\mathbf{R}_{\mathbf{a}|\mathbf{x}})^{-1} \mathbf{e}_1 \tag{8.562}$$

where λ is the factor that forces the first component of $\tilde{\mathbf{b}}_{opt}$ equal to -1. We note that, apart from the coefficient $-1/\lambda$, $\tilde{\mathbf{b}}_{opt}$ coincides with the first column of the matrix $(\mathbf{R}_{\mathbf{a}|\mathbf{x}})^{-1}$. With this choice of $\tilde{\mathbf{b}}_{opt}$, (8.554) yields the optimum FF filter coefficients

$$\mathbf{c}_{opt} = -\mathbf{R}_{\mathbf{x}}^{-1} \mathbf{R}_{\mathbf{xa}} \tilde{\mathbf{b}}_{opt} \tag{8.563}$$

Although this procedure is very general, valid for a general statistic of the information message, it is rather computationally expensive because it requires two matrix inversions, for $\mathbf{R}_{\mathbf{a}|\mathbf{x}}$ and $\mathbf{R}_{\mathbf{x}}$.

Observation 8.15
For an LE, the formulation is as in (8.563), however, without the filter $\mathbf{b}$, and with $\hat{\mathbf{a}}_k = [\hat{a}_{k-D}]$ and $\tilde{\mathbf{b}} = [-1]$.

8.B.2 Method based on the channel impulse response and i.i.d. symbols

This method is obtained by applying a matrix formulation similar to that of Section 8.5. We introduce a definition that will simplify the notation.

Definition 8.4
Let $\mathbf{q} = [q_1, \ldots, q_N]^T$ be a vector with N elements, and

$$\mathbf{q}_i{}^j = [q_i, q_{i+1}, \ldots, q_j]^T \tag{8.564}$$

denote the vector containing a subsequence of consecutive elements of $\mathbf{q}$.

Let $\mathbf{A} = [A_{n,m}]$, $n = 1, \ldots, N_R$, $m = 1, \ldots, N_C$, be a $N_R \times N_C$ matrix. If $\mathbf{A}_{\bullet,m}$ denotes the m-th column, then

$$\mathbf{A}_{\bullet,i}{}^j = [\mathbf{A}_{\bullet,i}, \mathbf{A}_{\bullet,i+1}, \ldots, \mathbf{A}_{\bullet,j}] \tag{8.565}$$

denotes the matrix containing a subsequence of consecutive columns of the matrix $\mathbf{A}$.

Figure 8.57 illustrates the vector representation of the scheme of Figure 8.5, extended to include a DFE.

Let $\{h_i\}$, $i = -N_1, \ldots, N_2$, be the overall impulse response at the equalizer input. Introducing the vectors

$$\begin{aligned}
\mathbf{x}_k &= [x_k, x_{k-1}, \ldots, x_{k-M_1+1}]^T \\
\mathbf{a}_k &= [a_{k+N_1}, \ldots, a_k, \ldots, a_{k-N_2-M_1+1}]^T \\
\tilde{\mathbf{w}}_k &= [\tilde{w}_k, \tilde{w}_{k-1}, \ldots, \tilde{w}_{k-M_1+1}]^T
\end{aligned} \tag{8.566}$$

and using (8.544), it is possible to express the input vector $\mathbf{x}_k$ of the FF filter as a linear combination of the transmitted symbols $\mathbf{a}_k$ plus noise

$$\mathbf{x}_k = \mathbf{H}\mathbf{a}_k + \tilde{\mathbf{w}}_k \tag{8.567}$$

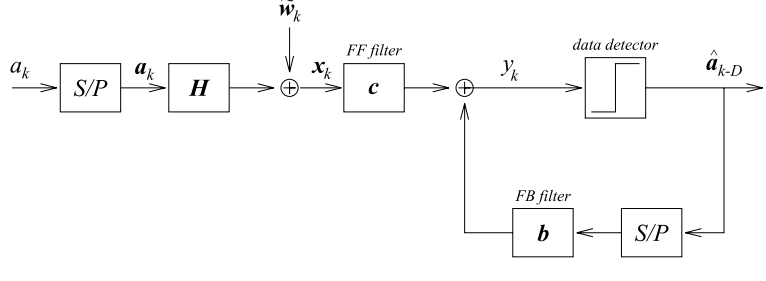


Figure 8.57. Overall system with DFE: vector formulation.

where

$$\mathbf{H} = \begin{bmatrix} h_{-N_1} & h_{-N_1+1} & h_{-N_1+2} & \dots & h_{N_2} & 0 & 0 & \dots & 0 \\ 0 & h_{-N_1} & h_{-N_1+1} & \dots & h_{N_2-1} & h_{N_2} & 0 & \dots & 0 \\ 0 & 0 & h_{-N_1} & \dots & h_{N_2-2} & h_{N_2-1} & h_{N_2} & \dots & 0 \\ \vdots & & & \ddots & & & & \ddots & \vdots \\ 0 & 0 & 0 & \dots & h_{-N_1} & h_{-N_1+1} & & & h_{N_2} \end{bmatrix} \tag{8.568}$$

is an $M_1 \times (N_1 + N_2 + M_1)$ Toeplitz matrix .

We note that in (8.566) the definition of $\mathbf{a}_k$ refers to the system memory at the output of the filter $\mathbf{c}$.

It is useful to express the operation of the FB filter assuming as an input the vector $\mathbf{a}_k$, given by (8.566). Therefore we introduce the vector

$$\mathbf{b}' = [\mathbf{0}_{1\times(N_1+D+1)}, \mathbf{b}^T, \mathbf{0}_{1\times M_3}]^T \tag{8.569}$$

where $M_3 = M_1 + N_2 - D - 1 - M_2$.

The sample y_k at the decision point can thus be expressed as

$$y_k = \mathbf{c}^T\mathbf{x}_k + \mathbf{b}'^T\mathbf{a}_k = \mathbf{c}^T\mathbf{H}\mathbf{a}_k + \mathbf{b}'^T\mathbf{a}_k + \mathbf{c}^T\tilde{\mathbf{w}}_k = (\mathbf{c}^T\mathbf{H} + \mathbf{b}'^T)\mathbf{a}_k + \mathbf{c}^T\tilde{\mathbf{w}}_k \tag{8.570}$$

Equation (8.570) suggests that the cancellation of ISI from the sample y_k, due to the symbols $\{a_{k-D-i}\}$, $i = 1, \dots, M_2$, is obtained by setting

$$\mathbf{b}' = [\mathbf{0}_{1\times(N_1+D+1)}, -(\mathbf{c}^T\mathbf{H})_{N_1+D+2}^{N_1+D+M_2+1}, \mathbf{0}_{1\times M_3}]^T \tag{8.571}$$

Fixed $\mathbf{b}'$, (8.570) becomes

$$y_k = \mathbf{c}^T(\mathbf{H}'\mathbf{a}_k + \tilde{\mathbf{w}}_k) \tag{8.572}$$

where

$$\begin{aligned} \mathbf{H}' &= [\mathbf{H}_{\bullet,1}^{N_1+D+1}, \mathbf{0}_{M_1\times M_2}, \mathbf{H}_{\bullet,N_1+D+M_2+2}^{N_1+N_2+M_1}] \\ &= \begin{bmatrix} h_{-N_1} & h_{-N_1+1} & \dots & h_D & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & h_{-N_1} & \dots & h_{D-1} & 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & & \ddots & \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 & h_{N_2-M_3+1} & \dots & h_{N_2} \end{bmatrix} \end{aligned} \tag{8.573}$$

We are now back to the case of one filter $\mathbf{c}$, having as input vector $(\mathbf{H}'\mathbf{a}_k + \tilde{\mathbf{w}}_k)$, whose coefficients are determined by minimizing the cost function

$$J = E[|y_k - a_{k-D}|^2] \tag{8.574}$$

From the Wiener theory, we know that to find the solution we must compute the autocorrelation matrix $\mathbf{R}$ of the input signal and the cross-correlation vector $\mathbf{p}$ between the desired output signal and the input signal. From

$$\mathbf{R} = E[(\mathbf{H}'\mathbf{a}_k + \tilde{\mathbf{w}}_k)^*(\mathbf{H}'\mathbf{a}_k + \tilde{\mathbf{w}}_k)^T] \tag{8.575}$$

assuming the symbols of the sequence $\{a_k\}$ are i.i.d. with variance σ_a^2, we obtain

$$\mathbf{R} = \sigma_a^2 \mathbf{H}'^* \mathbf{H}'^T + \mathbf{R}_{\tilde{\mathbf{w}}} \tag{8.576}$$

where $\mathbf{R}_{\tilde{\mathbf{w}}}$ is the autocorrelation matrix of the noise, assumed known.

Moreover, we have

$$\mathbf{p} = E[a_{k-D}(\mathbf{H}'\mathbf{a}_k + \tilde{\mathbf{w}}_k)^*] = \mathbf{H}'^* E[a_{k-D}\mathbf{a}_k^*] \tag{8.577}$$

given the statistical independence between transmitted symbols and noise. As the symbols are i.i.d., the vector $E[a_{k-D}\mathbf{a}_k^*]$ has all zero elements except that in position $N_1 + D + 1$, with value σ_a^2, and $\mathbf{p}$ corresponds to the (N_1+D+1)-th column of the matrix $\mathbf{H}'^*$ multiplied by the scalar σ_a^2,

$$\mathbf{p} = \sigma_a^2 \mathbf{H}'^*_{\bullet, N_1+D+1} = \sigma_a^2 [h_D, h_{D-1}, \ldots, h_{-N_1}, \mathbf{0}_{1\times M_1-(N_1+D+1)}]^H \tag{8.578}$$

According to the Wiener theory, the optimum FF filter coefficients are given by

$$\begin{aligned} \mathbf{c}_{opt} &= \mathbf{R}^{-1}\mathbf{p} \\ &= (\sigma_a^2 \mathbf{H}'^* \mathbf{H}'^T + \mathbf{R}_{\tilde{\mathbf{w}}})^{-1} \sigma_a^2 \mathbf{H}'^*_{\bullet, N_1+D+1} \end{aligned} \tag{8.579}$$

and the minimum value of the cost function J is

$$J_{min} = \sigma_a^2 - \mathbf{p}^H \mathbf{c}_{opt} \tag{8.580}$$

The feedback filter is obtained by substituting (8.579) in (8.571).

Observation 8.16

For an LE, the formulation is as in (8.579) and (8.580), however, without the filter $\mathbf{b}$, and with $\mathbf{H}'$ equal to $\mathbf{H}$.

8.B.3 Method based on the channel impulse response and any symbol statistic

For the signal (8.567), the correlation matrices (8.550) are given by

$$\begin{aligned} \mathbf{R}_{\mathbf{a}} &= E[\mathbf{a}_k^* \mathbf{a}_k^T] \\ \mathbf{R}_{\mathbf{x}} &= E[\mathbf{x}_k^* \mathbf{x}_k^T] = \mathbf{H}^* \mathbf{R}_{\mathbf{a}} \mathbf{H}^T + \mathbf{R}_{\tilde{\mathbf{w}}} \\ \mathbf{R}_{\mathbf{ax}} &= E[\mathbf{a}_k^* \mathbf{x}_k^T] = \mathbf{R}_{\mathbf{a}} \mathbf{H}^T \\ \mathbf{R}_{\mathbf{xa}} &= E[\mathbf{x}_k^* \mathbf{a}_k^T] = \mathbf{H}^* \mathbf{R}_{\mathbf{a}} \end{aligned} \tag{8.581}$$

Using (8.581) and (8.569) in (8.555), from (3.176) we obtain

$$\begin{aligned} J &= \tilde{\mathbf{b}}^H[\mathbf{R_a} - \mathbf{R_{ax}}\mathbf{R_x}^{-1}\mathbf{R_{xa}}]\tilde{\mathbf{b}} \\ &= \tilde{\mathbf{b}}^H[\mathbf{R_a}^{-1} + \mathbf{H}^T\mathbf{R}_{\tilde{\mathbf{w}}}^{-1}\mathbf{H}^*]^{-1}\tilde{\mathbf{b}} \end{aligned} \tag{8.582}$$

Setting

$$\mathbf{R}_D = [\mathbf{0}_{(M_2+1)\times(N_1+D)}, \mathbf{I}_{M_2+1}, \mathbf{0}_{(M_2+1)\times M_3}][\mathbf{R_a}^{-1} + \mathbf{H}^T\mathbf{R}_{\tilde{\mathbf{w}}}^{-1}\mathbf{H}^*]^{-1} \begin{bmatrix} \mathbf{0}_{(M_2+1)\times(N_1+D)} \\ \mathbf{I}_{M_2+1} \\ \mathbf{0}_{(M_2+1)\times M_3} \end{bmatrix} \tag{8.583}$$

(8.582) becomes

$$J = [-1, \mathbf{b}^H]\mathbf{R}_D \begin{bmatrix} -1 \\ \mathbf{b} \end{bmatrix} \tag{8.584}$$

This quadratic form is minimized by choosing as FB filter coefficients (see (8.562))

$$[-1, b_{opt,1}, \ldots, b_{opt,M_2}]^T = -\frac{\mathbf{R}_D^{-1}\mathbf{e}_1}{\mathbf{e}_1^T\mathbf{R}_D^{-1}\mathbf{e}_1} \tag{8.585}$$

With this choice for the FB filter, the minimum value of the MSE is given by

$$J_{min} = \frac{1}{\mathbf{e}_1^T\mathbf{R}_D^{-1}\mathbf{e}_1} \tag{8.586}$$

and the optimum FF filter can be obtained by (8.563).

Observation 8.17
We note that if the number of coefficients M_2 of the FB filter is chosen equal to the length of the overall channel impulse response at the decision point, that is if $M_2 = N_1 + N_2 + 1$, the performance is improved; moreover, it is possible to reduce the computational complexity by exploiting the special structure of the matrices [60].

8.B.4 FS-DFE

Although the FB filter necessarily operates with input samples having sampling period equal to T, the FF filter can be fractionally spaced. According to the Wiener theory, the optimum coefficients of $\mathbf{c}$ and $\mathbf{b}$ are obtained by one of the two methods presented in the previous sections; however, care must be taken in accounting for the fact that the FF filter operates with input samples having sampling period equal to T/F_0, where F_0 is the oversampling factor.

Therefore we introduce the vector notation

$$\underline{\mathbf{h}}_i = [h_{iF_0+(F_0-1)}, \ldots, h_{iF_0+1}, h_{iF_0}]^T \qquad i = -N_1, \ldots, N_2 \tag{8.587}$$

$$\underline{\tilde{\mathbf{w}}}_k = [\tilde{w}_{kF_0+(F_0-1)}, \ldots, \tilde{w}_{kF_0+1}, \tilde{w}_{kF_0}]^T \tag{8.588}$$

$$\underline{\mathbf{x}}_k = [x_{kF_0+(F_0-1)}, \ldots, x_{kF_0}]^T = \sum_{i=-N_1}^{N_2} \underline{\mathbf{h}}_i a_{k-i} + \underline{\tilde{\mathbf{w}}}_k \tag{8.589}$$

Now the FF filter $\mathbf{c}$ has a number of coefficients equal to $M_1 F_0$,

$$\mathbf{c}^T = [\underline{\mathbf{c}}_1^T, \ldots, \underline{\mathbf{c}}_{M_1}^T] \tag{8.590}$$

where

$$\underline{\mathbf{c}}_i^T = [c_{0,i}, c_{1,i}, \ldots, c_{(F_0-1),i}] \tag{8.591}$$

The input signal to the filter $\mathbf{c}$ is still denoted by the vector $\mathbf{x}_k$, whose components are the vectors $\{\underline{\mathbf{x}}_i\}$, $i = k, \ldots, k - M_1 - 1$, each with dimension F_0. Then we can write

$$\mathbf{x}_k = \begin{bmatrix} \underline{\mathbf{x}}_k \\ \vdots \\ \underline{\mathbf{x}}_{k-M_1-1} \end{bmatrix} = \mathbf{H}\mathbf{a}_k + \tilde{\mathbf{w}}_k \tag{8.592}$$

where $\tilde{\mathbf{w}}_k$ follows the same structure of $\mathbf{x}_k$, and

$$\mathbf{H} = \begin{bmatrix} \underline{\mathbf{h}}_{-N_1} & \underline{\mathbf{h}}_{-N_1+1} & \underline{\mathbf{h}}_{-N_1+2} & \cdots & \underline{\mathbf{h}}_{N_2} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \underline{\mathbf{h}}_{-N_1} & \underline{\mathbf{h}}_{-N_1+1} & \cdots & \underline{\mathbf{h}}_{N_2-1} & \underline{\mathbf{h}}_{N_2} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \underline{\mathbf{h}}_{-N_1} & & \underline{\mathbf{h}}_{N_2-2} & \underline{\mathbf{h}}_{N_2-1} & \underline{\mathbf{h}}_{N_2} & & \mathbf{0} \\ \vdots & & & \ddots & & & & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \underline{\mathbf{h}}_{-N_1} & \underline{\mathbf{h}}_{-N_1+1} & & & \underline{\mathbf{h}}_{N_2} \end{bmatrix} \tag{8.593}$$

Note that the above relation is formally identical to (8.567). Therefore, thanks to this property, we can obtain the optimum coefficients of the FF and FB filters by the methods introduced in the previous sections.

Appendix 8.C *Equalization based on the peak value of ISI*

The equalization algorithm discussed in this appendix is related to the eye diagram at the decision point [4]. For an equalizer with N coefficients, let ψ_n be the overall impulse response at the decision point:

$$\psi_n = \sum_{j=0}^{N-1} c_j h_{n-j} \qquad n = -N_1, \ldots, N_2 + N - 1 \tag{8.594}$$

We form the lead version of ψ_n

$$\eta_n = \psi_{n+D} \tag{8.595}$$

such that η_0 is the desired sample at the decision point. Typically,

$$D = \frac{N-1}{2} \tag{8.596}$$

To simplify the notation, we assume that η_0 coincides with the peak

$$\eta_0 = \max_n |\eta_n| \tag{8.597}$$

Set

$$L_1 = N_1 + D \qquad L_2 = N_2 + N - 1 - D \tag{8.598}$$

$$L = L_2 + L_1 + 1 \tag{8.599}$$

the cost function J to be minimized considers the peak value of ISI, that is

$$J = \frac{1}{\eta_0} \sum_{n=-L_1,\, n\neq 0}^{L_2} |\eta_n| \tag{8.600}$$

We note that $J = 0$ for

$$\eta_n = \begin{cases} \eta_0 \neq 0 & n = 0 \\ 0 & n \neq 0 \end{cases} \tag{8.601}$$

and the system is equalized, hence $\mathbf{c}$ is a zero forcing filter.

On the other hand, if $J > 1$, then the eye is completely shut. In any event, the effect of the noise is not considered.

As illustrated in Figure 8.58, the equalizer coefficients are determined iteratively in relation to a repetitive input sequence, with repetition index i, composed of a symbol equal to one followed by $L - 1$ zeros: $\{a_k,\ k = 0, \ldots, L-1\} = \{1, 0, \ldots, 0\}$.

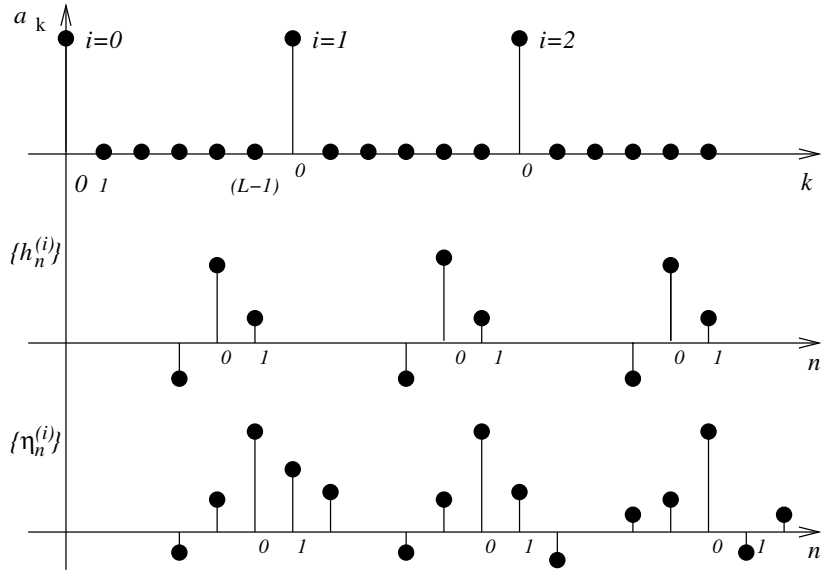


Figure 8.58. Illustration of the zero forcing iterative equalization method.

If we denote by $\{\eta_n^{(i)}\}$ the impulse response at the i-th iteration, the law of coefficient update by the gradient method is given by

$$c_j^{(i+1)} = c_j^{(i)} - \mu \operatorname{sgn} \eta_{j-D}^{(i)} \qquad j = 0, \ldots, N-1 \quad j \neq D \tag{8.602}$$

$$c_D^{(i+1)} = 1 - \sum_{j=0,\, j\neq D}^{N-1} c_j^{(i+1)} h_{D-j} \tag{8.603}$$

The last equation is obtained from the constraint $\eta_0 = 1$.

It can be proved that if the eye is open at the equalizer input, that is

$$\frac{1}{h_0} \sum_{n=-N_1,\, n\neq 0}^{N_2} |h_n| < 1 \tag{8.604}$$

then (8.602) converges and (8.601) is verified.

It is useful to observe that in the absence of noise it would be sufficient to send only one sequence $\{a_k\}$, $k = 0, \ldots, L-1$: in practice, the various estimates of $\{h_n\}$ are used for averaging with respect to noise.

We note the simplicity of implementing this algorithm: indeed, it was one of the first algorithms to equalize transmission lines. Now its use is justified only in systems with very large modulation rates, for example, optical fiber systems. In any case, for the algorithm to converge, the eye needs to be already opened, a constraint that did not exist in the MSE criterion.

Another distinct characteristic of this method derives from the fact that it uses test sequences $\{a_k\}$, different from those typically used for transmission.

Appendix 8.D *Description of a finite state machine (FSM)*

We consider a discrete-time system with input sequence $\{i_k\}$ and output sequence $\{o_k\}$, evaluated at instants kT. We say that the output sequence is generated by an FSM if there are a sequence of states $\{\mathbf{s}_k\}$ and two functions f_o and f_s, such that

$$o_k = f_o(i_k, \mathbf{s}_{k-1}) \tag{8.605}$$

$$\mathbf{s}_k = f_s(i_k, \mathbf{s}_{k-1}) \tag{8.606}$$

as illustrated in Figure 8.59. The first equation describes the fact that the output sample depends on the current input and the state of the system. The second equation represents the memory part of the FSM and describes the state evolution.

We note that if f_s is a one-to-one function of i_k, that is if every transition between two states is determined by a unique input value, then (8.605) can be written as

$$o_k = f(\mathbf{s}_k, \mathbf{s}_{k-1}) \tag{8.607}$$

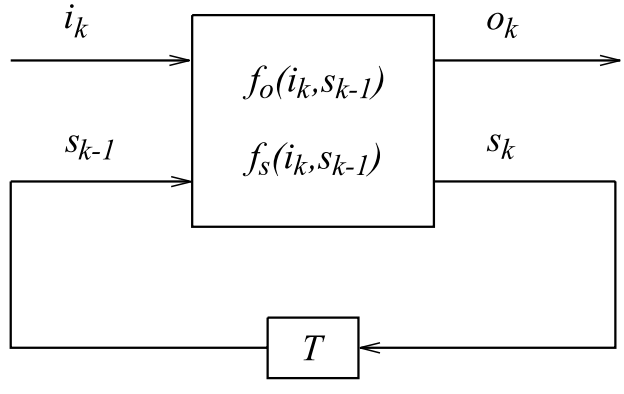


Figure 8.59. General block diagram of a finite state sequential machine.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 9

Orthogonal frequency division multiplexing

For channels that exhibit high signal attenuation at frequencies within the passband, a valid alternative to CAP/QAM is represented by a modulation technique based on filter banks, known as *orthogonal frequency division multiplexing* (OFDM), or *multicarrier modulation*. As the term implies, multicarrier modulation is obtained in principle by modulating several carriers in parallel using blocks of symbols, therefore using a symbol period that is typically much longer than the symbol period of a single-carrier system transmitting at the same bit rate. The resulting narrowband signals around the frequencies of the carriers are then added and transmitted over the channel. The narrowband signals are usually referred to as subchannel signals.

An advantage of OFDM with respect to single-carrier systems is represented by the lower complexity required for equalization, that under certain conditions can be performed by a filter with a single coefficient per subchannel. A long symbol period also yields a greater immunity of an OFDM system to impulse noise; however the symbol duration, and hence the number of subchannels, are limited for transmission over time-variant channels. As we will see in this chapter, another important aspect is represented by the efficient implementation of modulator and demodulator, obtained by sophisticated signal processing algorithms.[1]

9.1 OFDM systems

Orthogonal frequency division multiplexing is an efficient modulation technique by which blocks of $\mathcal{M}$ symbols are transmitted in parallel over $\mathcal{M}$ *subchannels*, using $\mathcal{M}$ modulation filters [1] with frequency responses $\mathcal{H}_i(f)$, $i = 0, \ldots, \mathcal{M}-1$. We consider the baseband equivalent OFDM system illustrated in Figure 9.1. The $\mathcal{M}$ input symbols at the k-th modulation interval are represented by the vector

$$\mathbf{a}_k = [a_k[0], \ldots, a_k[\mathcal{M}-1]]^T \tag{9.1}$$

[1] In this text, we refer to OFDM as a general technique for multicarrier transmission. In particular, this applies to both wired and wireless systems. When multicarrier transmission is achieved without filtering of the individual subchannel signals, other authors use the term discrete multitone (DMT) modulation for wired transmission systems, whereas they reserve the term OFDM for wireless systems.

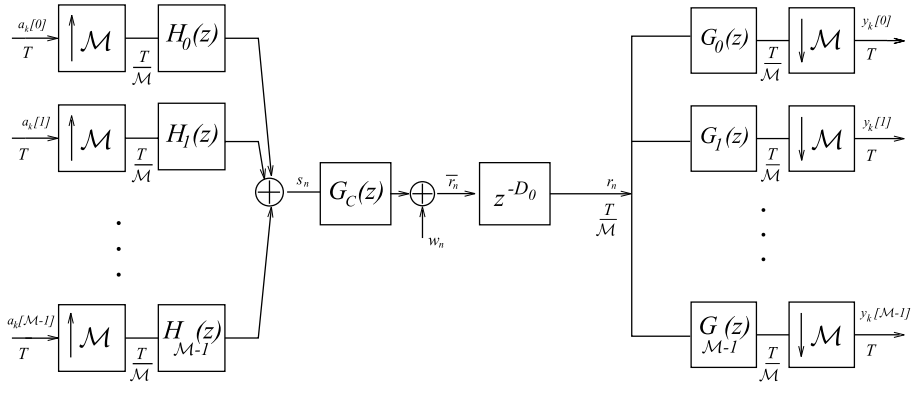


Figure 9.1. Block diagram of an OFDM system.

where $a_k[i] \in \mathcal{A}[i]$, $i = 0, \ldots, \mathcal{M}-1$. The alphabets $\mathcal{A}[i]$, $i = 0, \ldots, \mathcal{M}-1$, correspond to two dimensional constellations, which are not necessarily identical.

The symbol rate of each subchannel is equal to $1/T$ and corresponds to the rate of the input symbol vectors $\{\mathbf{a}_k\}$: $1/T$ is also called *modulation rate*. The sampling period of each input sequence is changed from T to $T/\mathcal{M}$ by upsampling, that is by inserting $\mathcal{M}-1$ zeros between consecutive symbols (see Appendix 1.A); each sequence thus obtained is then filtered by a bandlimited filter properly allocated in frequency. The filter on the i-th branch is characterized by the impulse response $\{h_n[i]\}$, with transfer function

$$H_i(z) = \sum_{n=-\infty}^{+\infty} h_n[i]\, z^{-n} \tag{9.2}$$

and frequency response $\mathcal{H}_i(f) = H_i(z)|_{z=e^{j2\pi fT/\mathcal{M}}}$, $i = 0, \ldots, \mathcal{M}-1$. The transmit filters, also called interpolator filters, work in parallel at the rate $\mathcal{M}/T$, called *transmission rate*. The transmitter output signal s_n is given by the sum of the filter output signals, that is

$$s_n = \sum_{i=0}^{\mathcal{M}-1} \sum_{k=-\infty}^{+\infty} a_k[i]\, h_{n-k\mathcal{M}}[i] \tag{9.3}$$

The received signal $\bar{r}_n$, suitably delayed of D_0 samples, is filtered by $\mathcal{M}$ filters in parallel at the transmission rate. The receive filters have impulse responses $\{g_n[i]\}$, $i = 0, \ldots, \mathcal{M}-1$, and transfer functions given by $G_i(z) = \sum_n g_n[i] z^{-n}$, $i = 0, \ldots, \mathcal{M}-1$. By downsampling the $\mathcal{M}$ output signals of the receive filters at the modulation rate $1/T$, we obtain the vector sequence $\{\mathbf{y}_k = [y_k[0], \ldots, y_k[\mathcal{M}-1]]^T\}$, which is used to detect the vector sequence $\{\mathbf{a}_k\}$; receive filters are also known as decimator filters. When the channel is ideal and noiseless, we get $\bar{r}_n = s_n$. The delay D_0 is chosen to obtain proper synchronization of the receive filter bank.

We consider as transmit filters FIR causal filters having length $\gamma\mathcal{M}$, and support $\{0, 1, \ldots, \gamma\mathcal{M}-1\}$, that is $h_n[i]$ may be non-zero for $n = 0, \ldots, \gamma\mathcal{M}-1$. We assume matched receive filters; as we are dealing with causal filters, we consider the expression of

the matched filters $\{g_n[i]\}$, for $i = 0, \ldots, \mathcal{M}-1$, given by

$$g_n[i] = h^*_{\gamma\mathcal{M}-n}[i] \qquad \forall n \tag{9.4}$$

We observe that the support of $\{g_n[i]\}$ is $\{1, 2, \ldots, \gamma\mathcal{M}\}$, and, for an ideal channel, $D_0 = 0$.

9.2 Orthogonality conditions

Assuming the channel is ideal and noiseless, if the transmit and receive filter banks are designed such that certain orthogonality conditions are satisfied, the subchannel output signals are delayed versions of the transmitted symbol sequences at the corresponding subchannel inputs. The orthogonality conditions, also called perfect reconstruction conditions, can be interpreted as a more general form of the Nyquist criterion.

Time domain

With reference to the general scheme of Figure 9.1, at the output of the j-th receive subchannel, before downsampling, the impulse response relative to the i-th input is given by

$$\sum_{p=0}^{\gamma\mathcal{M}-1} h_p[i]\, g_{n-p}[j] = \sum_{p=0}^{\gamma\mathcal{M}-1} h_p[i]\, h^*_{\gamma\mathcal{M}+p-n}[j] \qquad \forall n \tag{9.5}$$

We note that, for $j = i$, the peak of the sequence given by (9.5) is obtained for $n = \gamma\mathcal{M}$. Observing (9.5), transmission in the absence of intersymbol interference over a subchannel, as well as absence of interchannel interference (ICI) between subchannels, is achieved if orthogonality conditions are satisfied, that in the time domain are expressed as

$$\sum_{p=0}^{\gamma\mathcal{M}-1} h_p[i]\, h^*_{p+\mathcal{M}(\gamma-k)}[j] = \delta_{i-j}\, \delta_{k-\gamma} \qquad i, j = 0, \ldots, \mathcal{M}-1 \tag{9.6}$$

Hence, in the ideal channel case considered here, the vector sequence at the output of the decimator filter bank is a replica of the transmitted vector sequence with a delay of γ modulation intervals, that is $\{\mathbf{y}_k\} = \{\mathbf{a}_{k-\gamma}\}$. Sometimes the elements of a set of orthogonal impulse responses that satisfy (9.6) are called *wavelets*.

Frequency domain

In the frequency domain, the conditions (9.6) are expressed as

$$\sum_{\ell=0}^{\mathcal{M}-1} \mathcal{H}_i\left(f - \ell\,\frac{\mathcal{M}}{T}\right)\mathcal{H}^*_j\left(f - \ell\,\frac{\mathcal{M}}{T}\right) = \delta_{i-j} \qquad i, j = 0, \ldots, \mathcal{M}-1 \tag{9.7}$$

z-transform domain

In the ideal channel case, the relation between the inputs $a_k[i]$, $i = 0, \ldots, \mathcal{M}-1$, and the j-th output $y_k[j]$ is represented by the block diagram of Figure 9.2. We note that every

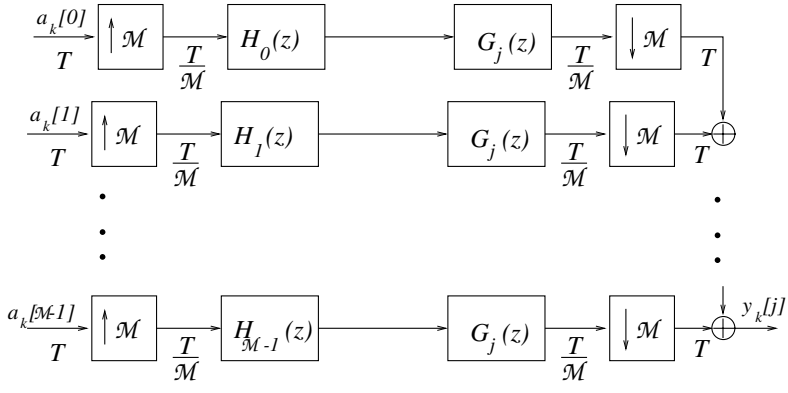


Figure 9.2. Block diagram that illustrates the relation between the $\mathcal{M}$ inputs and the j-th output.

branch in the diagram represents a time-invariant linear system. Setting

$$\mathbf{a}(z) = \left[\sum_k a_k[0] z^{-k}, \ldots, \sum_k a_k[\mathcal{M}-1] z^{-k} \right]^T \tag{9.8}$$

and

$$\mathbf{y}(z) = \left[\sum_k y_k[0] z^{-k}, \ldots, \sum_k y_k[\mathcal{M}-1] z^{-k} \right]^T \tag{9.9}$$

in general the input–output relation is given by (see Figure 9.6)

$$\mathbf{y}(z) = \mathbf{S}(z)\, \mathbf{a}(z) \tag{9.10}$$

where the element $[\mathbf{S}(z)]_{p,q}$ of the matrix $\mathbf{S}(z)$ is the transfer function between the q-th input and the p-th output. We note that the orthogonality conditions, expressed in the time domain by (9.6), are satisfied if

$$\mathbf{S}(z) = z^{-\gamma}\, \mathbf{I} \tag{9.11}$$

where $\mathbf{I}$ is the $\mathcal{M} \times \mathcal{M}$ identity matrix.

9.3 Efficient implementation of OFDM systems

For large values of $\mathcal{M}$, a direct implementation of an OFDM system as shown in Figure 9.1 would require an exceedingly large computational complexity, as all filtering operations are performed at a high rate equal to $\mathcal{M}/T$. A reduction in the number of operations per unit of time is obtained by resorting to the polyphase representation of the various filters [2].

OFDM implementation employing matched filters

Using the polyphase representation of both transmit and receive filter banks introduced in Appendix 1.A, we have (see (1.635))

$$H_i(z) = \sum_{\ell=0}^{\mathcal{M}-1} z^{-\ell}\, E_i^{(\ell)}(z^{\mathcal{M}}) \qquad i = 0, \ldots, \mathcal{M}-1 \tag{9.12}$$

where $E_i^{(\ell)}(z)$ is the transfer function of ℓ-th polyphase component of $H_i(z)$. We define the vector of the transfer functions of transmit filters as

$$\mathbf{h}(z) = [H_0(z), \ldots, H_{\mathcal{M}-1}(z)]^T \tag{9.13}$$

Let $\mathbf{e}(z)$ be the vector of delay elements given by

$$\mathbf{e}(z) = [z^{-(\mathcal{M}-1)}, z^{-(\mathcal{M}-2)}, \ldots, 1]^T \tag{9.14}$$

and $\mathbf{E}(z)$ the matrix of the transfer functions of the polyphase components of the transmit filters, given by

$$\mathbf{E}(z) = \begin{bmatrix} E_0^{(0)}(z) & E_1^{(0)}(z) & \cdots & E_{\mathcal{M}-1}^{(0)}(z) \\ E_0^{(1)}(z) & E_1^{(1)}(z) & \cdots & E_{\mathcal{M}-1}^{(1)}(z) \\ \vdots & \vdots & & \vdots \\ E_0^{(\mathcal{M}-1)}(z) & E_1^{(\mathcal{M}-1)}(z) & \cdots & E_{\mathcal{M}-1}^{(\mathcal{M}-1)}(z) \end{bmatrix} \tag{9.15}$$

From the identity

$$z^{-(\mathcal{M}-1)}\, \mathbf{e}^H\left(\frac{1}{z^*}\right) = [1, z^{-1}, \ldots, z^{-(\mathcal{M}-1)}] \tag{9.16}$$

$\mathbf{h}(z)$ can be expressed as

$$\mathbf{h}^T(z) = z^{-(\mathcal{M}-1)}\, \mathbf{e}^H\left(\frac{1}{z^*}\right) \mathbf{E}(z^{\mathcal{M}}) \tag{9.17}$$

Observing (9.17), the $\mathcal{M}$ interpolated input sequences are filtered by filters whose transfer functions are represented in terms of the polyphase components, expressed as columns of the matrix $\mathbf{E}(z^{\mathcal{M}})$. Because the transmitted signal $\{s_n\}$ is given by the sum of the filter outputs, the operations of the transmit filter bank are illustrated by the scheme of Figure 9.3, where the signal $\{s_n\}$ is obtained from a delay line that collects the outputs of the $\mathcal{M}$ vectors of filters with transfer functions given by the rows of $\mathbf{E}(z^{\mathcal{M}})$, and input given by the vector of interpolated input symbols.

At the receiver, using a representation equivalent to (1.637), that is obtained by a permutation of the polyphase components, we get

$$G_i(z) = \sum_{\ell=0}^{\mathcal{M}-1} z^{-(\mathcal{M}-1-\ell)}\, R_i^{(\ell)}(z^{\mathcal{M}}) \qquad i = 0, \ldots, \mathcal{M}-1 \tag{9.18}$$

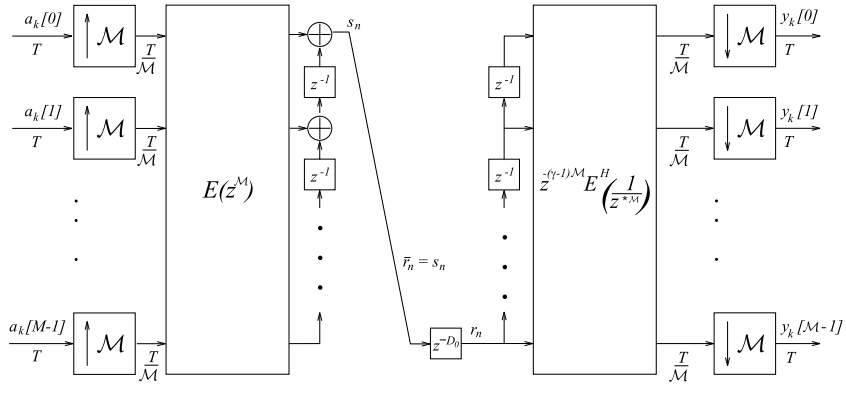


Figure 9.3. Implementation of an OFDM system using the polyphase representation.

where $R_i^{(\ell)}(z)$ is the transfer function of the $(\mathcal{M}-1-\ell)$-th polyphase component of $G_i(z)$. Then the vector of the transfer functions of the receive filters $\mathbf{g}(z) = [G_0(z), \ldots, G_{\mathcal{M}-1}(z)]^T$ can be expressed as

$$\mathbf{g}(z) = \mathbf{R}(z^{\mathcal{M}})\,\mathbf{e}(z) \tag{9.19}$$

where $\mathbf{R}(z)$ is the matrix of the transfer functions of the polyphase components of the receive filters, and is given by

$$\mathbf{R}(z) = \begin{bmatrix} R_0^{(0)}(z) & R_0^{(1)}(z) & \cdots & R_0^{(\mathcal{M}-1)}(z) \\ R_1^{(0)}(z) & R_1^{(1)}(z) & \cdots & R_1^{(\mathcal{M}-1)}(z) \\ \vdots & \vdots & & \vdots \\ R_{\mathcal{M}-1}^{(0)}(z) & R_{\mathcal{M}-1}^{(1)}(z) & \cdots & R_{\mathcal{M}-1}^{(\mathcal{M}-1)}(z) \end{bmatrix} \tag{9.20}$$

Observing (9.19), the $\mathcal{M}$ signals at the output of the receive filter bank before downsampling are obtained by filtering in parallel the received signal by filters whose transfer functions are represented in terms of the polyphase components, expressed as rows of the matrix $\mathbf{R}(z^{\mathcal{M}})$. Therefore the $\mathcal{M}$ output signals are equivalently obtained by filtering the vector of received signal samples $[\bar{r}_{n-(\mathcal{M}-1)}, \bar{r}_{n-\mathcal{M}+2}, \ldots, \bar{r}_n]^T$ by the $\mathcal{M}$ vectors of filters with transfer functions given by the rows of the matrix $\mathbf{R}(z^{\mathcal{M}})$.

In particular, recalling that the receive filters have impulse responses given by (9.4), we obtain

$$G_i(z) = z^{-\gamma\mathcal{M}}\, H_i^*\left(\frac{1}{z^*}\right) \qquad 0 \le i \le \mathcal{M}-1 \tag{9.21}$$

Substituting (9.21) in (9.19), and using (9.17), the expression of the vector $\mathbf{g}(z)$ of the transfer functions of the receive filters becomes

$$\mathbf{g}(z) = z^{-\gamma\mathcal{M}}\,\mathbf{h}^H\left(\frac{1}{z^*}\right) = z^{-\gamma\mathcal{M}+\mathcal{M}-1}\,\mathbf{E}^H\left(\frac{1}{z^{*\mathcal{M}}}\right)\mathbf{e}(z) \tag{9.22}$$

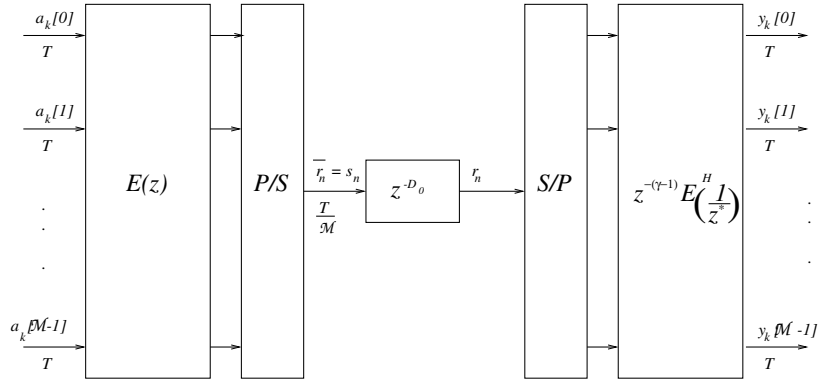


Figure 9.4. Equivalent OFDM system implementation.

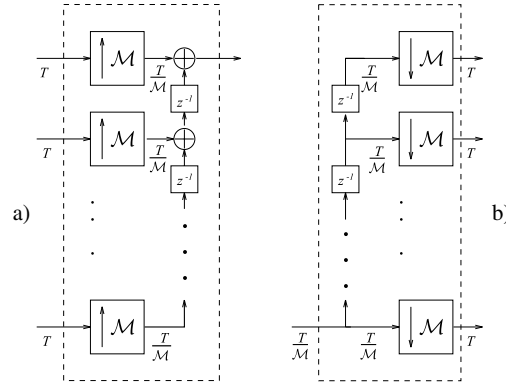


Figure 9.5. Block diagrams of (a) parallel to serial converter, (b) serial to parallel converter.

Apart from a delay term z^{-1}, the operations of the receive filter bank are illustrated by the scheme of Figure 9.3.

From Figure 9.3 using (9.16), (9.22), and applying the noble identities given in Figure 1.70, we obtain the system represented in Figure 9.4, where parallel to serial (P/S) and serial to parallel (S/P) converters are illustrated in Figures 9.5a and 9.5b, respectively.

Orthogonality conditions in terms of the polyphase components

We observe that the input–output relation of the cascade of the P/S converter, the delay $z^{-D_0} = z^{-1}$, and the S/P converter is expressed by the matrix $z^{-1}\mathbf{I}$, that means that any input is obtained at the output on the same branch with a delay equal to T: therefore the system of Figure 9.4 is equivalent to that of Figure 9.6, with matrix transfer functiongiven by

$$\mathbf{S}(z) = z^{-(\gamma-1)}\,\mathbf{E}^H\left(\frac{1}{z^*}\right) z^{-1}\,\mathbf{I}\,\mathbf{E}(z) = z^{-\gamma}\,\mathbf{E}^H\left(\frac{1}{z^*}\right)\mathbf{E}(z) \tag{9.23}$$

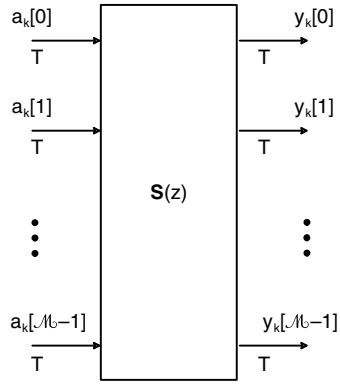


Figure 9.6. Equivalent OFDM system with input–output relation expressed in terms of the matrix $\mathbf{S}(z)$.

From (9.11), in terms of the polyphase components of the transmit filters in (9.15), the orthogonality conditions in the z-transform domain are therefore expressed as

$$\mathbf{E}^H\left(\frac{1}{z^*}\right)\mathbf{E}(z) = \mathbf{I} \tag{9.24}$$

Equivalently, using the polyphase representation of the receive filters in (9.20), we find the conditions

$$\mathbf{R}(z)\,\mathbf{R}^H\left(\frac{1}{z^*}\right) = \mathbf{I} \tag{9.25}$$

OFDM implementation employing a prototype filter

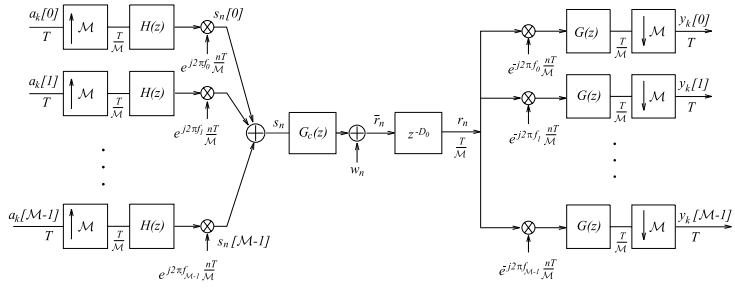
The complexity of an OFDM system can be further reduced by resorting to uniform filter banks [1, 2]. In this case the frequency responses of the various filters are obtained by shifting the frequency response of a *prototype filter* around the carrier frequencies, given by

$$f_i = \frac{i}{T} \qquad i = 0, \ldots, \mathcal{M} - 1 \tag{9.26}$$

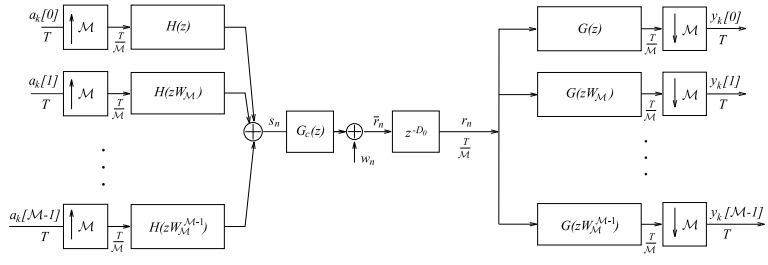
In other words, the spacing in frequency between the subcarriers is $\Delta f = 1/T$.

To derive an efficient implementation of uniform filter banks we consider the scheme represented in Figure 9.7a, where $H(z)$ and $G(z)$ are the transfer functions of the transmit and receive prototype filters, respectively. We consider as transmit prototype filter a causal FIR filter with impulse response $\{h_n\}$, having length $\gamma\mathcal{M}$ and support $\{0, 1, \ldots, \gamma\mathcal{M} - 1\}$; the receive prototype filter is a causal FIR filter matched to the transmit prototype filter, with impulse response given by

$$g_n = h^*_{\gamma\mathcal{M}-n} \qquad \forall n \tag{9.27}$$



(a) General scheme



(b) Equivalent scheme

Figure 9.7. Block diagram of an OFDM system with uniform filter banks: (a) general scheme, (b) equivalent scheme for $f_i = i/T$, $i = 0, \ldots, \mathcal{M} - 1$.

With reference to Figure 9.7a, the i-th subchannel signal at the channel input is given by

$$s_n[i] = e^{j2\pi \frac{in}{\mathcal{M}}} \sum_{k=-\infty}^{\infty} a_k[i]\, h_{n-k\mathcal{M}} \tag{9.28}$$

As

$$e^{j2\pi \frac{in}{\mathcal{M}}} = e^{j2\pi \frac{i(n-k\mathcal{M})}{\mathcal{M}}} \tag{9.29}$$

we obtain

$$s_n[i] = \sum_{k=-\infty}^{\infty} a_k[i]\, h_{n-k\mathcal{M}}\, e^{j2\pi \frac{i(n-k\mathcal{M})}{\mathcal{M}}} = \sum_{k=-\infty}^{\infty} a_k[i]\, h_{n-k\mathcal{M}}[i] \tag{9.30}$$

where

$$h_n[i] = h_n\, e^{j2\pi \frac{in}{\mathcal{M}}} \tag{9.31}$$

Recalling the definition $W_{\mathcal{M}} = e^{-j\frac{2\pi}{\mathcal{M}}}$, the z-transform of $\{h_n[i]\}$ is expressed as

$$H_i(z) = H(zW_{\mathcal{M}}^{i}) \qquad i = 0, \ldots, \mathcal{M} - 1 \tag{9.32}$$

Observing (9.30) and (9.32), we obtain the equivalent scheme of Figure 9.7b.

The scheme of Figure 9.7b may be considered as a particular case of the general OFDM scheme represented in Figure 9.1; in particular, the transfer functions of the filters can be expressed using the polyphase representations, which are, however, simplified with respect to the general case expressed by (9.15) and (9.20), as we show in the following. Observing (9.28) we express the overall signal s_n as

$$s_n = \sum_{i=0}^{\mathcal{M}-1} e^{j2\pi \frac{in}{\mathcal{M}}} \sum_{k=-\infty}^{+\infty} h_{n-k\mathcal{M}}\, a_k[i] \tag{9.33}$$

With the change of variables $n = m\mathcal{M} + \ell$, for $m = -\infty, \ldots, +\infty$, and $\ell = 0, 1, \ldots, \mathcal{M}-1$, we get

$$s_{m\mathcal{M}+\ell} = \sum_{i=0}^{\mathcal{M}-1} e^{j2\pi \frac{i}{\mathcal{M}}(m\mathcal{M}+\ell)} \sum_{k=-\infty}^{+\infty} h_{(m-k)\mathcal{M}+\ell}\, a_k[i] \tag{9.34}$$

Observing that $e^{j2\pi im} = 1$, setting $s_m^{(\ell)} = s_{m\mathcal{M}+\ell}$ and $h_m^{(\ell)} = h_{m\mathcal{M}+\ell}$, and interchanging the order of equations, we express the ℓ-th polyphase component of $\{s_n\}$ as

$$s_m^{(\ell)} = \sum_{k=-\infty}^{+\infty} h_{m-k}^{(\ell)} \sum_{i=0}^{\mathcal{M}-1} W_{\mathcal{M}}^{-i\ell}\, a_k[i] \tag{9.35}$$

The sequences $\{h_m^{(\ell)}\}$, $\ell = 0, \ldots, \mathcal{M}-1$, denote the polyphase components of the prototype filter impulse response, with transfer functions given by

$$H^{(\ell)}(z) = \sum_{m=0}^{\gamma-1} h_m^{(\ell)}\, z^{-m} \qquad \ell = 0, \ldots, \mathcal{M}-1 \tag{9.36}$$

Recalling the definition (1.94) of the DFT operator as $\mathcal{M} \times \mathcal{M}$ matrix, the IDFT of the vector $\mathbf{a}_k$ is expressed as

$$\mathbf{F}_{\mathcal{M}}^{-1}\, \mathbf{a}_k = \mathbf{A}_k = [A_k[0], \ldots, A_k[\mathcal{M}-1]]^T \tag{9.37}$$

We find that the inner summation in (9.35) yields

$$\frac{1}{\mathcal{M}} \sum_{i=0}^{\mathcal{M}-1} W_{\mathcal{M}}^{-i\ell}\, a_k[i] = A_k[\ell] \qquad \ell = 0, 1, \ldots, M-1 \tag{9.38}$$

and

$$s_m^{(\ell)} = \sum_{k=-\infty}^{\infty} \mathcal{M}\, h_{m-k}^{(\ell)}\, A_k[\ell] = \sum_{p=-\infty}^{\infty} \mathcal{M}\, h_p^{(\ell)}\, A_{m-p}[\ell] \tag{9.39}$$

Including the factor $\mathcal{M}$ in the definition of the prototype filter impulse response, or in $\mathcal{M}$ gain factors that establish the statistical power levels to be assigned to each subchannel signal, an efficient implementation of a uniform transmit filter bank is given by an IDFT, a polyphase network with $\mathcal{M}$ branches, and a P/S converter, as illustrated in Figure 9.8.

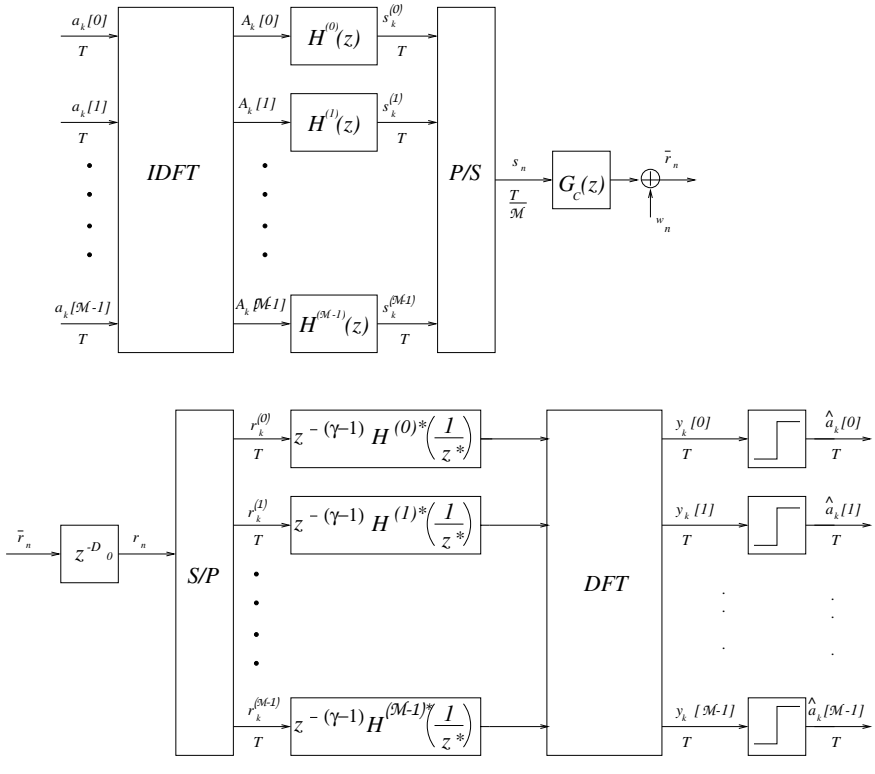


Figure 9.8. Block diagram of an OFDM system with efficient implementation.

Observing (9.35), we find that the matrix of the transfer functions of the polyphase components of the transmit filters is given by

$$\mathbf{E}(z) = \text{diag}\{H^{(0)}(z), \ldots, H^{(\mathcal{M}-1)}(z)\}\mathbf{F}_{\mathcal{M}}^{-1} \tag{9.40}$$

Therefore the vector of the transfer functions of the transmit filters is expressed as

$$\mathbf{h}^T(z) = z^{-(\mathcal{M}-1)}\,\mathbf{e}^H\left(\frac{1}{z^*}\right)\text{diag}\{H^{(0)}(z^{\mathcal{M}}), \ldots, H^{(\mathcal{M}-1)}(z^{\mathcal{M}})\}\mathbf{F}_{\mathcal{M}}^{-1} \tag{9.41}$$

We note that we would arrive at the same result by applying the notion of prototype filter with the condition (9.26) to (9.15).

The vector of the transfer functions of a receive filter bank, which employs a prototype filter with impulse response given by (9.27), is immediately obtained by applying (9.22), with the matrix of the transfer functions of the polyphase components given by (9.40). Therefore we get

$$\mathbf{g}(z) = z^{-\gamma\mathcal{M}+\mathcal{M}-1}\,\mathbf{F}_{\mathcal{M}}\,\text{diag}\left\{H^{(0)*}\left(\frac{1}{z^{*\mathcal{M}}}\right), \ldots, H^{(\mathcal{M}-1)*}\left(\frac{1}{z^{*\mathcal{M}}}\right)\right\}\mathbf{e}(z) \tag{9.42}$$

Hence an efficient implementation of a uniform receive filter bank, also illustrated in Figure 9.8, is given by a S/P converter, a polyphase network with $\mathcal{M}$ branches, and a DFT; we note that the filter of the i-th branch at the receiver is matched to the filter of the corresponding branch at the transmitter.

With reference to Figure 9.7a, it is interesting to derive (9.42) by observing the relation between the received sequence $r_n = \bar{r}_{n-D_0}$ and the output of the i-th subchannel $y_k[i]$, given by

$$y_k[i] = \sum_{n=-\infty}^{+\infty} g_{k\mathcal{M}-n}\, e^{-j2\pi \frac{i}{T} n \frac{T}{\mathcal{M}}}\, r_n \tag{9.43}$$

With the change of variables $n = m\mathcal{M}+\ell$, for $m = -\infty, \ldots, +\infty$, and $\ell = 0, \ldots, \mathcal{M}-1$, and recalling the expression of the prototype filter impulse response given by (9.27), we obtain

$$y_k[i] = \sum_{\ell=0}^{\mathcal{M}-1} \sum_{m=-\infty}^{+\infty} h^*_{(\gamma-k+m)\mathcal{M}+\ell}\, e^{-j2\pi \frac{i}{\mathcal{M}}(m\mathcal{M}+\ell)}\, r_{m\mathcal{M}+\ell} \tag{9.44}$$

Observing that $e^{-j2\pi \frac{i}{\mathcal{M}} m\mathcal{M}} = 1$, setting $r_m^{(\ell)} = r_{m\mathcal{M}+\ell}$, and $h_m^{(\ell)*} = h^*_{m\mathcal{M}+\ell}$, and interchanging the order of equations, we get

$$y_k[i] = \sum_{\ell=0}^{\mathcal{M}-1} e^{-j\frac{2\pi}{\mathcal{M}} i\ell} \sum_{m=-\infty}^{\infty} h^{(\ell)*}_{\gamma+m-k}\, r_m^{(\ell)} \tag{9.45}$$

Using the relation $e^{-j\frac{2\pi}{\mathcal{M}} i\ell} = W_{\mathcal{M}}^{i\ell}$, we finally find the expression

$$y_k[i] = \sum_{\ell=0}^{\mathcal{M}-1} W_{\mathcal{M}}^{i\ell} \sum_{m=-\infty}^{\infty} h^{(\ell)*}_{\gamma+m-k}\, r_m^{(\ell)} \tag{9.46}$$

Provided the orthogonality conditions are satisfied, from the output samples $y_k[i]$, $i = 0, \ldots, \mathcal{M}-1$, threshold detectors may be employed to yield the detected symbols $\hat{a}_k[i]$, $i = 0, \ldots, \mathcal{M}-1$, with a certain delay D.

As illustrated in Figure 9.8, all filtering operations are carried out at the low rate $1/T$. Also note that in practice the FFT and the inverse FFT are used in place of the DFT and the IDFT, respectively, thus further reducing the computational complexity.

9.4 Non-critically sampled filter banks

For the above discussion on the efficient realization of OFDM systems, we referred to Figure 9.8, where the number of subchannels, $\mathcal{M}$, coincides with the interpolation factor of the modulation filters in the transmit filter bank. These systems are called filter bank systems with critical sampling or, in short, *critically sampled filter banks*.

We now examine the general case of $\mathcal{M}$ modulators where each use an interpolation filter by a factor $\mathcal{K} > \mathcal{M}$: this system is called *non-critically sampled*. In principle the schemes

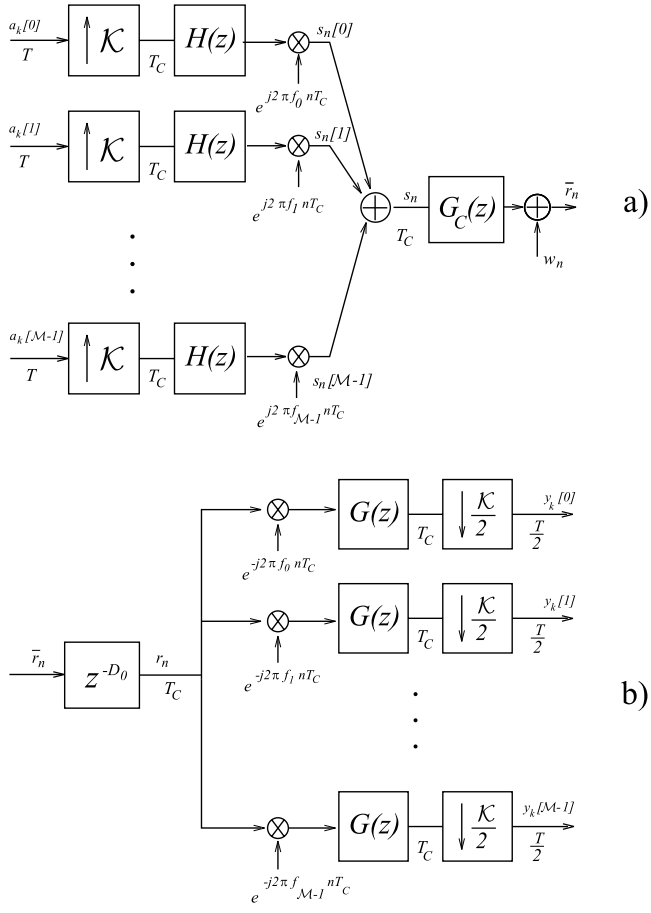


Figure 9.9. Block diagram of (a) transmitter and (b) receiver in a transmission system employing non-critically sampled filter banks, with $\mathcal{K} > \mathcal{M}$ and $f_i = (i\mathcal{K})/(\mathcal{M}T) = i/(\mathcal{M}T_c)$.

of transmit and receive non-critically sampled filter banks are illustrated in Figure 9.9. As in critically sampled systems, also in non-critically sampled systems it is advantageous to choose each transmit filter as the frequency-shifted version of a prototype filter with impulse response $\{h_n\}$, defined over a discrete-time domain with sampling period $T_c = T/\mathcal{K}$. At the receiver, each filter is the frequency-shifted version of a prototype filter with impulse response $\{g_n\}$, also defined over a discrete-time domain with sampling period $T_c = T/\mathcal{K}$.

As depicted in Figure 9.10, each subchannel filter has a bandwidth equal to $\mathcal{K}/(\mathcal{M}T)$, larger than $1/T$. Maintaining a spacing between subcarriers of $\Delta f = \mathcal{K}/(\mathcal{M}T)$, it is easier to avoid spectral overlapping between subchannels and consequently to avoid ICI. It is also possible to choose $\{h_n\}$, e.g., as the impulse response of a square root raised cosine filter, such that, at least for an ideal channel, the orthogonality conditions are satisfied and ISI is also avoided. We note that this advantage is obtained at the expense of a larger

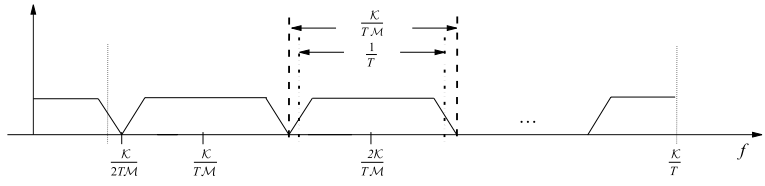


Figure 9.10. Filter frequency responses in a non-critically sampled system.

bandwidth required for the transmission channel, that changes from $\mathcal{M}/T$ for critically sampled systems to $\mathcal{K}/T$ for non-critically sampled systems. Therefore the system requires an excess bandwidth given by $(\mathcal{K}-\mathcal{M})/\mathcal{M}$.

Also for non-critically sampled filter banks it is possible to obtain an efficient implementation using the discrete Fourier transform [3, 4]. The transmitted signal is expressed as a function of the input symbol sequences as

$$s_n = \sum_{i=0}^{\mathcal{M}-1} e^{j2\pi \frac{i\mathcal{K}}{\mathcal{M}T} n \frac{T}{\mathcal{K}}} \sum_{k=-\infty}^{\infty} a_k[i]\, h_{n-k\mathcal{K}} \tag{9.47}$$

or, equivalently,

$$s_n = \sum_{k=-\infty}^{\infty} h_{n-k\mathcal{K}} \sum_{i=0}^{\mathcal{M}-1} a_k[i]\, W_{\mathcal{M}}^{-in} \tag{9.48}$$

With the change of indices

$$n = m\mathcal{M} + \ell \qquad m \in \mathcal{Z} \qquad \ell = 0, 1, \ldots, \mathcal{M}-1 \tag{9.49}$$

(9.48) becomes

$$s_{m\mathcal{M}+\ell} = \sum_{k=-\infty}^{\infty} h_{m\mathcal{M}-k\mathcal{K}+\ell} \sum_{i=0}^{\mathcal{M}-1} W_{\mathcal{M}}^{-i\ell}\, a_k[i] \tag{9.50}$$

Using the definition of the IDFT (9.38), apart from a factor $\mathcal{M}$ that can be included in the impulse response of the filter, and introducing the following polyphase representation of the transmitted signal

$$s_m^{(\ell)} = s_{m\mathcal{M}+\ell} \tag{9.51}$$

we obtain

$$s_m^{(\ell)} = \sum_{k=-\infty}^{\infty} h_{m\mathcal{M}-k\mathcal{K}+\ell}\, A_k[\ell] \tag{9.52}$$

By analogy with (1.561), (9.52) is obtained by interpolation of the sequence $\{A_k[\ell]\}$ by a factor $\mathcal{K}$, followed by decimation by a factor $\mathcal{M}$. From (1.569) and (1.570), we introduce the change of indices

$$p = \left\lfloor \frac{m\mathcal{M}}{\mathcal{K}} \right\rfloor - k \tag{9.53}$$

and

$$\Delta_m = \frac{m\mathcal{M}}{\mathcal{K}} - \left\lfloor \frac{m\mathcal{M}}{\mathcal{K}} \right\rfloor = \frac{(m\mathcal{M})_{mod\,\mathcal{K}}}{\mathcal{K}} \tag{9.54}$$

Using (1.576) it results in

$$\begin{aligned} s_m^{(\ell)} &= \sum_{p=-\infty}^{+\infty} h_{(p+\Delta_m)\mathcal{K}+\ell}\, A_{\left\lfloor \frac{m\mathcal{M}}{\mathcal{K}} \right\rfloor - p}[\ell] \\ &= \sum_{p=-\infty}^{+\infty} h_{p\mathcal{K}+\ell+(m\mathcal{M})_{mod\,\mathcal{K}}}\, A_{\left\lfloor \frac{m\mathcal{M}}{\mathcal{K}} \right\rfloor - p}[\ell] \end{aligned}$$

Letting

$$h_{p,m}^{(\ell)} = h_{p\mathcal{K}+\ell+(m\mathcal{M})_{mod\,\mathcal{K}}} \qquad p, m \in \mathcal{Z} \qquad \ell = 0, 1, \ldots, \mathcal{M}-1 \tag{9.55}$$

we obtain

$$s_m^{(\ell)} = \sum_{p=0}^{\infty} h_{p,m}^{(\ell)}\, A_{\left\lfloor \frac{m\mathcal{M}}{\mathcal{K}} \right\rfloor - p}[\ell] \tag{9.56}$$

The efficient implementation of the transmit filter bank is illustrated in Figure 9.11. We note that the system is now periodically time-varying, i.e. the impulse response of the filter components cyclically changes. The $\mathcal{M}$ elements of an IDFT output vector are input to $\mathcal{M}$ delay lines. Also note that within a modulation interval of duration T, the samples stored in some of the delay lines are used to produce more than one sample of the transmitted signal. Therefore the P/S element used for the realization of critically sampled filter banks needs to be replaced by a commutator. At instant $nT/\mathcal{K}$, the commutator is linked to the $\ell = n_{mod\,\mathcal{M}}$-th filtering element. The transmit signal s_n is then computed by

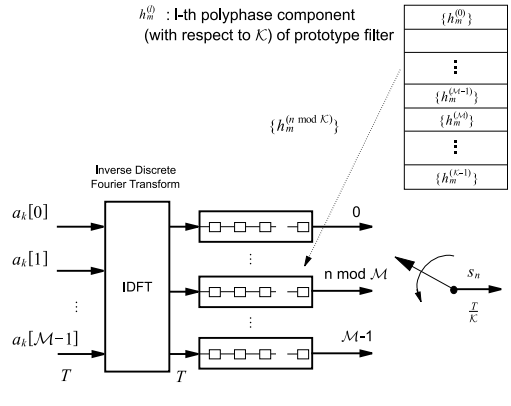


Figure 9.11. Efficient implementation of the transmitter of a system employing non-critically sampled filter banks; the filter components are periodically time-varying.

convolving the signal samples stored in the ℓ-th delay line with the $n_{mod\,\mathcal{K}}$-th polyphase component of the $T/\mathcal{K}$-spaced-coefficients prototype filter. In other terms, each element of the IDFT output frame is filtered by a periodically time-varying filter with period equal to $[l.c.m.(\mathcal{M},\mathcal{K})]T/\mathcal{K}$, where $l.c.m.(\mathcal{M},\mathcal{K})$ denotes the least common multiple of $\mathcal{M}$ and $\mathcal{K}$.

Likewise, the non-critically sampled filter bank at the receiver can also be efficiently implemented using the DFT. In particular, we consider the case of downsampling of the subchannel output signals by a factor $\mathcal{K}/2$, which yields samples at each subchannel output at an (over)sampling rate equal to $2/T$. With reference to Figure 9.9b, we observe that the output sequence of the i-th subchannel is given by

$$y_{n'}[i] = \sum_{n=-\infty}^{\infty} g_{n'\frac{\mathcal{K}}{2}-n}\, e^{-j2\pi\frac{in}{\mathcal{M}}}\, r_n \tag{9.57}$$

where $g_n = h^*_{\gamma\mathcal{M}-n}$.

With the change of indices

$$n = m\mathcal{M} + \ell \qquad m \in \mathcal{Z} \qquad \ell = 0, 1, \ldots, \mathcal{M}-1 \tag{9.58}$$

and letting $r_m^{(\ell)} = r_{m\mathcal{M}+\ell}$, from (9.57) we get

$$y_{n'}[i] = \sum_{\ell=0}^{\mathcal{M}-1} \left(\sum_{m=-\infty}^{\infty} g_{n'\frac{\mathcal{K}}{2}-m\mathcal{M}-\ell}\, r_m^{(\ell)} \right) W_{\mathcal{M}}^{i\ell} \tag{9.59}$$

We note that in (9.59) the term within parenthesis may be viewed as an interpolation by a factor $\mathcal{M}$ followed by a decimation by a factor $\mathcal{K}/2$.

Letting

$$q = \left\lfloor \frac{n'\mathcal{K}}{2\mathcal{M}} \right\rfloor - m \tag{9.60}$$

and

$$\Delta_{n'} = \frac{n'\mathcal{K}}{2\mathcal{M}} - \left\lfloor \frac{n'\mathcal{K}}{2\mathcal{M}} \right\rfloor = \frac{(n'\mathcal{K}/2)_{mod\,\mathcal{M}}}{\mathcal{M}} \tag{9.61}$$

the terms within parenthesis in (9.59) can be written as

$$\sum_{q=-\infty}^{\infty} g_{q\mathcal{M}+(n'\frac{\mathcal{K}}{2})_{mod\,\mathcal{M}}-\ell}\, r^{(\ell)}_{\left\lfloor \frac{n'\mathcal{K}/2}{\mathcal{M}} \right\rfloor - q} \tag{9.62}$$

Introducing the $\mathcal{M}$ periodically time-varying filters,

$$g_{q,n'}^{(\ell)} = g_{q\mathcal{M}+(n'\frac{\mathcal{K}}{2})_{mod\,\mathcal{M}}-\ell} \qquad q, n' \in \mathcal{Z} \qquad \ell = 0, 1, \ldots, \mathcal{M}-1 \tag{9.63}$$

and defining the DFT input samples

$$u_{n'}^{(\ell)} = \sum_{q=-\infty}^{\infty} g_{q,n'}^{(\ell)}\, r^{(\ell)}_{\left\lfloor \frac{n'\mathcal{K}}{2\mathcal{M}} \right\rfloor - q} \tag{9.64}$$

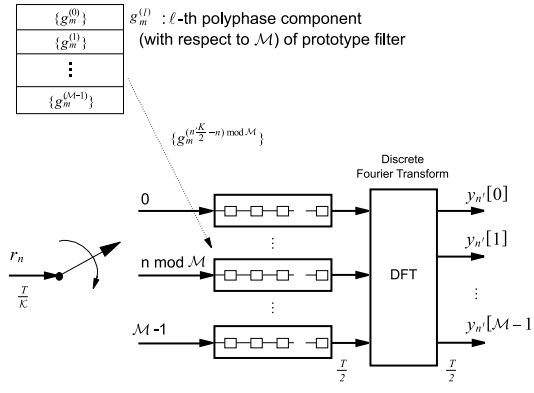


Figure 9.12. Efficient implementation of the receiver of a system employing non-critically sampled filter banks; the filter components are periodically time-varying (see (9.63)).

(9.59) becomes

$$y_{n'}[i] = \sum_{\ell=0}^{\mathcal{M}-1} u_{n'}^{(\ell)} \, W_{\mathcal{M}}^{i\ell} \tag{9.65}$$

The efficient implementation of the receive filter bank is illustrated in Figure 9.12, where we assume for the received signal the same sampling rate of $\mathcal{K}/T$ as for the transmitted signal, and a downsampling factor $\mathcal{K}/2$, so that the samples at each subchannel output are obtained at a sampling rate equal to $2/T$. Note that the delay element z^{-D_0} at the receiver input has been omitted, as the optimum timing phase for each subchannel can be recovered by using per-subchannel fractionally spaced equalization, as discussed in Section 8.4 for single-carrier modulation. Also note that within a modulation interval of duration T, more than one sample is stored in some of the delay lines to produce the DFT input vectors. Therefore the S/P element used for the realization of critically sampled filter banks needs to be replaced by a commutator. After the $\mathcal{M}$ elements of a DFT input vector are produced, the commutator is circularly rotated $\mathcal{K}/2$ steps clockwise from its current position, allowing a set of $\mathcal{K}/2$ consecutive received samples to be input into the delay lines. The content of each delay line is then convolved with one of the $\mathcal{M}$ polyphase components of the $T/\mathcal{K}$-spaced-coefficients receive prototype filter. A similar structure is obtained if in general a downsampling factor $\mathcal{K}' \leq \mathcal{K}$ is considered.

9.5 Examples of OFDM systems

We consider three simple examples of critically sampled filter bank modulation systems. For practical applications, equalization techniques and possibly non-critically sampled filter bank realizations are required, as will be discussed in the following sections.

Discrete multitone (DMT)

The transmit and receive filter banks use a prototype filter with impulse response given by [5, 6, 7, 8]

$$h_n = \begin{cases} 1 & \text{if } 0 \le n \le \mathcal{M}-1 \\ 0 & \text{otherwise} \end{cases} \tag{9.66}$$

The impulse responses of the polyphase components of the prototype filter are given by

$$\left\{h_n^{(\ell)}\right\} = \{\delta_n\} \qquad \ell = 0, \ldots, \mathcal{M}-1 \tag{9.67}$$

and we can easily verify that the orthogonality conditions (9.6) are satisfied.

As shown in Figure 9.13, because the frequency responses of the polyphase components are constant, we obtain directly the transmit signal by applying a P/S conversion at the output of the IDFT. Assuming an ideal channel, at the receiver a S/P converter forms blocks of $\mathcal{M}$ samples, with boundaries between blocks placed so that each block at the output of the IDFT at the transmitter is presented unchanged at the input of the DFT. At the DFT output, the input blocks of $\mathcal{M}$ symbols are reproduced without distortion with a delay equal to T. We note, however, that the orthogonality conditions are satisfied only if the channel is ideal.

From the frequency response of the prototype filter,

$$H\left(e^{j2\pi f\frac{T}{\mathcal{M}}}\right) = e^{-j2\pi f\frac{\mathcal{M}-1}{2}\frac{T}{\mathcal{M}}}\,\frac{\sin(\pi f T)}{\sin(\pi f T/\mathcal{M})} \tag{9.68}$$

using (9.32) the frequency responses of the individual subchannel filters $H_i(z)$ are obtained. Figure 9.14 shows the amplitude of the frequency responses of adjacent subchannel filters, obtained for $f \in \left(0, 0.06\frac{\mathcal{M}}{T}\right)$ and $\mathcal{M} = 64$. We note that the choice of a rectangular window of length $\mathcal{M}$ as impulse response of the baseband prototype filter leads to a significant overlapping of spectral components of transmitted signals in adjacent subchannels.

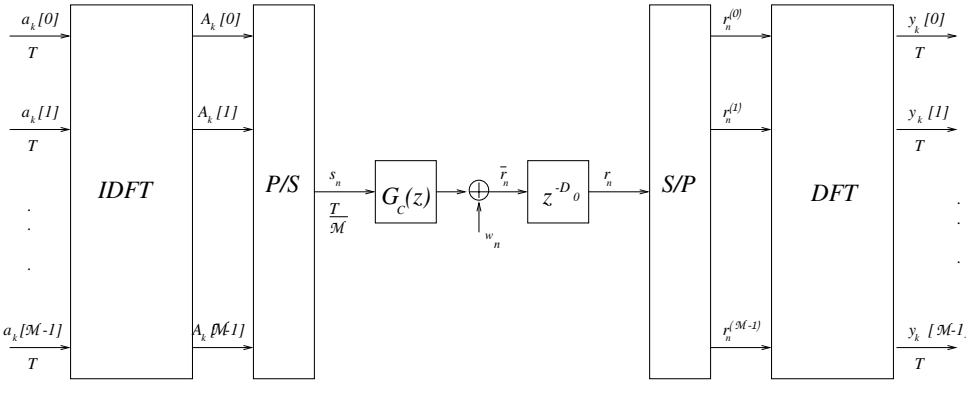


Figure 9.13. Block diagram of an OFDM system with impulse response of the prototype filter given by a rectangular window of length $\mathcal{M}$.

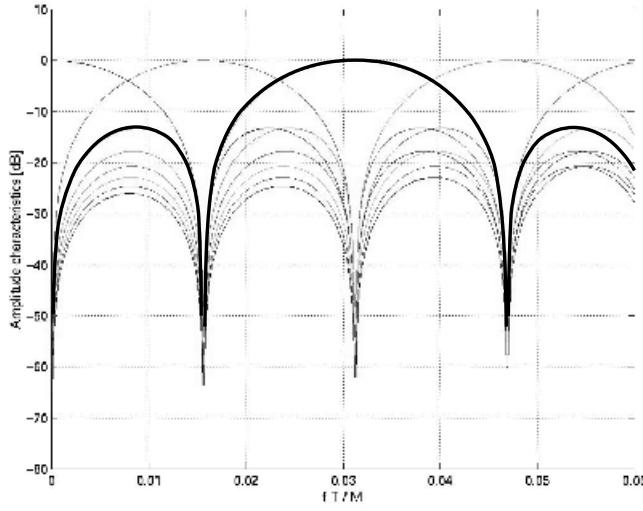


Figure 9.14. Amplitude of the frequency responses of adjacent subchannel filters in a DMT system for $f \in \left(0, 0.06\frac{\mathcal{M}}{T}\right)$ and $\mathcal{M} = 64$. [*From [4], © 2002 IEEE*]

Filtered multitone (FMT)

The transmit and receive filter banks use a prototype filter with frequency response given by [3, 4]

$$H\left(e^{j2\pi f\frac{T}{\mathcal{M}}}\right) = \begin{cases} \left|\dfrac{1+e^{-j2\pi fT}}{1+\rho e^{-j2\pi fT}}\right| & \text{if } |f| \le \dfrac{1}{2T} \\ 0 & \text{otherwise} \end{cases} \tag{9.69}$$

where the parameter $0 \le \rho \le 1$ controls the spectral roll-off of the filter. The frequency response exhibits spectral nulls at the band edges and, when used as the prototype filter characteristic, leads to transmission free of ICI but with ISI within a subchannel. For $\rho \to 1$, the frequency characteristic of each subchannel exhibits steep roll-off towards the band edge frequencies. On the other hand, for $\rho = 0$ the partial-response class I characteristic is obtained. In general, it is required that at the output of each subchannel the ICI is negligible with respect to the noise.

The amplitude of the frequency responses of subchannel filters obtained with a minimum-phase prototype FIR filter for $f \in \left(0, 0.06\frac{\mathcal{M}}{T}\right)$, and design parameters $\mathcal{M} = 64$, $\gamma = 10$, and $\rho = 0.1$ are illustrated in Figure 9.15.

Discrete wavelet multitone (DWMT)

As illustrated by the subchannel frequency responses in Figure 9.16, obtained for $f \in \left(0, 0.06\frac{\mathcal{M}}{T}\right)$ and $\mathcal{M} = 64$, in general, DWMT [9] has a higher spectral containment

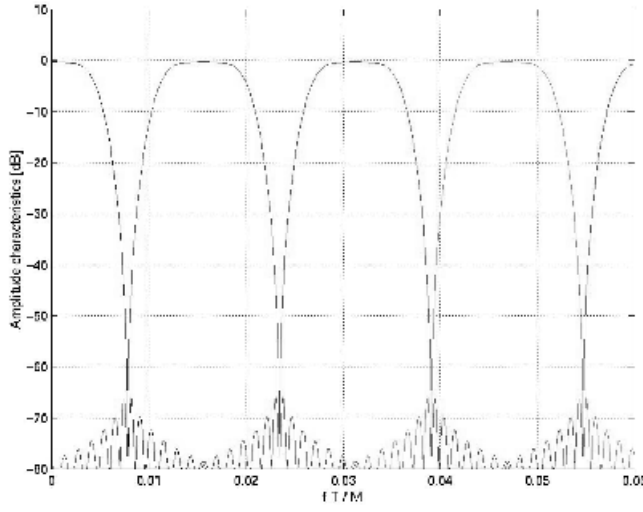


Figure 9.15. Amplitude of the frequency responses of adjacent subchannel filters in an FMT system for $f \in \left(0, 0.06\frac{\mathcal{M}}{T}\right)$, and design parameters $\mathcal{M} = 64$, $\gamma = 10$, and $\rho = 0.1$. [*From [4], © 2002 IEEE*]

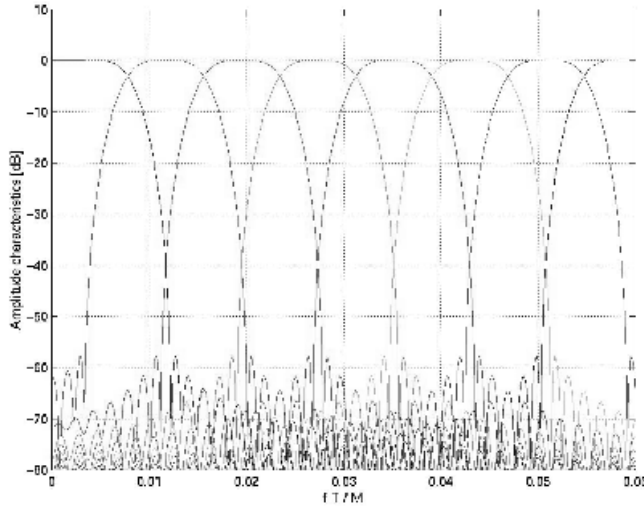


Figure 9.16. Amplitude of frequency responses of adjacent subchannel filters in a DWMT system for $f \in \left(0, 0.06\frac{\mathcal{M}}{T}\right)$ and $\mathcal{M} = 64$. [*From [4], © 2002 IEEE*]

of individual subchannel signals as compared to DMT. The orthogonality conditions are satisfied; each subchannel, however, requires a bandwidth larger than $1/T$.

In DWMT modulation, all signal processing operations involve real signals. Therefore, for the same number of dimensions per modulation interval of the transmitted signal, the minimum bandwidth of a subchannel for DWMT is half the minimum bandwidth of a subchannel for DMT or FMT modulation. The implementation of filter banks for DWMT is examined in Section 9.9.

9.6 Equalization of OFDM systems

Interpolator filter and virtual subchannels

In order to simplify the analysis, in this section we consider the case of a modulated signal with a sampling frequency of $\mathcal{M}/T$. The extension to the general case is obtained by considering a sampling frequency of $\mathcal{K}/T$, with $\mathcal{K} \geq \mathcal{M}$.

The signal $\{s_n\}$ with sampling rate $\mathcal{M}/T$ that we obtain at the output of the transmit filter bank must be converted into an analog signal before being sent over the transmission channel with impulse response $g_{Ch}(t)$, $t \in \Re$.

Assuming $\{s_n\}$ is a real signal, we refer to the scheme of Figure 9.17, where the task of the transmit analog filter g_{Tx}, together with the DAC interpolator filter g_I, is to attenuate the spectral components of $\{s_n\}$ at frequencies higher than $\mathcal{M}/(2T)$. Therefore aliasing is avoided at the receiver by sampling the signal with rate $\mathcal{M}/T$ at the output of the anti-aliasing filter g_{Rc}. We note that the filter g_{Rc} does not attempt to perform channel equalization, which would be difficult to implement whenever the channel frequency response presents large attenuations at frequencies within the passband, and is therefore entirely carried out in the digital domain.

With reference to Figure 9.17, to simplify the specifications of the transmit analog filter it is convenient to interpolate the signal $\{s_n\}$ by a digital filter before D/A conversion (see footnote 4 on page 339).

To further simplify the transmit filters g_I and g_{Tx}, it is possible to increase the transition band of $\{s_n\}$ by avoiding transmission on subchannels near the frequency $\mathcal{M}/(2T)$; in other words, we transmit sequences of all zeros, $a_k[i] = 0$, $\forall k$, for $i = (\mathcal{M}/2) - N_{CV}, \ldots, (\mathcal{M}/2), \ldots, (\mathcal{M}/2) + N_{CV}$. These subchannels are generally called *virtual channels* and their number, $2N_{CV} + 1$, may be a non-negligible percentage of $\mathcal{M}$; this is usually the case in DMT systems because of the large support of the prototype filter in the frequency domain. In FMT systems, choosing $N_{CV} = 1$ or 2 is sufficient. Typically, a square

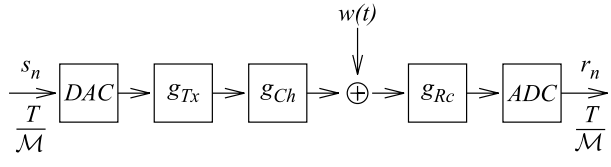


Figure 9.17. Baseband analog transmission of an OFDM signal.

root raised cosine filter, with Nyquist frequency equal to $\mathcal{M}/(2T)$, is selected to implement the interpolator filter and the anti-aliasing filter.

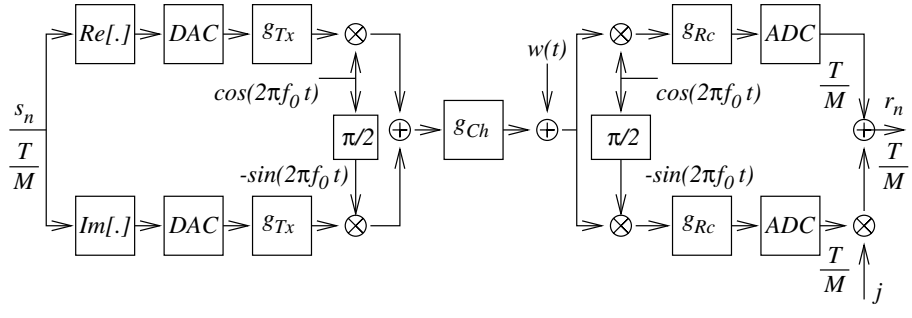
We recall now the conditions to obtain a real-valued transmitted signal $\{s_n\}$. For OFDM systems with the efficient implementation illustrated in Figure 9.8, it is sufficient that the coefficients of the prototype filter and the samples $A_k[i]$, $\forall k$, $i = 0, \ldots, \mathcal{M}-1$, are real-valued. Observing (9.38), the latter condition implies that the following Hermitian symmetry conditions must be satisfied

$$\begin{aligned} &a_k[0] \qquad a_k\left[\frac{\mathcal{M}}{2}\right] \in \Re \\ &a_k[i] = a_k^*[\mathcal{M}-i] \qquad i = 1, \ldots, \frac{\mathcal{M}}{2}-1 \end{aligned} \tag{9.70}$$

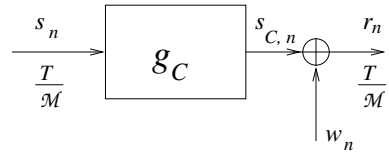
In this case, the symmetry conditions (9.70) also allow a further reduction of the implementation complexity of the IDFT and DFT.

When $\{s_n\}$ is a complex signal, the scheme of Figure 9.18 is adopted, where the filters g_{Tx} and g_{Rc} have the characteristics described above and f_0 is the carrier frequency.

We note that this scheme is analogous to that of a QAM system, with the difference that the transmit and receive lowpass filters, with impulse responses g_{Tx} and g_{Rc}, respectively, have different requirements. The baseband equivalent scheme shown in Figure 9.18b is obtained from the passband scheme by the method discussed in Chapter 7. As the receive filters approximate an ideal lowpass filter with bandwidth $\mathcal{M}/(2T)$, the signal-to-noise ratio Γ at the channel output is assumed equal to that obtained at the output of the baseband



(a) General scheme.



(b) Baseband equivalent scheme.

Figure 9.18. (a) Passband analog OFDM transmission scheme; (b) baseband equivalent scheme.

equivalent discrete-time channel, given by

$$\Gamma = \frac{\mathtt{M}_{sc}}{N_0 \mathcal{M}/T} \tag{9.71}$$

assuming the noise $\{w_n\}$ complex-valued with PSD N_0.

Equalization of DMT systems

We consider the baseband equivalent system shown in Figure 9.13, where the impulse response of the channel has support $\{0, 1, \ldots, N_c - 1\}$, with $N_c > 1$. In this case the orthogonality conditions for the DMT system described in Section 9.2 are no longer satisfied: indeed, the transfer matrix $\mathbf{S}(z)$, defined by (9.10) and evaluated for a non-ideal channel, has in general elements different from a delay factor along the main diagonal, meaning the presence of ISI for transmission over the individual subchannels, and non-zero elements off the main diagonal, meaning the presence of ICI. A simple equalization method is based on the concept of circular convolution introduced in Section 1.4, that allows a expression of a convolution in the time domain as a product of finite length vectors in the frequency domain (see (1.107)). Using the method indicated as *Relation 2* on page 23, we extend the block of samples $\mathbf{A}_k$ by repeating $N_c - 1$ elements: in this way we obtain the DMT system illustrated in Figure 9.19.

For the same channel bandwidth and hence for a given transmission rate $\mathcal{M}/T$, the IDFT (modulation) must be carried out at the rate $\frac{1}{T'} = \frac{\mathcal{M}}{(\mathcal{M}+N_c-1)T} < \frac{1}{T}$. After the modulation, each block of samples is cyclically extended by copying the $N_c - 1$ samples $A_k[\mathcal{M} - N_c + 1], \ldots, A_k[\mathcal{M} - 1]$ in front of the block, as shown in Figure 9.19. After

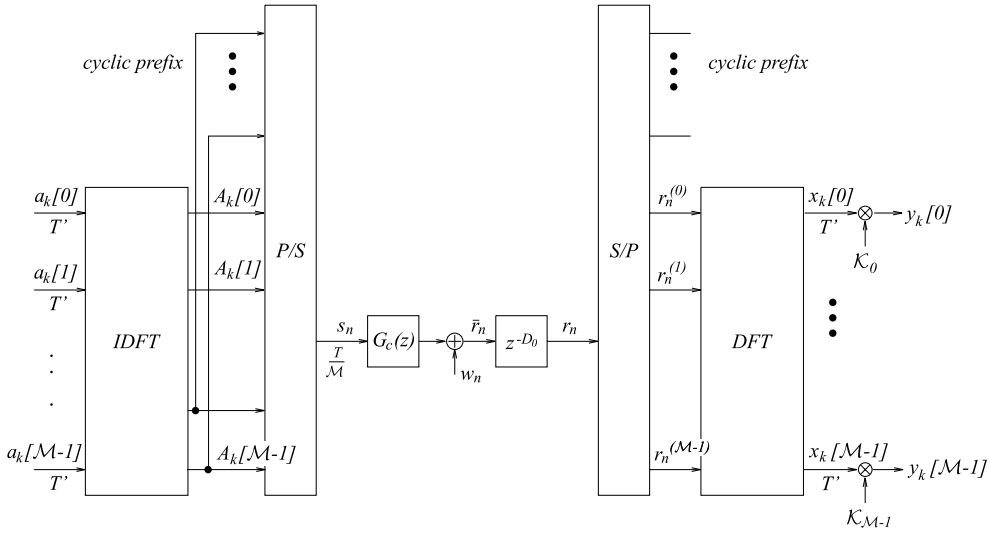


Figure 9.19. Block diagram of a DMT system with cyclic prefix and frequency-domain equalizer.

the P/S conversion, where the $N_c - 1$ samples of the cyclic extension are the first to be sent, the $N_c - 1 + \mathcal{M}$ samples are transmitted over the channel. At the receiver, blocks of samples of length $N_c - 1 + \mathcal{M}$ are taken; the boundaries between blocks are set so that the last $\mathcal{M}$ samples depend only on the elements of only one cyclically extended block of samples. The first $N_c - 1$ samples of a block are discarded.

We now recall the result (1.116). The vector $\mathbf{r}_k$ of the last $\mathcal{M}$ samples of the block received at the k-th modulation interval is expressed as

$$\mathbf{r}_k = \mathbf{\Xi}_k \, \mathbf{g}_C + \mathbf{w}_k \tag{9.72}$$

where $\mathbf{g}_C = [g_{C,0}, \ldots, g_{C,N_c-1}, 0, \ldots, 0]^T$ is the $\mathcal{M}$-component vector of the channel impulse response extended with $\mathcal{M} - N_c$ zeros, $\mathbf{w}_k$ is a vector of additive white Gaussian noise samples, and $\mathbf{\Xi}_k$ is an $\mathcal{M} \times \mathcal{M}$ circulant matrix, given by

$$\mathbf{\Xi}_k = \begin{bmatrix} A_k[0] & A_k[\mathcal{M}-1] & \cdots & A_k[1] \\ A_k[1] & A_k[0] & \cdots & A_k[2] \\ \vdots & \vdots & & \vdots \\ A_k[\mathcal{M}-1] & A_k[\mathcal{M}-2] & \cdots & A_k[0] \end{bmatrix} \tag{9.73}$$

Equation (9.72) is obtained by observing that only the elements of the first N_c columns of the matrix $\mathbf{\Xi}_k$ contribute to the convolution that determines the vector $\mathbf{r}_k$, as the last $\mathcal{M} - N_c$ elements of $\mathbf{g}_C$ are equal to zero. The elements of the last $\mathcal{M} - N_c$ columns of the matrix $\mathbf{\Xi}_k$ are chosen so that the matrix is circulant, even though they might have been chosen arbitrarily. Moreover, we observe that the matrix $\mathbf{\Xi}_k$, being circulant, satisfies the relation

$$\mathbf{F}_{\mathcal{M}} \, \mathbf{\Xi}_k \, \mathbf{F}_{\mathcal{M}}^{-1} = \begin{bmatrix} a_k[0] & 0 & \cdots & 0 \\ 0 & a_k[1] & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & a_k[\mathcal{M}-1] \end{bmatrix} = \text{diag}\{\mathbf{a}_k\} \tag{9.74}$$

Defining the DFT of the vector $\mathbf{g}_C$ as

$$\mathbf{G}_C = [\mathcal{G}_{C,0}, \mathcal{G}_{C,1}, \ldots, \mathcal{G}_{C,\mathcal{M}-1}]^T = \mathbf{F}_{\mathcal{M}} \, \mathbf{g}_C \tag{9.75}$$

and using (9.74), we find that the demodulator output is given by

$$\mathbf{x}_k = \mathbf{F}_{\mathcal{M}} \, \mathbf{r}_k = \text{diag}\{\mathbf{a}_k\} \, \mathbf{G}_C + \mathbf{W}_k \tag{9.76}$$

where $\mathbf{W}_k = \mathbf{F}_{\mathcal{M}} \mathbf{w}_k$ is given by the DFT of the vector $\mathbf{w}_k$. Recalling the properties of $\mathbf{w}_k$, $\mathbf{W}_k$ is a vector of independent Gaussian r.v.s.

Equalizing the channel using the zero-forcing criterion, the signal $\mathbf{x}_k$ (9.76) is multiplied by the diagonal matrix $\mathbf{K}$, whose elements on the diagonal are given by[2]

$$\mathcal{K}_i = [\mathbf{K}]_{i,i} = \frac{1}{\mathcal{G}_{C,i}} \qquad i = 0, 1, \ldots, \mathcal{M} - 1 \tag{9.77}$$

[2] To be precise, the operation indicated by (9.77), rather than equalizing the signal, that is received in the absence of ISI, normalizes the amplitude and adjusts the phase of the desired signal.

Therefore the input to the data detector is given by

$$\mathbf{y}_k = \mathbf{K}\mathbf{x}_k = \mathbf{a}_k + \mathbf{K}\mathbf{W}_k \tag{9.78}$$

We assume that the sequence of input symbol vectors $\{\mathbf{a}_k\}$ is a sequence of i.i.d. random vectors. Equation (9.78) shows that the sequence $\{\mathbf{a}_k\}$ can be detected by assuming transmission over $\mathcal{M}$ independent and orthogonal subchannels in the presence of additive white Gaussian noise.

A drawback of this simple equalization scheme is the reduction in the modulation rate by a factor $(\mathcal{M} + N_c - 1)/\mathcal{M}$. Therefore it is essential that the length of the channel impulse response is much smaller than the number of subchannels, so that the reduction of the modulation rate due to the cyclic extension can be considered negligible.

To reduce the length of the channel impulse response one approach is to equalize the channel before demodulation [8, 10, 11]. With reference to Figure 9.18b, a linear equalizer with input r_n is used; it is usually chosen as the FF filter of a DFE that is determined by imposing a prefixed length of the feedback filter, smaller than the length of the cyclic prefix.

Equalization of FMT systems

We analyze three schemes.

Per-subchannel fractionally spaced equalization. We consider an FMT system with non-critically sampled transmit and receive filter banks, so that transmission within individual subchannels with non-zero excess bandwidth is achieved, and subchannel output signals obtained at a sampling rate equal to $2/T$, as discussed in Section 9.4. We recall that the frequency responses of FMT subchannels are characterized by steep roll-off towards the band-edge frequencies, where they exhibit near spectral nulls. This suggests that per-subchannel decision-feedback equalization be performed to recover the transmitted symbols. The block diagram of an FMT receiver employing per-subchannel fractionally spaced equalization is illustrated in Figure 9.20. Over the i-th subchannel, the DFE is designed for an overall impulse response given by

$$h_{overall,n}[i] = \sum_{n_1=-\infty}^{+\infty} g_{n\mathcal{M}-n_1}[i] \sum_{n_2=0}^{N_c-1} h_{n_1-n_2}[i]\, g_{C,n_2} \qquad n \in \mathcal{Z} \tag{9.79}$$

In the given scheme, the $\mathcal{M}$ DFEs depend on the transmission channel. If the transmission channel is time variant, each DFE must be able to track the channel variations, or it must be recomputed periodically. Error propagation inherent to decision-feedback equalization can be avoided by resorting to precoding techniques, as discussed in Chapter 13. The application of precoding techniques in conjunction with trellis-coded modulation (TCM) for FMT transmission is addressed in [4].

Per-subchannel T-spaced equalization. We consider now an FMT system with critically sampled filter banks, and subchannel output signals obtained at the sampling rate of $1/T$. The high level of spectral containment of the transmit filters suggests that, if the number

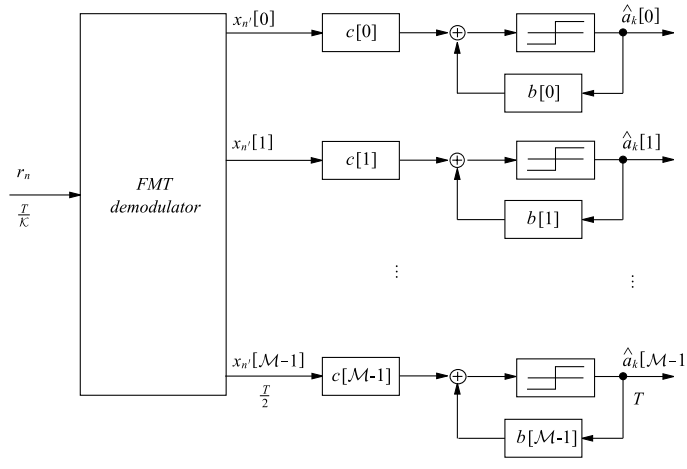


Figure 9.20. Per-subchannel equalization for an FMT system with non-critically sampled filter banks.

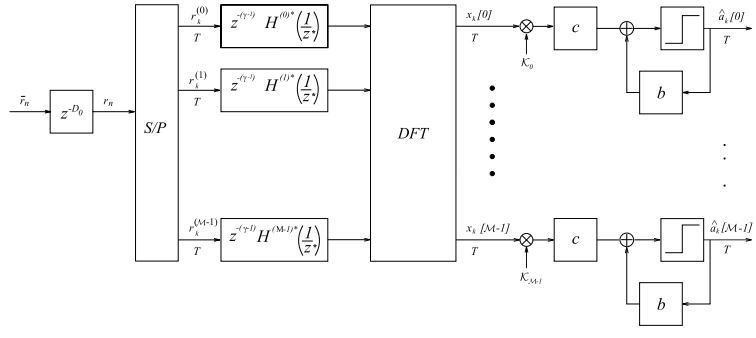
of subchannels is sufficiently high, and the group delay in the passband of the transmission channel is approximately constant, the frequency response of every subchannel becomes approximately a constant. In this case, the effect of the transmission channel is that of multiplying every subchannel signal by a complex value. Therefore, as for DMT systems with cyclic prefix, equalization of the transmission channel can be performed by choosing a suitable constant for every subchannel. We note, however, that, whereas for a DMT system with cyclic prefix the model of the transmission channel as a multiplicative constant for each subchannel is exact if the length of the cyclic prefix is larger than the length of the channel impulse response, for an FMT system such a model is valid only as an approximation. The degree of the approximation depends on the dispersion of the transmission channel and on the number $\mathcal{M}$ of subchannels.

Assuming a constant frequency response for transmission over each subchannel, the equalization scheme is given in Figure 9.21a, where K_i is defined in (9.77), and the DFE is designed to equalize only the cascade of the transmit and receive filters. Using (9.27) and (9.31) we find that the convolution of transmit and receive filters is independent of the subchannel index: in fact, we obtain

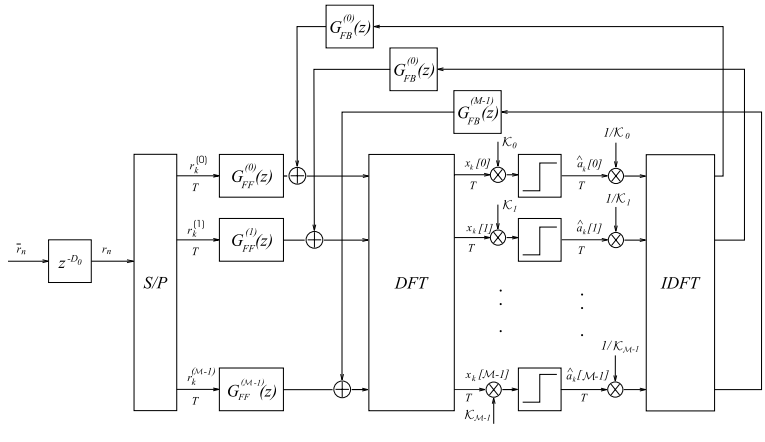
$$\sum_{n_1=0}^{\infty} h_{n_1}[i]\, g_{n\mathcal{M}-n_1}[i] = \sum_{n_1=0}^{\infty} h_{n_1}\, g_{n\mathcal{M}-n_1} = h_{eq,n} \tag{9.80}$$

In this case, all DFEs are equal.

Simplified per-subchannel T-spaced equalization. A further simplification is obtained by using the implementation of Figure 9.21b. The idea is that, in the presence of a transmission channel with flat frequency response for each subchannel, a reconstruction of the signal is achieved by designing the ℓ-th polyphase component of the receive prototype filter, $g^{(\ell)}$, to equalize the corresponding polyphase component $h^{(\ell)}$ of the transmit prototype filter. In



(a) General scheme



(b) Simplified scheme

Figure 9.21. (a) Equalization scheme for FMT in the case of approximately constant frequency response for transmission over each subchannel; (b) simplified scheme.

general, a DFE scheme can be used, where the ℓ-th polyphase components of the receive filters, $g_{FF}^{(\ell)}$ and $g_{FB}^{(\ell)}$, equalize the corresponding i-th polyphase component $h^{(\ell)}$ of the overall subchannel impulse response.

9.7 Synchronization of OFDM systems

Various algorithms may be applied to achieve synchronization of OFDM systems for transmission over dispersive channels, depending on the system and on the type of equalization adopted.

For DMT systems two synchronization processes are identified: synchronization of the clock of the A/D converter at the receiver front-end, or *clock synchronization*, and synchronization of the vector $\mathbf{r}_k$ at the output of the S/P element, or *frame synchronization*.

Clock synchronization guarantees alignment of the timing phase at the receiver with that at the transmitter; frame synchronization, on the other hand, extracts from the sequence of received samples the blocks of $\mathcal{M} + N_c - 1$ samples that form the received frames, and determines the boundaries of the sequence of vectors $\mathbf{r}_k$ that are presented at the input of the DFT. In principle, for a channel input sequence given by $s_0 = 1$, and $s_n = 0$, $n \neq 0$, the channel impulse response of length N_c must appear in the first N_c positions of the receive vector (see Figure 9.19).

For the initial convergence of both synchronization processes, training sequences without cyclic prefix are usually employed [12]. For FMT systems with non-critically sampled filter banks and fractionally-spaced equalization, the synchronization is limited to clock recovery (see Section 8.4).

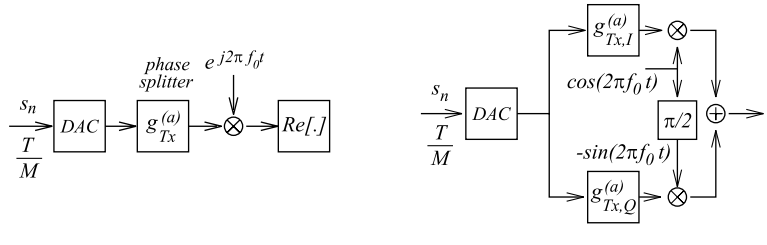
9.8 Passband OFDM systems

For a passband OFDM transmission system, the signal $\{s_n\}$ is in general complex valued and is shifted to passband adopting for example the scheme illustrated in Figure 9.18.

Passband DWMT systems

Suppose a passband OFDM transmission system adopts DWMT modulation. Then, as described in detail in Section 9.9, the filters have real-valued impulse responses, and the baseband transmitted signal is generated by real-valued input symbols. Therefore the signal spectrum has Hermitian symmetry around DC; hence it is sufficient to consider the spectrum in the band $[0, \mathcal{M}/(2T)]$, that corresponds to the analytic signal. In principle, the passband signal is obtained by the scheme illustrated in Figure 9.22a, where the discrete-time signal generated by the DWMT modulator is converted into a continuous time signal by a D/A converter.

The continuous-time signal is then filtered by a phase splitter, $g_{Tx}^{(a)}$, which yields the analytic signal. This signal is shifted to high frequency by a modulator with carrier frequency f_0, and the real part of the resulting signal forms the input signal of the passband transmission channel. In practice, the scheme illustrated in Figure 9.22b is adopted, where $g_{Tx,I}^{(a)} = Re[g_{Tx}^{(a)}]$ and $g_{Tx,Q}^{(a)} = Im[g_{Tx}^{(a)}]$, that is equivalent to a single side band (SSB) modulator (see Example 1.7.4 on page 58). If the phase splitter filter exhibits a suitable



(a) Scheme with complex-valued signals.

(b) Equivalent scheme with real-valued signals.

Figure 9.22. Block diagram of an SSB modulator for a passband DWMT signal.

roll-off around DC we obtain a *vestigial side band* (VSB) modulator.[3] However, because of the difficult recovery of the phase and frequency of the carrier, digital transmission systems using SSB and VSB modulators are characterized by lower performance as compared to systems that consider transmission of the double-sided signal spectrum. To overcome this difficulty and preserve the spectral efficiency of the transmission scheme, a pilot tone may be used to provide the required information for the carrier recovery. The transmission of pilot tones, however, does not represent in many cases a practical solution, as it reduces the power efficiency of the system and introduces one or more spectral lines in the signal spectrum.

Passband DMT and FMT systems

For DMT and FMT systems it is not required that the baseband output signal at the output of the modulator be real-valued. Therefore we remove the constraint that complex-valued input symbols satisfy the Hermitian symmetry conditions (9.70), and we obtain a complex-valued baseband signal. Consequently, the passband signal is given in principle by the scheme illustrated in Figure 9.18a, where g_{Tx} is the real-valued impulse response of a lowpass filter with cut-off frequency equal to $\mathcal{M}/(2T)$. This scheme is equivalent to a modulator for complex-valued signals, sometimes called double sideband modulator with amplitude and phase modulation (DSB-AM/PM); in this case carrier recovery does not represent a difficult problem.

Multiple access DMT and FMT systems. Other difficulties arise, however, in the case of transmission in multiple-access networks. Then two or more users transmit signals simultaneously over subsets of the available subchannels. We recall that in DMT systems the channel impulse response needs to be shortened to reduce the length of the cyclic extension. Consequently in a multiple-access system the impulse response of each user's channel must be shortened. We observe that, even if a cyclic extension of sufficient length is used, the orthogonality conditions are satisfied only if the subchannel signals are synchronous. Because of the spectral overlapping between signals on adjacent subchannels in a DMT system, a signal that is presented at the receiver input with an incorrect timing phase violates the orthogonality conditions, and disturbs many other subchannels: this situation cannot be avoided, for example, when a station sends a signal over a given subchannel without knowledge of the propagation delay.

To solve the problems raised by the transmission of DMT signals in a multiple-access network, we resort to FMT systems, which present large attenuation of the signal spectrum outside the allocated subchannels. In this manner, ICI is avoided even if the various subchannel signals are received from stations without knowledge of the propagation delay.

Comparison between OFDM and QAM systems

It can be shown that OFDM, or multicarrier, systems and QAM, or single carrier, systems achieve the same theoretical performance for transmission over ideal AWGN channels [13].

[3] SSB and VSB modulations are used, for example, for the analog transmission of video signals and can also be considered for digital communication systems.

In practice, however, OFDM systems offer some considerable advantages with respect to CAP/QAM systems.

- OFDM systems achieve higher spectral efficiency if the channel frequency response exhibits large attenuations at frequencies within the passband. In fact, the band used for transmission can be varied by increments equal to the modulation rate $1/T$ Hz, and optimized for each channel. Moreover, if the noise exhibits strong components in certain regions of the spectrum, the total band can be subdivided in two or more sub-bands.
- OFDM systems guarantee a higher robustness with respect to impulse noise. If the average arrival rate of the pulses is lower than the modulation rate, the margin against the impulse noise is of the order of $10\log_{10}(\mathcal{M})$ dB.
- For typical values of $\mathcal{M}$, OFDM systems achieve the same performance as QAM systems with a complexity that can be considerably lower.
- In multiple-access systems, the finer granularity of OFDM systems allows a greater flexibility in the spectrum allocation.

On the other hand, OFDM systems present also a few drawbacks with respect to QAM systems.

- In OFDM systems the transmitted signals exhibit a higher peak-to-average power ratio, that contributes to an increase in the susceptibility of these systems to non-linear distortion.
- Because of the block processing of samples, a higher latency is introduced by OFDM systems in the transmission of information.

9.9 DWMT modulation

In DMT systems the reduction of the modulation rate, equal to the factor $(\mathcal{M}+N_c-1)/\mathcal{M}$, and the consequent need to reduce this penalty by shortening the impulse response of the transmission channel are due to the non-negligible spectral overlap of signals on adjacent subchannels. As a cyclic prefix of length N_c-1 is used for equalization, the orthogonality between signals on different subchannels is verified only for channels with length of the impulse response smaller than or equal to N_c. Ideally, if the frequency response of the prototype filter is chosen such that signals of different subchannels do not overlap, the orthogonality is maintained independently of the impulse response of the transmission channel.

Therefore it is interesting to consider a filter bank modulation scheme where the FIR filters are such that a large attenuation of the filter frequency responses outside the assigned subchannel bands is achieved and that the conditions (9.6) are satisfied. These objectives are achieved by OFDM systems that are usually known as discrete wavelet multitone modulation (DWMT). In a DWMT system, the elements of the input vector $\mathbf{a}_k$ are *real-valued symbols* and the impulse response of the transmit and receive filter banks are also *real-valued*.

Transmit and receive filter banks

To investigate the principles of a DWMT system, we initially consider a uniform filter bank with $2\mathcal{M}$ filters. Let $P_0(z)$ be the transfer function of the prototype filter and $P_i(z)$ the transfer functions of the subchannel filters,

$$P_i(z) = P_0(zW_{2\mathcal{M}}^{i}) \qquad i = 0, \ldots, 2\mathcal{M}-1 \tag{9.81}$$

also let $P^{(i)}(z)$, $i = 0, 1, \ldots, 2\mathcal{M}-1$, be the transfer functions of the polyphase components of $P_0(z)$. The amplitude characteristics of the filters are illustrated in Figure 9.23. The basic idea of DWMT consists in combining pairs of these $2\mathcal{M}$ filters so that they are used with $\mathcal{M}$ real input signals.

We assume that the impulse response of the prototype filter $\{p_n[0]\}$ is real-valued. Therefore the function $\left|P_0\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right)\right|$ is symmetric with respect to $f = 0$. Ideally, the prototype filter is a lowpass filter with bandwidth equal to $1/(2T)$ Hz. From (9.31) we find that the impulse response of the i-th filter is given by $p_n[i] = p_n[0]\, W_{2\mathcal{M}}^{-in}$. We consider now a shifted version of $1/(2T)$ Hz of the original set of $2\mathcal{M}$ frequency responses, obtained by the change of variable $z \to zW_{2\mathcal{M}}^{1/2}$.

We define

$$Q_i(z) = P_0\left(zW_{2\mathcal{M}}^{i+\frac{1}{2}}\right) \qquad 0 \le i \le 2\mathcal{M}-1 \tag{9.82}$$

The amplitude characteristics of the shifted frequency responses are illustrated in Figure 9.24. As the coefficients of $P_0(z)$ are real, the property $Q_{2\mathcal{M}-1-i}(z) = Q_i^*(z^*)$

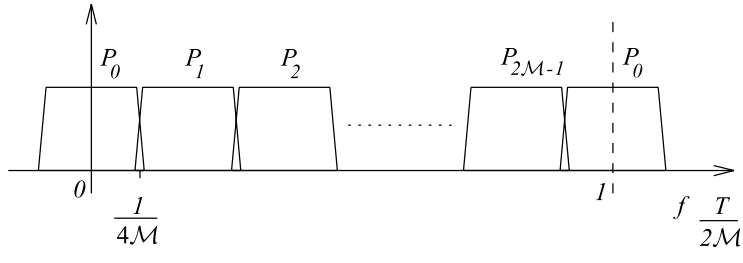


Figure 9.23. Amplitude of the frequency responses of the filters of an OFDM system with $2\mathcal{M}$ subchannels.

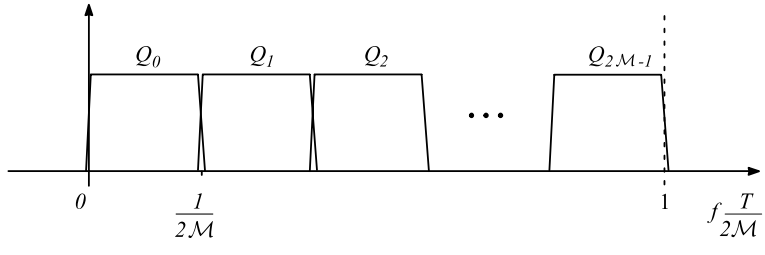


Figure 9.24. Amplitude of the frequency responses of the filters shifted in frequency.

holds, and consequently we get

$$\left|Q_{2\mathcal{M}-1-i}\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right)\right| = \left|Q_i\left(e^{-j\frac{2\pi fT}{2\mathcal{M}}}\right)\right| \tag{9.83}$$

We set

$$U_i(z) = \beta_i\, P_0\left(z\, W_{2\mathcal{M}}^{i+\frac{1}{2}}\right) = \beta_i\, Q_i(z) \qquad 0 \le i \le \mathcal{M}-1 \tag{9.84}$$

$$V_i(z) = \beta_i^*\, P_0\left(z\, W_{2\mathcal{M}}^{-\left(i+\frac{1}{2}\right)}\right) = \beta_i^*\, Q_{2\mathcal{M}-1-i}(z) \qquad 0 \le i \le \mathcal{M}-1 \tag{9.85}$$

$$= \beta_i^*\, Q_i^*(z^*) \tag{9.86}$$

and we define the transfer functions of a new filter bank with $\mathcal{M}$ transmit filters, real input symbol sequences, and modulation rate equal to $2/T$, as

$$H_i(z) = \alpha_i\, U_i(z) + \alpha_i^*\, V_i(z) \qquad 0 \le i \le \mathcal{M}-1 \tag{9.87}$$

In the previous equations α_i and β_i are constants with absolute value equal to one. The amplitude of the frequency response of the filter $H_i(z)$ is illustrated in Figure 9.25. We note that $H_i(z)$ has a frequency response with positive frequency content due to $U_i(z)$, and negative frequency content due to $V_i(z)$.

We assume that the original prototype filter $P_0(z)$ is an FIR filter with length $\gamma\mathcal{M}$ and transfer function given by

$$P_0(z) = \sum_{n=0}^{\gamma\mathcal{M}-1} p_n(0)\, z^{-n} \tag{9.88}$$

The $\mathcal{M}$ filters defined by (9.87) are also FIR filters of length $\gamma\mathcal{M}$ and transfer functions defined as

$$H_i(z) = \sum_{n=0}^{\gamma\mathcal{M}-1} h_n[i]\, z^{-n} \qquad 0 \le i \le \mathcal{M}-1 \tag{9.89}$$

Because the coefficients of $P_0(z)$ are real-valued, the coefficients of $U_i(z)$ are obtained as the complex conjugate of the coefficients of $V_i(z)$. Consequently in (9.87) the coefficients $h_n[i]$, $i = 0, \ldots, \mathcal{M}-1$, are real-valued.

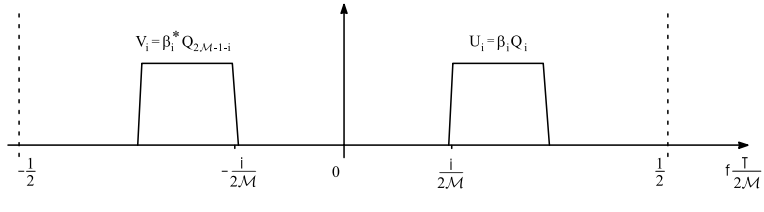


Figure 9.25. Amplitude of the frequency response of the filter $H_i(z)$.

We assume, moreover, that the prototype filter $P_0(z)$ is a linear-phase filter and that the relation $p_{\gamma\mathcal{M}-1-n}[0] = p_n[0]$ holds. Therefore we get

$$P_0^*\left(\frac{1}{z^*}\right) = z^{-(\gamma\mathcal{M}-1)}\, P_0(z) \tag{9.90}$$

The frequency response of the filter can be expressed as

$$P_0\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right) = e^{-j2\pi f\,\frac{\gamma\mathcal{M}-1}{2}\frac{T}{2\mathcal{M}}}\, P_R\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right) \tag{9.91}$$

where $P_R\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right)$ is a real-valued function.

We choose the values of the constants β_i so that $U_i(z)$ and $V_i(z)$ have the same linear phase as $P_0(z)$; observing that

$$U_i\left(e^{j2\pi\frac{fT}{2\mathcal{M}}}\right) = \beta_i\, W_{2\mathcal{M}}^{-\left(i+\frac{1}{2}\right)\frac{\gamma\mathcal{M}-1}{2}}\, e^{-j2\pi f\frac{\gamma\mathcal{M}-1}{2}\frac{T}{2\mathcal{M}}}\, P_R\left(e^{j2\pi\left(\frac{fT}{2\mathcal{M}}-\frac{i+1/2}{2\mathcal{M}}\right)}\right) \tag{9.92}$$

we let

$$\beta_i = W_{2\mathcal{M}}^{\left(i+\frac{1}{2}\right)\frac{\gamma\mathcal{M}-1}{2}} \tag{9.93}$$

Therefore we get

$$\begin{aligned} U_i\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right) &= e^{-j2\pi f\frac{\gamma\mathcal{M}-1}{2}\frac{T}{2\mathcal{M}}}\, P_R\left(e^{j2\pi\left(\frac{fT}{2\mathcal{M}}-\frac{i+1/2}{2\mathcal{M}}\right)}\right) \\ V_i\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right) &= e^{-j2\pi f\frac{\gamma\mathcal{M}-1}{2}\frac{T}{2\mathcal{M}}}\, P_R\left(e^{j2\pi\left(\frac{fT}{2\mathcal{M}}+\frac{i+1/2}{2\mathcal{M}}\right)}\right) \end{aligned} \tag{9.94}$$

and the functions $U_i(z)$ and $V_i(z)$ indeed exhibit the same linear phase as $P_0(z)$. Because $U_i(z)$ and $V_i(z)$ have a linear phase, analogously to (9.90), the following relations hold:

$$\begin{aligned} U_i^*\left(\frac{1}{z^*}\right) &= z^{-(\gamma\mathcal{M}-1)}\, U_i(z) \\ V_i^*\left(\frac{1}{z^*}\right) &= z^{-(\gamma\mathcal{M}-1)}\, V_i(z) \end{aligned} \tag{9.95}$$

Moreover, we assume that the receive filters are matched, that is $g_n[i] = h^*_{\gamma\mathcal{M}-n}[i] = h_{\gamma\mathcal{M}-n}[i]$, $0 \le i \le \mathcal{M}-1$. Hence the transfer functions of the receive filters are given by

$$G_j(z) = z^{-\gamma\mathcal{M}}\, H_j^*\left(\frac{1}{z^*}\right) = z^{-\gamma\mathcal{M}}\, H_j(z^{-1}) \qquad 0 \le j \le \mathcal{M}-1 \tag{9.96}$$

From (9.87) we get

$$G_j(z) = z^{-1}[\alpha_i^*\, U_j(z) + \alpha_i\, V_j(z)] \qquad 0 \le j \le \mathcal{M}-1 \tag{9.97}$$

Approximate interchannel interference suppression

From (9.11), we recall that to obtain a system without ICI at the output of the j-th subchannel it is necessary that the polyphase components with index 0 of the filters $G_j(z)\,H_i(z)$, $j \neq i$, are zero. These components will be denoted by $[G_j(z)\,H_i(z)]\downarrow_{\mathcal{M}}$. In practice, as the prototype filter $P_0(z)$ is a lowpass filter, an approximate suppression of ICI can be obtained in many cases under the condition that only the components $[G_j(z)\,H_{j\pm1}(z)]\downarrow_{\mathcal{M}}$ are cancelled, as shown in Figure 9.26 for $\mathcal{M}=4$, $j=1$ and $i=2$.

Taking, for example, the case $i=j+1$, from (1.580) we get

$$
\begin{aligned}
[G_j(z)H_{j+1}(z)]\downarrow_{\mathcal{M}} &= \frac{1}{\mathcal{M}}\sum_{\ell=0}^{\mathcal{M}-1} G_j\Big(z^{\frac{1}{\mathcal{M}}}\,W_{\mathcal{M}}^{\ell}\Big)H_{j+1}\Big(z^{\frac{1}{\mathcal{M}}}\,W_{\mathcal{M}}^{\ell}\Big)\\
&= \frac{1}{\mathcal{M}}\sum_{\ell=0}^{\mathcal{M}-1}\Big(z^{\frac{1}{\mathcal{M}}}\,W_{\mathcal{M}}^{\ell}\Big)^{-1}\Big(\alpha_j^{*}U_j(z^{\frac{1}{\mathcal{M}}}\,W_{\mathcal{M}}^{\ell})+\alpha_j V_j(z^{\frac{1}{\mathcal{M}}}\,W_{\mathcal{M}}^{\ell})\Big)\\
&\quad\times\Big(\alpha_{j+1}U_{j+1}\Big(z^{\frac{1}{\mathcal{M}}}\,W_{\mathcal{M}}^{\ell}\Big)+\alpha_{j+1}^{*}V_{j+1}\Big(z^{\frac{1}{\mathcal{M}}}\,W_{\mathcal{M}}^{\ell}\Big)\Big)
\end{aligned}
$$

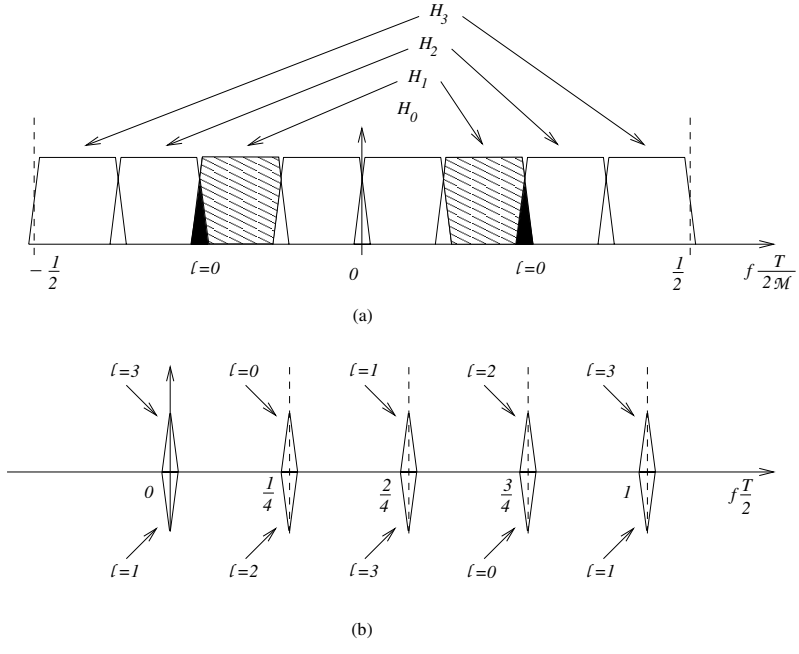


Figure 9.26. (a) Amplitude of the filter frequency responses for $\mathcal{M}=4$; (b) spectral components of ICI evaluated from the i-th input to the j-th output, for $j=1$ and $i=2$, after downsampling (see (9.98)).

$$\simeq \frac{1}{\mathcal{M}} \sum_{\ell=0}^{\mathcal{M}-1} \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right)^{-1} \alpha_j^* \alpha_{j+1} U_j \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right) U_{j+1} \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right)$$

$$+ \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right)^{-1} \alpha_j \alpha_{j+1}^* V_j \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right) V_{j+1} \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right) \quad (9.98)$$

where we have used the observation that in the case of ideal filters the functions U_{j_1} and V_{j_2} do not overlap in frequency. Therefore we assume that their product is negligible. From the definition (9.85) of the function $V_j(z)$ and from (9.82) we get

$$\begin{aligned} V_j \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell+j+1}\right) &= \beta_j^* \, Q_{j+1} \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right) \\ V_{j+1} \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell+j+1}\right) &= \beta_{j+1}^* \, Q_j \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right) \end{aligned} \quad (9.99)$$

Substituting the previous equations in (9.98), observing (9.85) and (9.86), and using the periodicity of $W_{\mathcal{M}}^{\ell}$, we obtain

$$[G_j(z) H_{j+1}(z)] \downarrow \mathcal{M} = \frac{1}{\mathcal{M}} \sum_{\ell=0}^{\mathcal{M}-1} \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right)^{-1} [\alpha_j^* \alpha_{j+1} \beta_j \beta_{j+1}$$

$$+ \alpha_j \alpha_{j+1}^* \beta_j^* \beta_{j+1}^* W_{\mathcal{M}}^{-(j+1)}] Q_j \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right) Q_{j+1} \left(z^{\frac{1}{\mathcal{M}}} W_{\mathcal{M}}^{\ell}\right) \quad (9.100)$$

The condition to suppress the ICI in the j-th subchannel due to the signal transmitted on the $(j+1)$-th subchannel is therefore given by

$$\alpha_j^* \alpha_{j+1} \beta_j \beta_{j+1} + \alpha_j \alpha_{j+1}^* \beta_j^* \beta_{j+1}^* W_{\mathcal{M}}^{-(j+1)} = 0 \quad (9.101)$$

Substitution of (9.93) in (9.101) yields the condition

$$\alpha_j^* \alpha_{j+1} = -\alpha_j \alpha_{j+1}^* \quad (9.102)$$

Analogously, for the suppression of the ICI in the j-th subchannel due to the signal transmitted on the $(j-1)$-th channel, the condition $\alpha_j^* \alpha_{j-1} = -\alpha_j \alpha_{j-1}^*$ is found. We note that, setting $\alpha_j = e^{j\varphi_j}$, we find that the condition for the approximate suppression of the ICI can be expressed as

$$\varphi_j = \varphi_{j-1} \pm \frac{\pi}{2} \quad (9.103)$$

Equation (9.103) sets a constraint on the sequence of the phases of the constants α_j, $0 \le j \le \mathcal{M}-1$; to define the whole sequence it is necessary to determine the phase of α_0. From (9.87) and (9.97), we observe that

$$G_j(z) \, H_j(z) = z^{-1} [U_j^2(z) + V_j^2(z) + (\alpha_j^2 + \alpha_j^{*2}) \, U_j(z) \, V_j(z)] \quad (9.104)$$

where the products $U_j(z)\,V_j(z)$ are negligible except for $j = 0$ and $j = \mathcal{M} - 1$. In these cases, to avoid that the function $\left|H_0\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right)\right|$ is distorted at frequencies near zero and that the function $\left|H_{\mathcal{M}-1}\left(e^{j\frac{2\pi fT}{2\mathcal{M}}}\right)\right|$ is distorted at frequencies near $\mathcal{M}/T$, it must be

$$\alpha_j^2 + \alpha_j^{*2} = 0 \qquad j = 0, \mathcal{M} - 1 \tag{9.105}$$

Therefore we choose

$$a_0^4 = a_{\mathcal{M}-1}^4 = -1 \tag{9.106}$$

For example, a sequence $\{\varphi_j\}_{j=0}^{\mathcal{M}-1}$ that satisfies both the (9.103) and the further condition (9.106) is given by

$$\varphi_j = (-1)^j\,\frac{\pi}{4} \qquad j = 0, 1, \ldots, \mathcal{M} - 1 \tag{9.107}$$

Moreover, from (9.104), the condition for the absence of ISI is expressed as

$$[z^{-1}(U_j^2(z) + V_j^2(z))] \downarrow_{\mathcal{M}} = \text{constant} \qquad j = 0, 1, \ldots, \mathcal{M} - 1 \tag{9.108}$$

Summarizing, for the design of the system we may start from a prototype filter $P_0(z)$ that approximates a square root raised cosine filter with Nyquist frequency $1/(2T)$. This leads to verifying the condition (9.108). The $\mathcal{M}$ subchannel filters are obtained using (9.84), (9.85), and (9.87), where β_i is defined in (9.93) and the phase of α_i is given in (9.107).

An efficient implementation of the transmit filter bank is illustrated in Figure 9.27, where $P^{(i)}(z)$ are the $2\mathcal{M}$ polyphase components of the prototype filter $P_0(z)$, and $d_i = \alpha_i\beta_i$, $i = 0, 1, \ldots, \mathcal{M} - 1$. An efficient implementation of the receive filter bank is illustrated in Figure 9.28.

Perfect interchannel interference suppression

We derive now the conditions on the FIR filters of a DWMT system for the absence of intersymbol as well as interchannel interference in the case of an ideal transmission channel, i.e. the orthogonality conditions. We consider the system of Figure 9.27. The relation between the vector of real-valued input symbols $\mathbf{a}_k^T = [a_k[0], \ldots, a_k[\mathcal{M} - 1]]$ and the vector of real-valued samples $\mathbf{A}_k^T = [A_k[0], \ldots, A_k[2\mathcal{M} - 1]]$ can be expressed in terms of the matrix

$$\mathbf{T}^T = \text{diag}\left\{1, W_{2\mathcal{M}}^{-\frac{1}{2}}, \ldots, W_{2\mathcal{M}}^{-\frac{2\mathcal{M}-1}{2}}\right\}\mathbf{F}_{2\mathcal{M}}^{-1}\,\text{diag}\{d_0, d_1, \ldots, d_1^*, d_0^*\}\begin{pmatrix}\mathbf{I}\\ \mathbf{J}_{\mathcal{M}}\end{pmatrix} \tag{9.109}$$

where

$$d_i = \alpha_i\beta_i = e^{j(-1)^i\frac{\pi}{4}}\,W_{2\mathcal{M}}^{\left(i+\frac{1}{2}\right)\frac{\gamma\mathcal{M}-1}{2}} \qquad i = 0, \ldots, \mathcal{M} - 1 \tag{9.110}$$

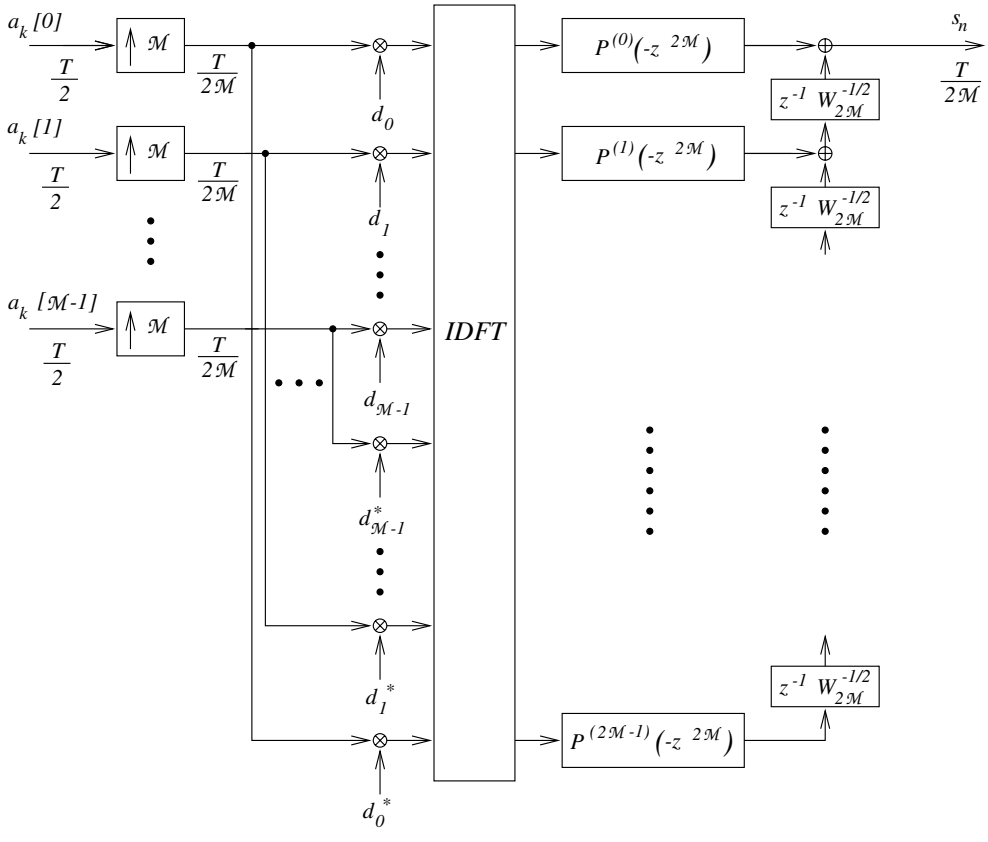


Figure 9.27. OFDM system with approximate suppression of the ICI: transmit filter bank.

while $\mathbf{J}_{\mathcal{M}}$ denotes the $\mathcal{M} \times \mathcal{M}$ matrix that has elements equal to one on the antidiagonal and all other elements equal to zero:

$$\mathbf{J}_{\mathcal{M}} = \begin{bmatrix} 0 & \cdots & 0 & 0 & 1 \\ 0 & \cdots & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \tag{9.111}$$

We assume that the parameters γ and $\mathcal{M}$ that determine the length $\gamma\mathcal{M}$ of the prototype filter are even numbers. The element t_{in} of matrix $\mathbf{T}$ is then given by

$$\begin{aligned} t_{in} &= W_{2\mathcal{M}}^{-\frac{n}{2}}\, W_{2\mathcal{M}}^{-ni}\, d_i + W_{2\mathcal{M}}^{-\frac{n}{2}}\, W_{2\mathcal{M}}^{-n(2\mathcal{M}-1-i)}\, d_i^* \\ &= 2\cos\left[\frac{\pi}{\mathcal{M}}\left(i+\frac{1}{2}\right)\left(n-\frac{\gamma\mathcal{M}-1}{2}\right)+(-1)^i\,\frac{\pi}{4}\right] \\ &= 2(\hat{c}_{in}\cos\phi_i - \hat{s}_{in}\sin\phi_i) \qquad 0 \le i \le \mathcal{M}-1 \quad 0 \le n \le 2\mathcal{M}-1 \end{aligned} \tag{9.112}$$

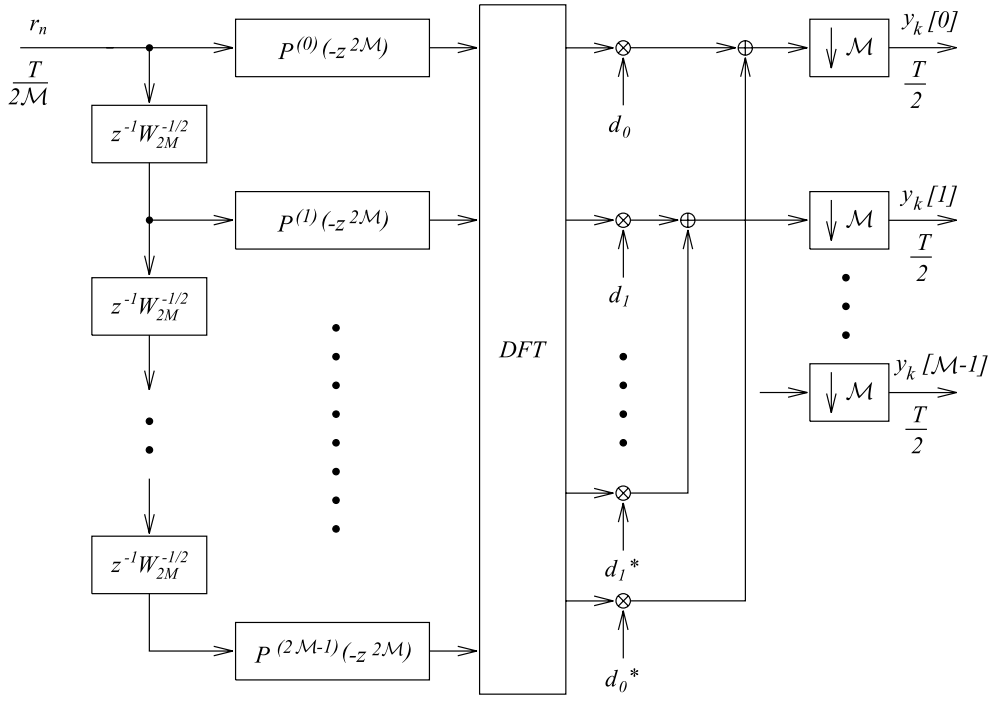


Figure 9.28. OFDM system with approximate suppression of the ICI: receive filter bank.

where

$$\begin{aligned}
\hat{c}_{in} &= \cos\left(\frac{\pi}{\mathcal{M}}\left(i+\frac{1}{2}\right)\left(n+\frac{1}{2}\right)\right) \\
\hat{s}_{in} &= \sin\left(\frac{\pi}{\mathcal{M}}\left(i+\frac{1}{2}\right)\left(n+\frac{1}{2}\right)\right) \\
\phi_i &= -\pi\left(i+\frac{1}{2}\right)\frac{\gamma}{2}+(-1)^i\frac{\pi}{4}
\end{aligned} \tag{9.113}$$

Therefore using the matrix $\mathbf{T}^T$ we obtain an equivalent block diagram to that of Figure 9.27, as illustrated in Figure 9.29.

We give the following definitions:

1. $\mathbf{C}$ is the matrix of the $\mathcal{M}$-point discrete cosine transform (DCT), whose element in the position i, n is given by

$$[\mathbf{C}]_{i,n} = \left[\sqrt{\frac{2}{\mathcal{M}}}\cos\left(\frac{\pi}{\mathcal{M}}\left(i+\frac{1}{2}\right)\left(n+\frac{1}{2}\right)\right)\right] \tag{9.114}$$

$$i = 0, \ldots, \mathcal{M}-1 \quad n = 0, \ldots, \mathcal{M}-1$$

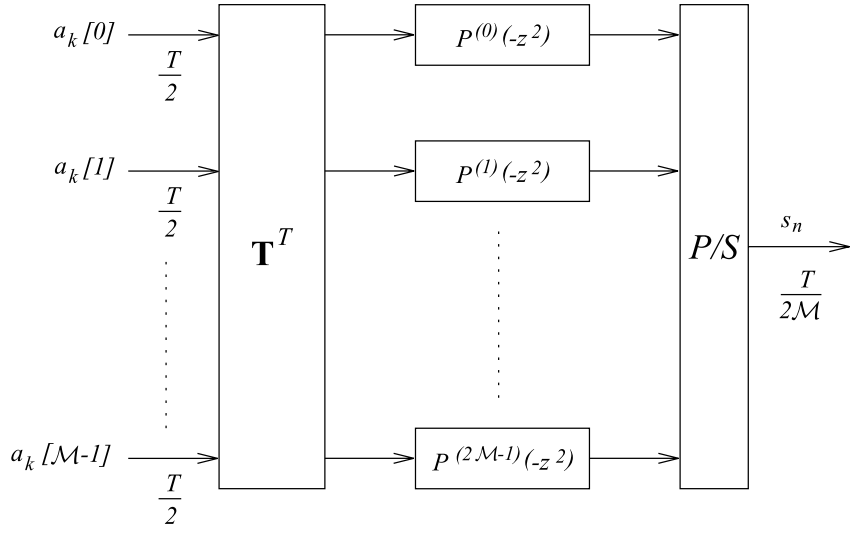


Figure 9.29. DWMT system implemented by the matrix **T**.

2. **S** is the matrix of the $\mathcal{M}$-point discrete sine transform (DST), whose element in the position i, n is given by

$$[\mathbf{S}]_{i,n} = \left[\sqrt{\frac{2}{\mathcal{M}}} \sin\left(\frac{\pi}{\mathcal{M}}\left(i+\frac{1}{2}\right)\left(n+\frac{1}{2}\right)\right)\right] \tag{9.115}$$

3. $\mathbf{\Lambda}_c$ and $\mathbf{\Lambda}_s$ are diagonal matrices given by

$$[\mathbf{\Lambda}_c]_{ii} = \cos\left(\pi\left(i+\frac{1}{2}\right)\frac{\gamma}{2}\right) \tag{9.116}$$

and

$$[\mathbf{\Lambda}_s]_{ii} = \sin\left(\pi\left(i+\frac{1}{2}\right)\frac{\gamma}{2}\right) \tag{9.117}$$

respectively;

4.

$$\mathbf{\Gamma}_{\mathcal{M}} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & (-1)^{\mathcal{M}-1} \end{bmatrix} \tag{9.118}$$

5.

$$\mathbf{T} = [\mathbf{A}_0, \mathbf{A}_1] \tag{9.119}$$

6. $\mathbf{A}_0$ and $\mathbf{A}_1$ are $\mathcal{M} \times \mathcal{M}$ matrices given by

$$\begin{aligned} &\mathbf{A}_0 = \sqrt{\mathcal{M}}\,\mathbf{\Lambda}_c(\mathbf{C} - \mathbf{\Gamma}_{\mathcal{M}}\,\mathbf{S}) \qquad \mathbf{A}_1 = -\sqrt{\mathcal{M}}\,\mathbf{\Lambda}_c(\mathbf{C} + \mathbf{\Gamma}_{\mathcal{M}}\,\mathbf{S}) \qquad \frac{\gamma}{2} \text{ even} \\ &\mathbf{A}_0 = \sqrt{\mathcal{M}}\,\mathbf{\Lambda}_c(\mathbf{C} + \mathbf{\Gamma}_{\mathcal{M}}\,\mathbf{S}) \qquad \mathbf{A}_1 = \sqrt{\mathcal{M}}\,\mathbf{\Lambda}_c(\mathbf{C} - \mathbf{\Gamma}_{\mathcal{M}}\,\mathbf{S}) \qquad \frac{\gamma}{2} \text{ odd} \end{aligned} \tag{9.120}$$

We consider the polyphase representation of the prototype filter $P_0(z)$ by the $2\mathcal{M}$ polyphase components $P^{(\ell)}(z)$, $\ell = 0, 1, \ldots, 2\mathcal{M}-1$, each having $\gamma/2$ coefficients, and define

$$\begin{aligned} \mathbf{p}_0(z) &= \text{diag}\{P^{(0)}(-z), \ldots, P^{(\mathcal{M}-1)}(-z)\} \\ \mathbf{p}_1(z) &= \text{diag}\{P^{(\mathcal{M})}(-z), \ldots, P^{(2\mathcal{M}-1)}(-z)\} \end{aligned} \tag{9.121}$$

Using (9.119) and (9.120), the vector of the transfer functions of the transmit filters $\mathbf{h}^T(z)$ can be expressed as

$$\mathbf{h}^T(z) = z^{-(2\mathcal{M}-1)} \left[\mathbf{e}^H\left(\frac{1}{z^*}\right),\ z^{\mathcal{M}}\,\mathbf{e}^H\left(\frac{1}{z^*}\right) \right] \begin{bmatrix} \mathbf{p}_0(z^{2\mathcal{M}}) & \mathbf{0}_{\mathcal{M}} \\ \mathbf{0}_{\mathcal{M}} & \mathbf{p}_1(z^{2\mathcal{M}}) \end{bmatrix} \begin{bmatrix} \mathbf{A}_0^T \\ \mathbf{A}_1^T \end{bmatrix} \tag{9.122}$$

Comparing (9.122) with (9.17), we find that the matrix of the transfer functions of the polyphase components of the transmit filters is given by

$$\mathbf{E}(z) = [z^{-1}\,\mathbf{p}_0(z^2), \mathbf{p}_1(z^2)] \begin{bmatrix} \mathbf{A}_0^T \\ \mathbf{A}_1^T \end{bmatrix} \tag{9.123}$$

We recall from (9.24) that we get a system without ISI and ICI if $\mathbf{E}^H(1/z^*)\mathbf{E}(z) = \mathbf{I}$. We consider the product

$$\begin{aligned} \mathbf{E}(z)\,\mathbf{E}^H\left(\frac{1}{z^*}\right) &= [z^{-1}\,\mathbf{p}_0(z^2), \mathbf{p}_1(z^2)] \begin{bmatrix} \mathbf{A}_0^T \\ \mathbf{A}_1^T \end{bmatrix} [\mathbf{A}_0, \mathbf{A}_1] \begin{bmatrix} z\,\mathbf{p}_0^*\left(\frac{1}{z^{*2}}\right) \\ \mathbf{p}_1^*\left(\frac{1}{z^{*2}}\right) \end{bmatrix} \\ &= \mathbf{p}_0(z^2)\mathbf{A}_0^T\mathbf{A}_0\mathbf{p}_0^*\left(\frac{1}{z^{*2}}\right) + z^{-1}\mathbf{p}_0(z^2)\mathbf{A}_0^T\mathbf{A}_1\mathbf{p}_1^*\left(\frac{1}{z^{*2}}\right) \\ &\quad \times z\mathbf{p}_1(z^2)\mathbf{A}_1^T\mathbf{A}_0\mathbf{p}_0^*\left(\frac{1}{z^{*2}}\right) + \mathbf{p}_1(z^2)\mathbf{A}_1^T\mathbf{A}_1\mathbf{p}_1^*\left(\frac{1}{z^{*2}}\right) \end{aligned} \tag{9.124}$$

Using definition (9.120) of matrices $\mathbf{A}_0$ and $\mathbf{A}_1$ we obtain

$$\begin{aligned} \mathbf{E}(z)\,\mathbf{E}^H\left(\frac{1}{z^*}\right) &= 2\mathcal{M}\left(\mathbf{p}_0(z^2)\,\mathbf{p}_0^*\left(\frac{1}{z^{*2}}\right) + \mathbf{p}_1(z^2)\,\mathbf{p}_1^*\left(\frac{1}{z^{*2}}\right)\right) \\ &\quad - 2\mathcal{M}(-1)^{\frac{\gamma}{2}}\left(\mathbf{p}_0(z^2)\,\mathbf{J}_{\mathcal{M}}\mathbf{p}_0^*\left(\frac{1}{z^{*2}}\right) - \mathbf{p}_1(z^2)\mathbf{J}_{\mathcal{M}}\mathbf{p}_1^*\left(\frac{1}{z^{*2}}\right)\right) \end{aligned} \tag{9.125}$$

We recall that each polyphase component $P^{(\ell)}(z)$ has length $\gamma/2$. Moreover, from (9.90), the relation $p_n[0] = p_{\gamma\mathcal{M}-1-n}[0]$ implies the following constraints on the polyphase components:

$$P^{(\ell)}(z) = z^{-\left(\frac{\gamma}{2}-1\right)}\left[P^{(2\mathcal{M}-1-\ell)}\left(\frac{1}{z^*}\right)\right]^* \qquad \ell = 0, \ldots, 2\mathcal{M}-1 \tag{9.126}$$

From property (9.126), we find that the diagonal matrices $\mathbf{p}_0(z)$ and $\mathbf{p}_1(z)$ satisfy the relation

$$\mathbf{p}_1(z) = z^{-\left(\frac{\gamma}{2}-1\right)}(-1)^{\frac{\gamma}{2}-1}\,\mathbf{J}_{\mathcal{M}}\,\mathbf{p}_0^*\left(\frac{1}{z^*}\right)\mathbf{J}_{\mathcal{M}} \tag{9.127}$$

Then, using (9.127), we find that the second term in the (9.125) vanishes. Therefore we obtain

$$\mathbf{E}(z)\,\mathbf{E}^H\left(\frac{1}{z^*}\right) = 2\mathcal{M}\left(\mathbf{p}_0(z^2)\,\mathbf{p}_0^*\left(\frac{1}{z^{*2}}\right) + \mathbf{p}_1(z^2)\,\mathbf{p}_1^*\left(\frac{1}{z^{*2}}\right)\right) \tag{9.128}$$

Recalling that for two matrices whose product is the identity matrix the commutative property holds, we get $\mathbf{E}^H(1/z^*)\,\mathbf{E}(z) = \mathbf{E}(z)\,\mathbf{E}^H(1/z^*) = \mathbf{I}$ if and only if

$$\mathbf{p}_0(z)\,\mathbf{p}_0^*\left(\frac{1}{z^*}\right) + \mathbf{p}_1(z)\,\mathbf{p}_1^*\left(\frac{1}{z^*}\right) = \frac{1}{2\mathcal{M}}\,\mathbf{I} \tag{9.129}$$

Using (9.129), we find the conditions on the polyphase components of the prototype filter for perfect suppression of ISI and ICI, given by

$$\left[P^{(\ell)}\left(\frac{1}{z^*}\right)\right]^* P^{(\ell)}(z) + \left[P^{(\mathcal{M}+\ell)}\left(\frac{1}{z^*}\right)\right]^* P^{(\mathcal{M}+\ell)}(z) = \frac{1}{2\mathcal{M}} \qquad 0 \le \ell \le \mathcal{M}-1 \tag{9.130}$$

The conditions (9.130) can be used for the design of filters for DWMT systems. An efficient filter bank implementation is obtained by the DCT [1].

Bibliography

[1] P. P. Vaidyanathan, *Multirate systems and filter banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[2] M. G. Bellanger, G. Bonnerot, and M. Coudreuse, "Digital filtering by polyphase network: application to sample-rate alteration and filter banks", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. ASSP-24, pp. 109–114, Apr. 1976.

[3] G. Cherubini, E. Eleftheriou, S. Ölçer, and J. M. Cioffi, "Filter bank modulation techniques for very high-speed digital subscriber lines", *IEEE Communications Magazine*, vol. 38, pp. 98–104, May 2000.

[4] G. Cherubini, E. Eleftheriou, and S. Ölcer, "Filtered multitone modulation for very-high-speed digital subscriber lines", *IEEE Journal on Selected Areas in Communications*, June 2002.

[5] S. B. Weinstein and P. M. Ebert, "Data transmission by frequency-division multiplexing using the discrete Fourier transform", *IEEE Trans. on Communications*, vol. 19, pp. 628–634, Oct. 1971.

[6] J. A. C. Bingham, "Multicarrier modulation for data transmission: an idea whose time has come", *IEEE Communications Magazine*, vol. 28, pp. 5–14, May 1990.

[7] H. Sari, G. Karam, and I. Jeanclaude, "Transmission techniques for digital terrestrial TV broadcasting", *IEEE Communications Magazine*, vol. 33, pp. 100–109, Feb. 1995.

[8] J. S. Chow, J. C. Tu, and J. M. Cioffi, "A discrete multitone transceiver system for HDSL applications", *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 895–908, Aug. 1991.

[9] S. D. Sandberg and M. A. Tzannes, "Overlapped discrete multitone modulation for high speed copper wire communications", *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 1571–1585, Dec. 1995.

[10] P. J. W. Melsa, R. C. Younce, and C. E. Rohrs, "Impulse response shortening for discrete multitone transceivers", *IEEE Trans. on Communications*, vol. 44, pp. 1662–1672, Dec. 1996.

[11] R. Baldemair and P. Frenger, "A time-domain equalizer minimizing intersymbol and intercarrier interference in DMT systems", in *Proc. GLOBECOM '01*, San Antonio, TX, Nov. 2001.

[12] T. Pollet and M. Peeters, "Synchronization with DMT modulation", *IEEE Communications Magazine*, vol. 37, pp. 80–86, Apr. 1999.

[13] J. M. Cioffi, G. P. Dudevoir, M. V. Eyuboglu, and G. D. Forney, Jr., "MMSE decision-feedback equalizers and coding. Part I and Part II", *IEEE Trans. on Communications*, vol. 43, pp. 2582–2604, Oct. 1995.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 10

Spread spectrum systems

The term spread spectrum systems [1, 2, 3, 4, 5, 6, 7] was coined to indicate communication systems in which the bandwidth of the signal obtained by a standard modulation method (see Chapter 6) is spread by a certain factor before transmission over the channel, and then despread, by the same factor, at the receiver. The operations of spreading and despreading are the inverse of each other, i.e. for an ideal and noiseless channel the received signal after despreading is equivalent to the transmitted signal before spreading. For transmission over an ideal AWGN channel these operations do not offer therefore any improvement in performance with respect to a system that does not use spread spectrum. However, the practical applications of spread spectrum systems are numerous, for example, in multiple-access systems, narrowband interference rejection, and transmission over channels with fading (see Section 10.2).

10.1 Spread spectrum techniques

We consider the two most common spread spectrum techniques: *direct sequence* (DS) and *frequency hopping* (FH).

10.1.1 Direct sequence systems

The baseband equivalent model of a DS system is illustrated in Figure 10.1. We consider the possibility that U users in a multiple-access system simultaneously transmit, using the same frequency band, by *code division multiple access* (CDMA) (see Section 6.13.2).

The sequence of information bits $\{b_\ell^{(u)}\}$ of user u undergoes the following transformations.

1) *Bit-mapper.* From the sequence of information bits, a sequence of i.i.d. symbols $\{a_k^{(u)}\}$ with statistical power $\mathtt{M}_a$ is produced. The symbols assume values in an M-ary constellation, using one of the maps described in Chapter 6; typically, in mobile radio systems, BPSK or QPSK modulation is used. Let T be the symbol period.

2) *Spreading.* We indicate by the integer N_{SF} the *spreading factor*, and by T_{chip} the *chip period*. These two parameters are related to the symbol period T by the relation

$$T_{chip} = \frac{T}{N_{SF}} \tag{10.1}$$

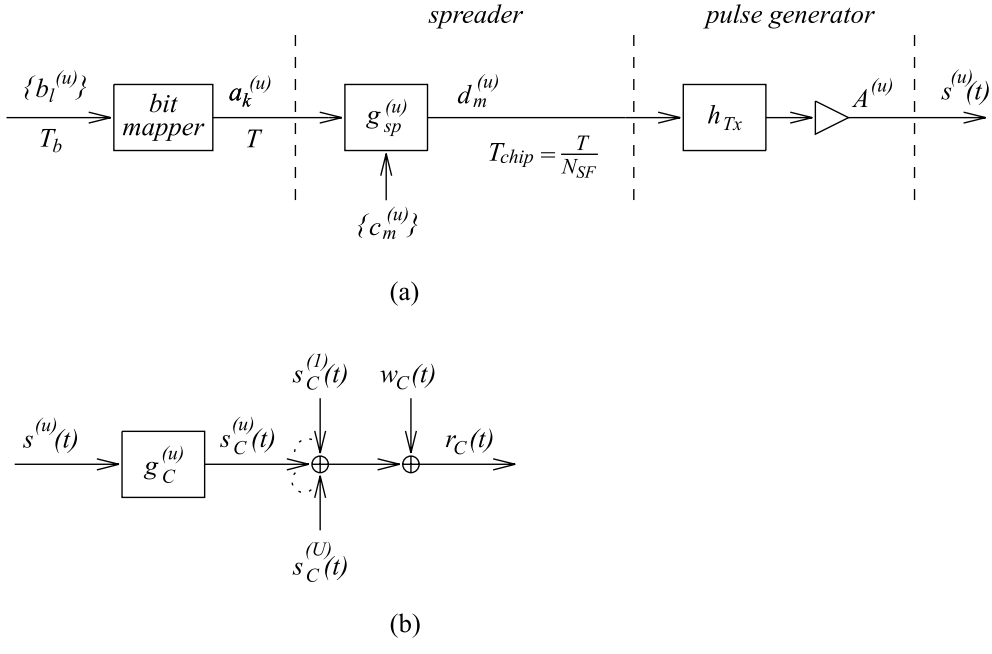


Figure 10.1. Baseband equivalent model of a DS system: (a) transmitter, (b) multiuser channel.

We recall from Appendix 6.D the definition of the Walsh–Hadamard sequences of length N_{SF}. Here we refer to these sequences as channelization code $\{c_{Ch,m}^{(u)}\}$, $m = 0, 1, \ldots, N_{SF} - 1$, $u \in \{1, \ldots, U\}$. Moreover, $c_{Ch,m}^{(u)} \in \{-1, 1\}$, and $|c_{Ch,m}^{(u)}| = 1$. The Walsh–Hadamard sequences are orthogonal, that is

$$\frac{1}{N_{SF}} \sum_{m=0}^{N_{SF}-1} c_{Ch,m}^{(u_1)} \, c_{Ch,m}^{(u_2)*} = \begin{cases} 1 & \text{if } u_1 = u_2 \\ 0 & \text{if } u_1 \neq u_2 \end{cases} \tag{10.2}$$

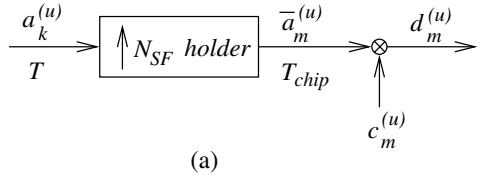
We assume that the sequences of the channelization code are periodic, that is $c_{Ch,m}^{(u)} = c_{Ch,m_{mod\, N_{SF}}}^{(u)}$.

We now introduce the *user code* $\{c_m^{(u)}\}$, also called *signature sequence* or *spreading sequence*, that we initially assume to be equal to the channelization code,

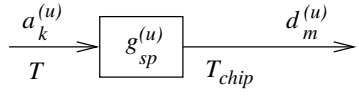
$$c_m^{(u)} = c_{Ch,m}^{(u)} \tag{10.3}$$

Consequently, $\{c_m^{(u)}\}$ is also a periodic sequence of period N_{SF}.

The operation of spreading consists of associating with each symbol $a_k^{(u)}$ a sequence of N_{SF} symbols of period T_{chip}, that is obtained as follows. First, each symbol $a_k^{(u)}$ is repeated N_{SF} times with period T_{chip}: as illustrated in Figure 10.2, this operation is equivalent to upsampling $\{a_k^{(u)}\}$, so that $(N_{SF} - 1)$ zeros are inserted between two consecutive symbols,



(a)



(b)

Figure 10.2. Spreading operation: (a) correlator, (b) interpolator filter.

and using a *holder* of N_{SF} values. The obtained sequence is then multiplied by the user code. Formally we have

$$\begin{aligned} \bar{a}_m^{(u)} &= a_k^{(u)} \qquad\qquad m = kN_{SF}, \ldots, kN_{SF} + N_{SF} - 1 \\ d_m^{(u)} &= \bar{a}_m^{(u)}\, c_m^{(u)} \end{aligned} \tag{10.4}$$

If we introduce the filter

$$g_{sp}^{(u)}(i\, T_{chip}) = c_i^{(u)} \qquad i = 0, \ldots, N_{SF} - 1 \tag{10.5}$$

the correlation of Figure 10.2a can be substituted by an interpolation with the interpolator filter $g_{sp}^{(u)}$, as illustrated in Figure 10.2b.

Recalling that $|c_m^{(u)}| = 1$, from (10.4) we get

$$\mathtt{M}_d = \mathtt{M}_a \tag{10.6}$$

3) *Pulse-shaping.* Let h_{Tx} be the modulation pulse, typically a square root raised cosine function or rectangular window. The baseband equivalent of the transmitted signal of user u is expressed as

$$s^{(u)}(t) = A^{(u)} \sum_{m=-\infty}^{+\infty} d_m^{(u)}\, h_{Tx}(t - m\, T_{chip}) \tag{10.7}$$

where $A^{(u)}$ accounts for the transmit signal power. In fact, if E_h is the energy of h_{Tx} and $\{d_m^{(u)}\}$ is assumed i.i.d., the average statistical power of $s^{(u)}(t)$ is given by (see (1.399))

$$\bar{\mathtt{M}}_{s^{(u)}} = (A^{(u)})^2\, \mathtt{M}_d\, \frac{E_h}{T_{chip}} \tag{10.8}$$

Using (10.4), an alternative expression for (10.7) is given by

$$s^{(u)}(t) = A^{(u)} \sum_{k=-\infty}^{+\infty} a_k^{(u)} \sum_{\ell=0}^{N_{SF}-1} c_{\ell+kN_{SF}}^{(u)} \, h_{Tx}(t - (\ell + kN_{SF})\, T_{chip}) \tag{10.9}$$

In the scheme of Figure 10.1a we note that, if condition (10.3) holds, then $g_{sp}^{(u)}$ is invariant with respect to the symbol period, and the two filters $g_{sp}^{(u)}$ and h_{Tx} can be combined into one filter (see also (10.9))

$$h_T^{(u)}(t) = \sum_{\ell=0}^{N_{SF}-1} c_\ell^{(u)} \, h_{Tx}(t - \ell \, T_{chip}) \tag{10.10}$$

and

$$s^{(u)}(t) = A^{(u)} \sum_{k=-\infty}^{+\infty} a_k^{(u)} \, h_T^{(u)}(t - kT) \tag{10.11}$$

As shown in Figure 10.3, the equivalent scheme to the cascade of spreader and pulse-shaping filter is still a QAM modulator. The peculiarity is that the filter $h_T^{(u)}$ has a bandwidth much larger than the Nyquist frequency $1/(2T)$. Therefore a DS system can be interpreted as a QAM system either with input symbols $\{d_m^{(u)}\}$ and transmit pulse h_{Tx}, or with input symbols $\{a_k^{(u)}\}$ and pulse $h_T^{(u)}$; later both interpretations will be used.

4) *Transmission channel.* Modeling the transmission channel as a filter having impulse response $g_C^{(u)}$, the output signal is given by

$$s_C^{(u)}(t) = (s^{(u)} * g_C^{(u)})(t) \tag{10.12}$$

The possibility that many users transmit simultaneously over the same frequency band leads to a total signal

$$s_C(t) = \sum_{u=1}^{U} s_C^{(u)}(t) \tag{10.13}$$

We assume we are interested in reconstructing the message $\{a_k^{(1)}\}$ of user $u = 1$, identified as *desired user*. If $\mathtt{M}_{s_C^{(u)}}$ is the statistical power of $s_C^{(u)}$, the following *signal-to-interference ratios* (SIRs) define the relative powers of the user signals:

$$\Gamma_i^{(u)} = \frac{\mathtt{M}_{s_C^{(1)}}}{\mathtt{M}_{s_C^{(u)}}} \qquad u = 2, 3, \ldots, U \tag{10.14}$$

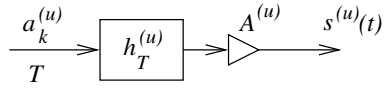


Figure 10.3. Equivalent scheme of spreader and pulse-shaping filter in a DS system.

5) *Noise.* In Figure 10.1b the term w_C includes both the noise of the receiver and possible additional interference, such as the interference due to signals coming from *other cells* in a wireless system; w_C is modeled as white noise with PSD equal to N_0.

Two signal-to-noise ratios are of interest. To measure the performance of the system in terms of P_{bit}, it is convenient to refer to the signal-to-noise ratio defined in Chapter 6 for passband transmission,

$$\Gamma_s = \frac{\mathtt{M}_{s_C^{(1)}}}{N_0/T} \tag{10.15}$$

We recall that for an uncoded sequence of symbols $\{a_k^{(1)}\}$, the following relation holds:

$$\frac{E_b}{N_0} = \frac{\Gamma_s}{\log_2 M} \tag{10.16}$$

However, there are cases, as for example in the evaluation of the performance of the channel impulse response estimation algorithm, when it is useful to measure the power of the noise over the whole transmission bandwidth. Hence we define the ratio

$$\Gamma_c = \frac{\mathtt{M}_{s_C^{(1)}}}{N_0/T_{chip}} = \frac{\Gamma_s}{N_{SF}} \tag{10.17}$$

6) *Receiver.* The receiver structure varies according to the channel model and number of users. Deferring until Section 10.3 the analysis of more complicated system configurations, here we limit ourselves to considering the case of an ideal AWGN channel with $g_C^{(u)}(t) = \delta(t)$ and *synchronous users*. The latter assumption implies that the transmitters of the various users are synchronized and transmit at the same instant. For an ideal AWGN channel this means that at the receiver the optimum timing phase of signals of different users is the same.

With these assumptions, we verify that the optimum receiver is simply given by the matched filter to $h_T^{(1)}(t)$. According to the analog or discrete-time implementation of the matched filters, we get the schemes of Figure 10.4 or Figure 10.5, respectively; note that in Figure 10.5 the receiver front-end comprises an anti-aliasing filter followed by a sampler with sampling period $T_c = T_{chip}/2$. Let t_0 be the optimum timing phase at the matched filter output (see Section 14.7).

For an ideal AWGN channel it results in

$$\begin{aligned} r_C(t) &= \sum_{u=1}^{U} s^{(u)}(t) + w_C(t) \\ &= \sum_{u=1}^{U} A^{(u)} \sum_{i=-\infty}^{+\infty} a_i^{(u)} \sum_{\ell=0}^{N_{SF}-1} c_\ell^{(u)}\, h_{Tx}(t - (\ell + iN_{SF})\, T_{chip}) + w_C(t) \end{aligned} \tag{10.18}$$

In the presence only of the desired user, that is for $U = 1$, it is clear that in the absence of ISI the structure with the matched filter to $h_T^{(1)}(t)$ is optimum. We verify that the presence of other users is cancelled at the receiver, given that the various user codes are orthogonal.

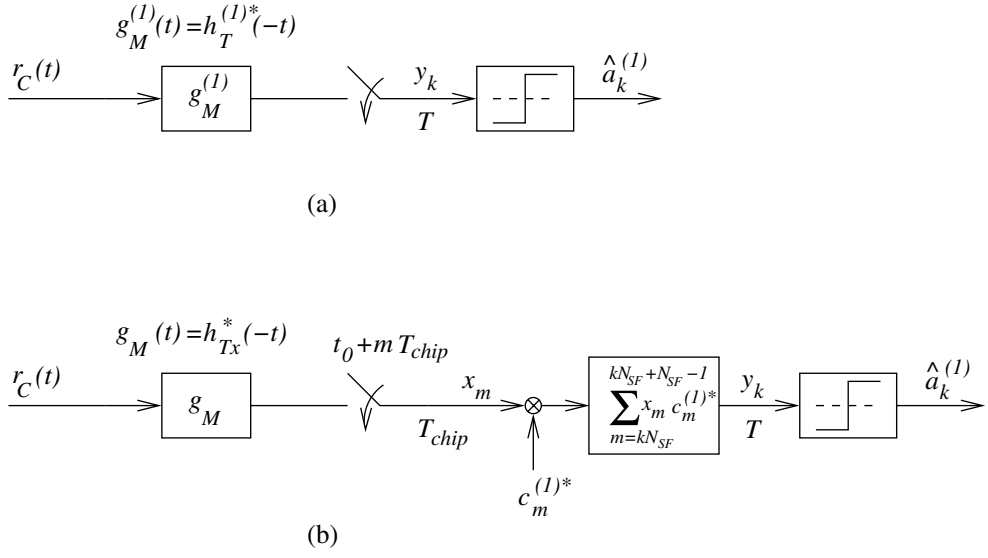


Figure 10.4. Optimum receiver with analog filters for a DS-CDMA system with ideal AWGN channel and synchronous users. Two equivalent structures: (a) overall matched filter, (b) matched filter to h_{Tx} and despreading correlator.

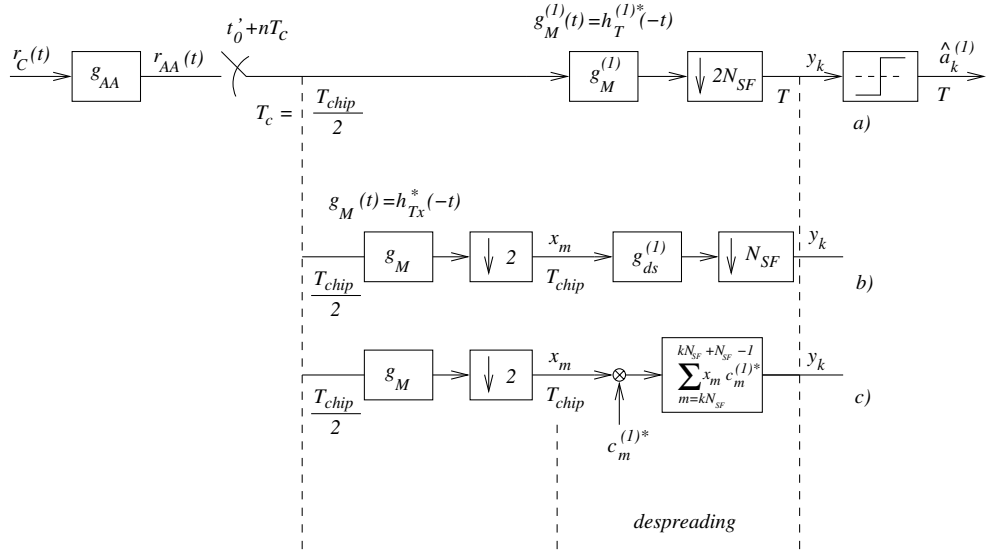


Figure 10.5. Optimum receiver with discrete-time filters for a DS-CDMA system with ideal AWGN channel and synchronous users. Three equivalent structures: (a) overall matched filter, (b) matched filter of h_{Tx} and despreading filter, (c) matched filter to h_{Tx} and despreading correlator.

We assume that the overall analog impulse response of the system is a Nyquist pulse, hence

$$(h_{Tx} * g_C^{(u)} * g_{AA} * g_M)(t)|_{t=t_0+j\,T_{chip}} = (h_{Tx} * g_M)(t)|_{t=t_0+j\,T_{chip}} = E_h\,\delta_j \tag{10.19}$$

We note that, if t_0 is the instant at which the peak of the overall pulse at the output of g_M is observed, then t_0' in Figure 10.5 is given by $t_0' = t_0 - t_{g_M}$, where t_{g_M} is the duration of g_M. Moreover, from (10.19), we get that the noise at the output of g_M, sampled with sampling rate $1/T_{chip}$,

$$\tilde{w}_m = (w_C * g_M)(t)|_{t=t_0+m\,T_{chip}} \tag{10.20}$$

is an i.i.d. sequence with variance $N_0 E_h$.

Hence, from (10.9) and (10.19), the signal at the output of g_M, sampled with sampling rate $1/T_{chip}$, has the following expression:

$$x_m = E_h \sum_{u=1}^{U} A^{(u)} \sum_{i=-\infty}^{+\infty} a_i^{(u)} \sum_{\ell=0}^{N_{SF}-1} c_{\ell+iN_{SF}}^{(u)}\,\delta_{m-\ell-iN_{SF}} + \tilde{w}_m \tag{10.21}$$

With the change of indices $m = j + kN_{SF}$, $j = 0, 1, \ldots, N_{SF}-1$, k integer, we get

$$x_{j+kN_{SF}} = E_h \sum_{u=1}^{U} A^{(u)}\, a_k^{(u)}\, c_{j+kN_{SF}}^{(u)} + \tilde{w}_{j+kN_{SF}} \tag{10.22}$$

7) *Despreading*. We now correlate the sequence of samples $\{x_m\}$, suitably synchronized, with the code sequence of the desired user, and we form the signal

$$y_k = \sum_{j=0}^{N_{SF}-1} x_{j+kN_{SF}}\, c_{j+kN_{SF}}^{(1)*} \tag{10.23}$$

As usual, introducing the filter $g_{ds}^{(u)}$ given by

$$g_{ds}^{(u)}(i\,T_{chip}) = c_{N_{SF}-1-i}^{(1)*} \qquad i = 0, 1, \ldots, N_{SF}-1 \tag{10.24}$$

the correlation (10.23) is implemented through the filter (10.24), followed by a downsampler, as illustrated in Figure 10.5b.

Substitution of (10.22) in (10.23) yields

$$y_k = N_{SF}\,E_h \sum_{u=1}^{U} A^{(u)}\, a_k^{(u)}\, \mathfrak{r}_{c^{(u)}\,c^{(1)}}(0) + w_k \tag{10.25}$$

where in general

$$\mathfrak{r}_{c^{(u_1)}\,c^{(u_2)}}(n_D) = \frac{1}{N_{SF} - |n_D|} \sum_{j=0}^{N_{SF}-1-|n_D|} c_{j+kN_{SF}+n_D}^{(u_1)}\, c_{j+kN_{SF}}^{(u_2)*} \tag{10.26}$$

$$n_D = -(N_{SF}-1), \ldots, -1, 0, 1, \ldots, N_{SF}-1$$

is the cross-correlation at lag n_D between the user codes u_1 and u_2.

In the considered case, from (10.3) and (10.2) we get $\mathtt{r}_{c^{(u_1)}\,c^{(u_2)}}(0) = \delta_{u_1-u_2}$. Therefore (10.25) simply becomes

$$y_k = N_{SF}\, E_h\, A^{(1)}\, a_k^{(1)} + w_k \tag{10.27}$$

where $N_{SF}\, E_h$ is the energy of the pulse associated with $a_k^{(1)}$ (see (10.10)).

In (10.27) the noise is given by

$$w_k = \sum_{j=0}^{N_{SF}-1} \tilde{w}_{j+kN_{SF}}\, c_{j+kN_{SF}}^{(1)*} \tag{10.28}$$

therefore assuming $\{\tilde{w}_m\}$ i.i.d., the variance of w_k is given by

$$\sigma_w^2 = N_{SF}\, \sigma_{\tilde{w}}^2 = N_{SF}\, N_0\, E_h \tag{10.29}$$

8) *Data detector.* Using a threshold detector, from (10.27) the signal-to-noise ratio at the decision point is given by (see (7.106))

$$\gamma = \left(\frac{d_{min}}{2\sigma_I}\right)^2 = \frac{(N_{SF}\, E_h\, A^{(1)})^2}{N_{SF}\, N_0\, E_h/2} = \frac{N_{SF}\, E_h (A^{(1)})^2}{N_0/2} \tag{10.30}$$

On the other hand, from (10.8) and (10.15) we get

$$\Gamma_s = \frac{(A^{(1)})^2\, \mathtt{M}_a\, E_h/T_{chip}}{N_0/(N_{SF}\, T_{chip})} = \frac{N_{SF}\, E_h (A^{(1)})^2\, \mathtt{M}_a}{N_0} = \frac{E_s}{N_0} \tag{10.31}$$

where E_s is the average energy per symbol of the transmitted signal.

In other words, the relation between γ and Γ_s is optimum, as given by (7.114). Therefore, with regard to user $u = 1$, at the decision point the system is equivalent to an M-QAM system. However, as observed before, the transmit pulse $h_T^{(1)}$ has a bandwidth much larger than $1/(2T)$.

9) *Multi-user receiver.* The derivation of the optimum receiver carried out for user 1 can be repeated for each user. Therefore we obtain the multiuser receiver of Figure 10.6, composed of a matched filter to the transmit pulse and a despreader bank, where each branch employs a distinct user code.

We observe that for the ideal AWGN channel case, spreading the bandwidth of the transmit signal by a factor U allows the simultaneous transmission of U messages using the same frequency band.

Classification of CDMA systems

Synchronous systems. This is the case just examined, in which the user codes are orthogonal and the user signals are time-aligned. In a wireless cellular radio system, this situation occurs in the forward or downlink transmission from the base station to the mobile stations.

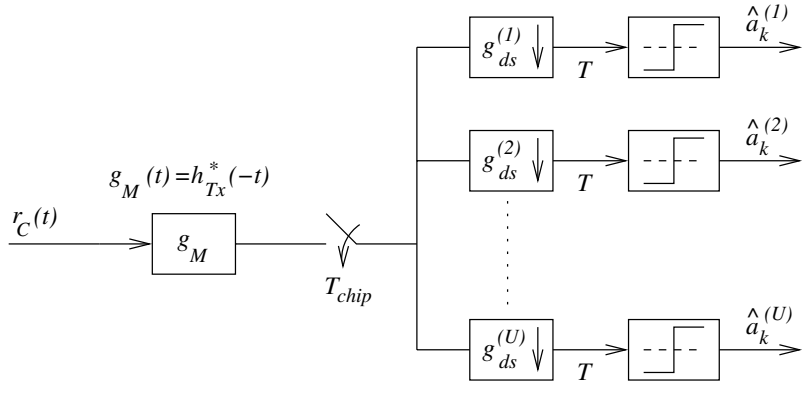


Figure 10.6. Multiuser receiver for a CDMA synchronous system with an ideal AWGN channel.

From the point of view of each mobile station, all U users share the same channel. Therefore, although the channel impulse response depends on the site of the mobile, we have

$$g_C^{(u)}(t) = g_C(t) \qquad u = 1, \ldots, U \tag{10.32}$$

and the residual interference is due to signals originating from adjacent cells in addition to the multipath interference introduced by the channel. In general, interference due to the other users within the same cell is called *multi-user interference* (MUI) or *co-channel interference* (CCI).

Asynchronous systems. In this case the various user signals are not time-aligned. In a wireless system, this situation typically occurs in the reverse or uplink transmission from the mobile stations to the base station.

Because the Walsh–Hadamard codes do not exhibit good cross-correlation properties for lags different from zero, PN scrambling sequences are used (see Appendix 3.A). The user code is then given by

$$c_m^{(u)} = c_{Ch,m}^{(u)}\, c_{scr,m} \tag{10.33}$$

where $\{c_{scr,m}\}$ may be the same for all users in a cell.

It is necessary now to make an important observation. In some systems the period of $\{c_{scr,m}\}$ is equal to the length of $\{c_{Ch,m}\}$, that is N_{SF}, whereas in other systems it is much larger than N_{SF}.[1] In the latter case, spreading and despreading operations remain unchanged, even if they are *symbol time varying*, as $\{c_m^{(u)}\}$ changes from symbol to symbol; note that consequently the receiver is also symbol time varying.

[1] This observation must not be confused with the distinction between the use of short (of period $\simeq 2^{15}$) or long (of period $\simeq 2^{42}$) PN scrambling sequences, which are employed to identify the base stations or the users and to synchronize the system [8].

Asynchronous systems are characterized by codes with low cross-correlation for non-zero lags; however, there is always a residual non-zero correlation among the various user signals. Especially in the presence of multipath channels, the residual correlation is the major cause of interference in the system, which now originates from signals within the cell: for this reason the MUI is usually characterized as *intracell* MUI.

Synchronization

The despreading operation requires that the receiver is capable of reproducing a user code sequence synchronous with that used for spreading. Therefore the receiver must first perform *acquisition*, that is the code sequence $\{c_m^{(u)}\}$ produced by the local generator must be synchronized with the code sequence of the desired user, so that the error in the time alignment between the two sequences is less than one chip interval.

As described in Section 14.7, acquisition of the desired user code sequence is generally obtained by a sequential searching algorithm that, at each step, delays the local code generator by a fraction of a chip, typically half a chip, and determines the correlation between the signals $\{x_m\}$ and $\{c_m^{(u)}\}$; the search terminates when the correlation level exceeds a certain threshold value, indicating that the desired time alignment is attained. Following the acquisition process, a *tracking* algorithm is used to achieve, in the steady state, a time alignment between the signals $\{x_m\}$ and $\{c_m^{(u)}\}$ that has the desired accuracy; the more commonly used tracking algorithms are the *delay-locked loop* and the *tau-dither loop*. The synchronization method also suggests the use of PN sequences as user code sequences. In practice, the chip frequency is limited to values of the order of hundreds of Mchip/s because of the difficulty in obtaining an accuracy of the order of a fraction of a nanosecond in the synchronization of the code generator. In turn, this determines the limit in the bandwidth of a DS signal.

10.1.2 Frequency hopping systems

The FH spread spectrum technique is typically used for the spreading of M-FSK signals. We consider an M-FSK signal (see Example 6.7.1 on page 486) with carrier frequency f_0 expressed in complex form, i.e. we consider the analytic signal, as $A\, e^{j2\pi(f_0+\Delta f(t))t}$, where $\Delta f(t) = \sum_{k=-\infty}^{+\infty} a_k\, \mathrm{w}_T(t-kT)$, with $\{a_k\}$ sequence of i.i.d. symbols taken from the alphabet $\mathcal{A} = \{-(M-1), \ldots, -1, +1, \ldots, M-1\}$, at the symbol rate $1/T$. An FH/M-FSK signal is obtained by multiplying the M-FSK signal by a signal $c_{FH}(t)$ given by

$$c_{FH}(t) = \sum_{i=-\infty}^{+\infty} e^{j(2\pi f_{0,i} t + \varphi_{0,i})}\, \mathrm{w}_{T_{hop}}(t - i T_{hop}) \tag{10.34}$$

where $\{f_{0,i}\}$ is a pseudorandom sequence that determines shifts in frequency of the FH/M-FSK signal, $\{\varphi_{0,i}\}$ is a sequence of random phases associated with the sequence of frequency shifts, and $\mathrm{w}_{T_{hop}}$ is a rectangular window of duration equal to a hop interval T_{hop}. In an FH/M-FSK system, the transmitted signal is then given by

$$s(t) = Re[c_{FH}(t)\, e^{j2\pi(f_0+\Delta f(t))t}] \tag{10.35}$$

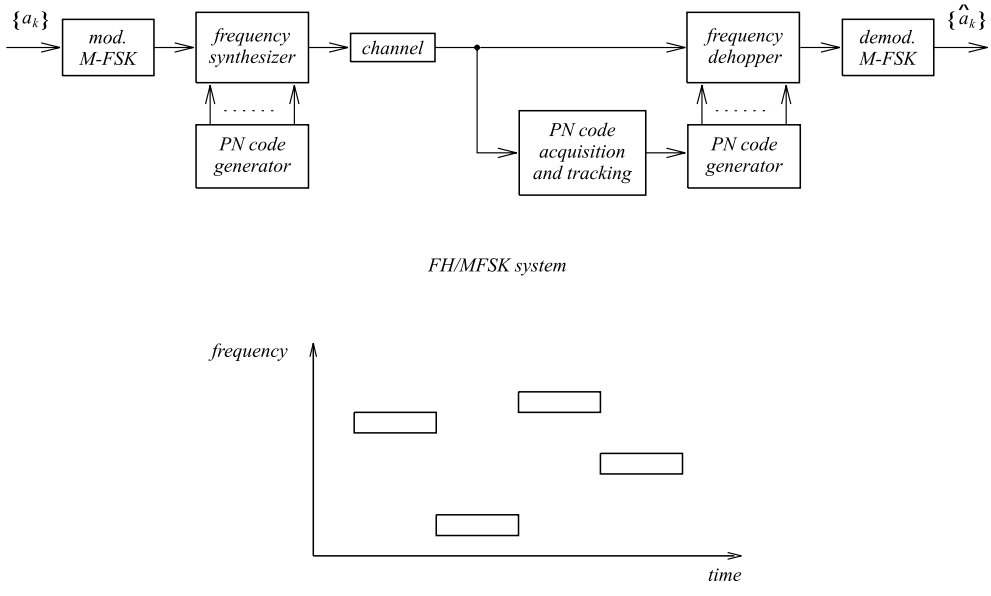


Figure 10.7. Block diagram of an FH/M-FSK system.

In practice, the signal $c_{FH}(t)$ is not generated at the transmitter; the transmitted signal $s(t)$ is obtained by applying the sequence of pseudorandom frequency shifts $\{f_{0,i}\}$ directly to the frequency synthesizer that generates the carrier at frequency f_0. With reference to the implementation illustrated in Figure 10.7, segments of L consecutive chips from a PN sequence, not necessarily disjoint, are applied to a frequency synthesizer that makes the carrier frequency hop over a set of 2^L frequencies. As the band over which the synthesizer must operate is large, it is difficult to maintain the carrier phase coherent between two consecutive hops [9]; if the synthesizer is not equipped with any device to maintain a coherent phase, it is necessary to include a random phase $\varphi_{0,i}$ as in the expression (10.34). In a time interval that is long with respect to T_{hop}, the bandwidth of the signal $s(t)$, B_{SS}, can be in practice of the order of several GHz. However, in a short time interval during which no frequency hopping occurs, the bandwidth of an FH/M-FSK signal is the same as the bandwidth of the M-FSK signal that carries the information, usually much lower than B_{SS}.

Despreading, in this case also called dehopping, is ideally carried out by multiplying the received signal $r(t)$ by a signal $\hat{c}_{FH}(t)$ equal to that used for spreading, apart from the sequence of random phases associated with the frequency shifts. For non-coherent demodulation, the sequence of random phases can be modelled as a sequence of i.i.d. random variables with uniform probability density in $[0, 2\pi)$. The operation of despreading yields the signal $x(t)$, given by the sum of the M-FSK signal, the noise and possibly interference. The signal $x(t)$ is then filtered by a lowpass filter and presented to the input of the receive section comprising a non-coherent demodulator for M-FSK signals. As in the case of DS systems, the receiver must perform acquisition and tracking of the FH signal, so that the waveform generated by the synthesizer for dehopping reproduces as accurately as possible the signal $c_{FH}(t)$.

Classification of FH systems

FH systems are traditionally classified according to the relation between T_{hop} and T. Fast frequency-hopped (FFH) systems are characterized by one or more frequency hops per symbol interval, that is $T = NT_{hop}$, N integer, and *slow frequency-hopped* (SFH) systems are characterized by the transmission of several symbols per hop interval, that is $T_{hop} = NT$.

Moreover, a chip frequency F_{chip} is defined also for FH systems, and is given by the largest value among $F_{hop} = 1/T_{hop}$ and $F = 1/T$. Therefore the chip frequency F_{chip} corresponds to the highest among the clock frequencies used by the system. The frequency spacing between the tones of an FH/M-FSK signal is related to the chip frequency and is therefore determined differently for FFH and SFH systems.

SFH systems. For SFH systems, $F_{chip} = F$, and the spacing between FH/M-FSK tones is equal to the spacing between the M-FSK tones themselves. In a system that uses a non-coherent receiver for M-FSK signals, orthogonality of tones corresponding to M-FSK symbols is obtained if the frequency spacing is an integer multiple of $1/T$. Assuming the minimum spacing is equal to F, the bandwidth B_{SS} of an FH/M-FSK signal is partitioned into $N_f = B_{SS}/F = B_{SS}/F_{chip}$ sub-bands with equally spaced center frequencies; in the most commonly used FH scheme the N_f tones are grouped into $N_b = N_f/M$ adjacent bands without overlap in frequency, each one having a bandwidth equal to $MF = MF_{chip}$, as illustrated in Figure 10.8. Assuming M-FSK modulation symmetric around the carrier frequency, the center frequencies of the $N_b = 2^L$ bands represent the set of carrier frequencies generated by the synthesizer, each associated with an L-uple of binary symbols. According to this scheme, each of the N_f tones of the FH/M-FSK signal corresponds to a unique combination of carrier frequency and M-FSK symbol.

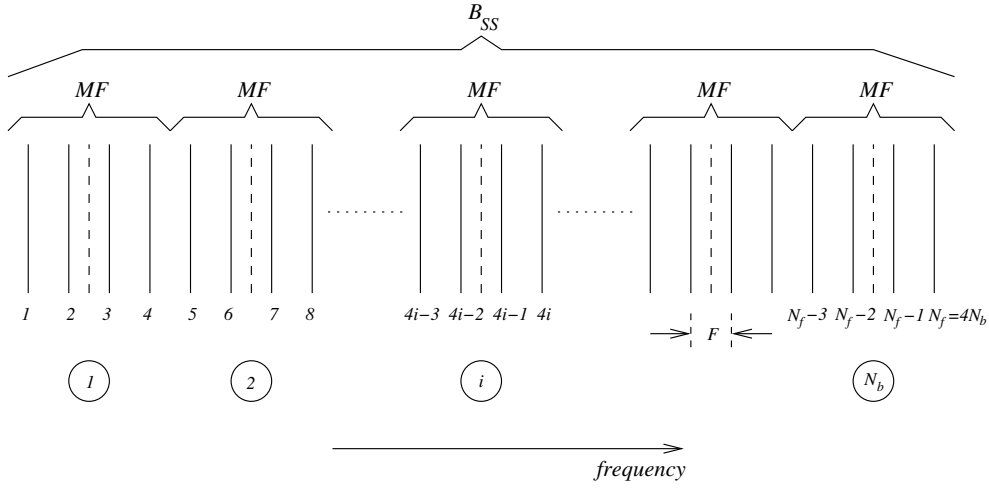


Figure 10.8. Frequency distribution for an FH/4-FSK system with bands non-overlapping in frequency; the dashed lines indicate the carrier frequencies.

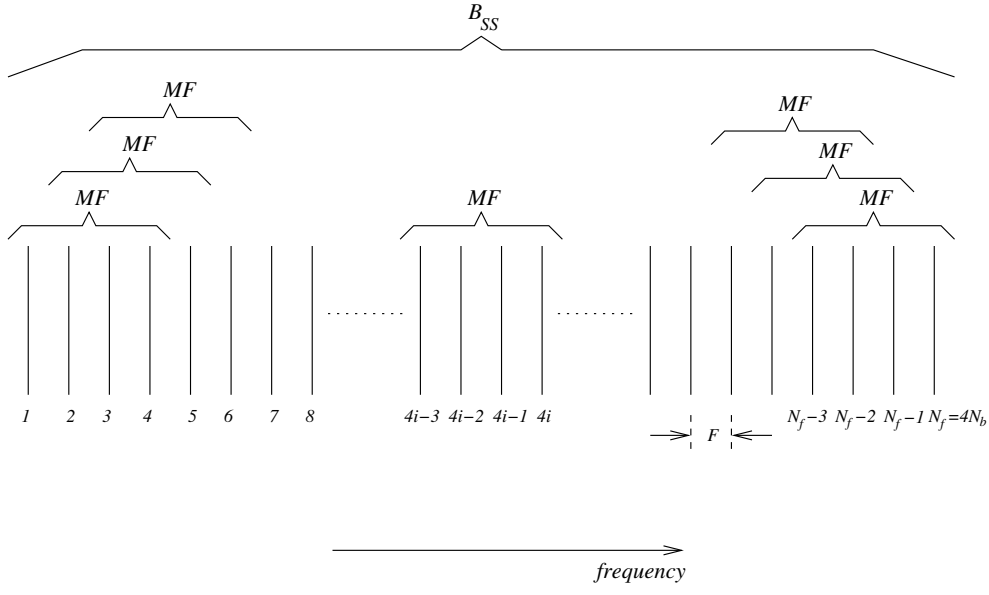


Figure 10.9. Frequency distribution for an FH/4-FSK system with bands overlapping in frequency.

In a different scheme, that yields a better protection against an intentional jammer using a sophisticated disturbance strategy, adjacent bands exhibit an overlap in frequency equal to $(M-1)F_{chip}$ Hz, as illustrated in Figure 10.9. Assuming that the center frequency of each band corresponds to a possible carrier frequency, as all N_f tones except $(M-1)$ are available as center frequencies, the number of carrier frequencies increases from N_f/M to $N_f-(M-1)$, which for $N_f \gg M$ represents an increase by a factor M of the randomness in the choice of the carrier frequency.

FFH systems. For FFH systems, where $F_{chip} = F_{hop}$, the spacing between tones of an FH/M-FSK signal is equal to the hop frequency. Therefore the bandwidth of the spread spectrum signal is partitioned into a total of $N_f = B_{SS}/F_{hop} = B_{SS}/F_{chip}$ sub-bands with equally spaced center frequencies, each corresponding to a unique L-uple of binary symbols. Because there are F_{hop}/F hops per symbol, the metric used to decide upon the symbol with a non-coherent receiver is suitably obtained by summing F_{hop}/F components of the received signal.

10.2 Applications of spread spectrum systems

The most common applications of spread spectrum systems, that will be discussed in the next sections, may be classified as follows.

1. *Multiple access*. In alternative to FDMA and TDMA systems, introduced in Section 6.13.2, spread spectrum systems allow the simultaneous transmission of messages by several users over the channel, as discussed in Section 10.1.1.

2. *Narrowband interference rejection*. We consider the DS case. Because interference is introduced in the channel after signal spreading, at the receiver the despreading operation compresses the bandwidth of the desired signal to the original value, and at the same time it expands by the same factor the bandwidth of the interference, thus reducing the level of the interference power spectral density. After demodulation the ratio between the desired signal power and the interference power is therefore larger than that obtained without spreading the signal spectrum.

3. *Robustness against fading*. Widening the signal bandwidth allows exploitation of the multipath diversity of a radio channel affected by fading. Applying a DS spread spectrum technique, intuitively, has the effect of modifying a channel model that is adequate for transmission of narrowband signals in the presence of flat fading or multipath fading with a few rays, to a channel model with many rays. Using a receiver that combines the desired signal from the different propagation rays, the power of the desired signal at the decision point increases. In an FH system, on the other hand, we obtain diversity in the time domain, as the channel changes from one hop interval to the next. The probability that the signal is affected by strong fading during two consecutive hop intervals is usually low. To recover the transmitted message in a hop interval during which strong fading is experienced, error correction codes with very long interleaver and ARQ schemes are used (see Chapter 11).

10.2.1 Anti-jam communications

Narrowband interference We consider the baseband equivalent signals of an M-QAM passband communication system with symbol rate $F = 1/T$, transmitted signal power equal to $\mathtt{M}_s$, and PSD with minimum bandwidth, i.e. $\mathcal{P}_s(f) = E_s \,\mathrm{rect}(f/F)$, where $E_s F = \mathtt{M}_s$.

We now consider the application of a DS spread spectrum modulation system. Due to spreading, the bandwidth of the transmitted signal is expanded from F to $B_{SS} = N_{SF}F$. Therefore, for the same transmitted signal power, the PSD of the transmitted signal becomes $\mathcal{P}_{s'}(f) = (E_s/N_{SF})\,\mathrm{rect}(f/B_{SS})$, where $E_s/N_{SF} = \mathtt{M}_s/B_{SS}$. We note that spreading has decreased the amplitude of the PSD by the factor N_{SF}, as illustrated in Figure 10.10.

In the band of the spread spectrum signal, in addition to additive white Gaussian noise with PSD N_0, we assume the channel introduces an additive interference signal or jammer with power $\mathtt{M}_\mathtt{j}$, uniformly distributed on a bandwidth $B_\mathtt{j}$, with $B_\mathtt{j} < 1/T$.

With regard to the operation of despreading, we consider the signals after the multiplication by the user code sequence. The interference signal spectrum is expanded and has a PSD equal to $\mathcal{P}_{\mathtt{j}'}(f) = N_\mathtt{j}\,\mathrm{rect}(f/B_{SS})$, with $N_\mathtt{j} = \mathtt{M}_\mathtt{j}/B_{SS}$. The noise, that originally has a uniformly distributed power over all the frequencies, still has PSD equal to N_0, i.e. spreading has not changed the PSD of the noise.

At the output of the despreading the desired signal exhibits the original PSD equal to $E_s\,\mathrm{rect}(f/F)$. Modeling the despreader filter as an ideal lowpass filter with bandwidth $1/(2T)$, for the signal-to-noise ratio γ at the decision point the following relation holds:

$$\left(\frac{1}{2}\mathtt{M}_a\right)\gamma = \frac{E_s}{N_0 + N_\mathtt{j}} = \frac{\mathtt{M}_s/F}{(N_0 + \mathtt{M}_\mathtt{j}/B_{SS})} \tag{10.36}$$

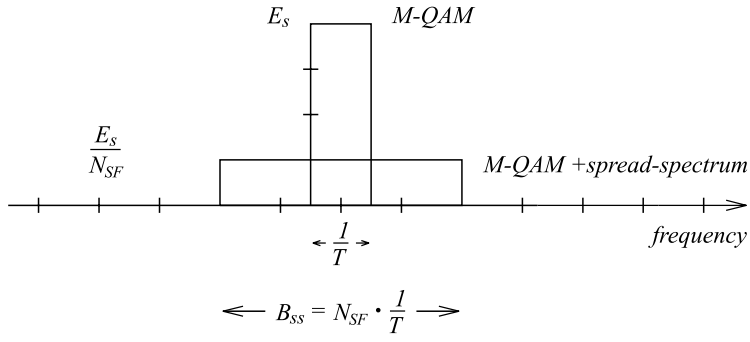


Figure 10.10. Power spectral density of an M-QAM signal with minimum bandwidth and of a spread spectrum M-QAM signal with spreading factor $N_{SF} = 4$.

In practice, performance is usually limited by interference and the presence of white noise can be ignored. Therefore, assuming $N_{\mathtt{j}} \gg N_0$, (10.36) becomes

$$\left(\frac{1}{2}\,\mathtt{M}_a\right)\gamma \simeq \frac{E_s}{N_{\mathtt{j}}} = \frac{\mathtt{M}_s/F}{\mathtt{M}_{\mathtt{j}}/B_{SS}} = \frac{\mathtt{M}_s}{\mathtt{M}_{\mathtt{j}}}\,\frac{B_{SS}}{F} \tag{10.37}$$

where $\mathtt{M}_s/\mathtt{M}_{\mathtt{j}}$ is the ratio between the power of the desired signal and the power of the jammer, and B_{SS}/F is the spreading ratio N_{SF} also defined as the *processing gain* of the system.

The above considerations are now defined more precisely in the following case.

Sinusoidal interference. We assume that the baseband equivalent received signal is expressed as

$$r_C(t) = s(t) + \mathtt{j}(t) + w_C(t) \tag{10.38}$$

where $s(t)$ is a DS signal given by (10.9) with amplitude $A^{(u)} = 1$, $w_C(t)$ is AWGN with spectral density N_0, and the interferer is given by

$$\mathtt{j}(t) = A_{\mathtt{j}}\, e^{j\varphi} \tag{10.39}$$

In (10.39) $A_{\mathtt{j}} = \sqrt{\mathtt{M}_{\mathtt{j}}}$ is the amplitude of the jammer and φ a random phase with uniform distribution in $[0, 2\pi)$. We also assume a minimum bandwidth transmit pulse, $h_{Tx}(t) = \sqrt{E_h/T_{chip}}\,\mathrm{sinc}(t/T_{chip})$, hence $g_M(t) = h_{Tx}(t)$, and $\mathcal{G}_M(0) = \sqrt{E_h\,T_{chip}}$.

For the coherent receiver of Figure 10.4, at the detection point the sample at instant kT is given by

$$y_k = N_{SF}\,E_h\,a_k + w_k + A_{\mathtt{j}}\,e^{j\varphi}\,\mathcal{G}_M(0)\sum_{j=0}^{N_{SF}-1} c^*_{j+kN_{SF}} \tag{10.40}$$

Modeling the sequence $\{c^*_{kN_{SF}}, c^*_{kN_{SF}+1}, \ldots, c^*_{kN_{SF}+N_{SF}-1}\}$ as a sequence of i.i.d. random variables, the variance of the summation in (10.40) is equal to N_{SF}, and the ratio γ

is given by

$$\gamma = \frac{(N_{SF}\, E_h)^2}{(N_{SF}\, N_0\, E_h + \mathtt{M}_\mathtt{j}\, E_h\, T_{chip}\, N_{SF})/2} \tag{10.41}$$

Using (10.8) and the relation $E_s = \mathtt{M}_s T$, we obtain

$$\left(\frac{1}{2}\,\mathtt{M}_a\right)\gamma = \frac{1}{N_0/E_s + \mathtt{M}_\mathtt{j}/(N_{SF}\,\mathtt{M}_s)} \tag{10.42}$$

We note that in the denominator of (10.42) the ratio $\mathtt{M}_\mathtt{j}/\mathtt{M}_s$ is divided by N_{SF}. Recognizing that $\mathtt{M}_\mathtt{j}/\mathtt{M}_s$ is the ratio between the power of the jammer and the power of the desired signal before the despreading operation, and that $\mathtt{M}_\mathtt{j}/(N_{SF}\,\mathtt{M}_s)$ is the same ratio after the despreading, we find that, by analogy with the previous case of narrowband interference, also in the case of a sinusoidal jammer the use of the DS technique reduces the effect of the jammer by a factor equal to the processing gain.

10.2.2 Multiple-access systems

Spread spectrum multiple-access communication systems represent an alternative to TDMA or FDMA systems and are normally referred to as CDMA systems (see Section 6.13.2 and Section 10.1.1). With CDMA, a particular spreading sequence is assigned to each user to access the channel; unlike FDMA, where users transmit simultaneously over non-overlapping frequency bands, or TDMA, where users transmit over the same band but in disjoint time intervals, users in a CDMA system transmit simultaneously over the same frequency band.

Because in CDMA systems correlation receivers are usually employed, it is important that the spreading sequences are characterized by low cross-correlation values. We have already observed that CDMA systems may be classified as synchronous or asynchronous. In the first case the symbol transition instants of all users are aligned; this allows the use of orthogonal sequences as spreading sequences and consequently the elimination of interference caused by one user signal to another; in the second case the interference caused by multiple access limits the channel capacity, but the system design is simplified.

CDMA has received particular interest for applications in wireless communications systems, for example, cellular radio systems, personal communications services (PCS), and wireless local-area networks; this interest is mainly due to performance that spread spectrum systems achieve for the transmission over channels characterized by multipath fading.

Other properties make CDMA interesting for application to cellular radio systems, for example the possibility of applying the concept of frequency reuse (see Chapter 17). In cellular radio systems based on FDMA or TDMA, to avoid excessive levels of interference from one cell onto neighboring cells, the frequencies used in one cell are not used in neighboring cells. In other words, the system is designed so that there is a certain spatial separation between cells that use the same frequencies. For CDMA, this spatial separation

is not necessary, making it possible, in principle, to reuse all frequencies. Moreover, as CDMA systems tend to be limited by interference, an increase in system capacity is obtained by detecting the speech signal activity. This gain is made possible by the fact that in every telephone conversation each user speaks only for about half the time, while in the silence intervals he does not contribute to instantaneous interference. If several users can be served by the system, on average only half of them are active at a given instant, and the effective capacity can be doubled.

10.2.3 Interference rejection

Besides the above described properties, that are relative to the application in multiple-access systems, the robustness of spread spectrum systems in the presence of narrowband interferers is key in other applications, for example, in systems where interference is unintentionally generated by other users that transmit over the same channel. We have CCI when a certain number of services are simultaneously offered to users transmitting over the same frequency band. Although in these cases some form of spatial separation among signals interfering with each other is usually provided, for example, by using directional antennas, it is often desirable to use spread spectrum systems for their inherent interference suppression capability. In particular, we consider a scenario in which a frequency band is only partially occupied by a set of narrowband conventional signals: to increase the spectral efficiency of the system, a set of spread spectrum signals can simultaneously be transmitted over the same band, thus allowing two sets of users to access the transmission channel. Clearly, this scheme can be implemented only if the mutual interference, which a signal set imposes on the other, remains within tolerable limits.

10.3 Chip matched filter and rake receiver

Before introducing a structure that is often employed in receivers for DS spread spectrum signals, we make the following considerations on the radio channel model introduced in Section 4.6.

Number of resolvable rays in a multipath channel

We want to represent a multipath radio channel with a number of rays having gains modeled as complex valued, Gaussian *uncorrelated* random processes. From (4.206), apart from a complex constant, the channel impulse response with *infinite bandwidth* is given by

$$g_C(\tau) = \sum_{i=0}^{N_{c,\infty}-1} g_i \, \delta(\tau - iT_Q) \tag{10.43}$$

where for simplicity we have assumed the absence of Doppler spread. Therefore the non-zero gains $\{g_i\}$ are uncorrelated random variables and the delays $\tau_i = iT_Q$ are multiples of a sufficiently small period T_Q.

Hence, from (10.43), the channel output signal s_C is related to the input signal s by

$$s_C(t) = \sum_{i=0}^{N_{c,\infty}-1} g_i \, s(t - iT_Q) \tag{10.44}$$

Now the number of resolvable or uncorrelated rays in (10.44) is generally less than $N_{c,\infty}$ and is related to the bandwidth of s by the following rule: if s has a bandwidth B, the uncorrelated rays are spaced by a delay of the order of $1/B$. Consequently, for a channel with a delay spread τ_{rms} and bandwidth $B \propto 1/T_{chip}$, the number of resolvable rays is given by

$$N_{c,res} \propto \frac{\tau_{rms}}{T_{chip}} \tag{10.45}$$

Using the notion of channel coherence bandwidth, $B_{ccb} \propto 1/\tau_{rms}$, (10.45) may be rewritten as

$$N_{c,res} \propto \frac{B}{B_{ccb}} \tag{10.46}$$

We now give an example that illustrates the above considerations. Let $\{g_C(nT_Q)\}$ be a realization of the channel impulse response with uncorrelated coefficients having a given power delay profile; the "infinite bandwidth" of the channel will be equal to $B = 1/(2T_Q)$. We now filter $\{g_C(nT_Q)\}$ with two filters having, respectively, bandwidth $B = 0.1/(2T_Q)$ and $B = 0.01/(2T_Q)$, and we compare the three pulse shapes given by the input sequence and the two output sequences. We note that the output obtained in correspondence of the filter with the narrower bandwidth has fewer resolvable rays. In fact, in the limit for $B \to 0$ the output is modeled as a single random variable.

Another way to derive (10.45) is to observe that, for t within an interval of duration $1/B$, s does not vary much. Therefore, letting

$$N_{cor} = \frac{N_{c,\infty}}{N_{c,res}} \tag{10.47}$$

equation (10.44) can be written as

$$s_C(t) = \sum_{j=0}^{N_{c,res}-1} g_{res,j} \, s(t - jN_{cor}T_Q) \tag{10.48}$$

where

$$g_{res,j} \simeq \sum_{i=0}^{N_{cor}-1} g_{i+jN_{cor}} \tag{10.49}$$

are the gains of the resolvable rays.

The conclusion is that, assuming the symbol period T is given and DS spread spectrum modulation is adopted, the larger, the N_{SF}, the greater the resolution of the radio channel, that is, the channel can be modeled with a larger number of uncorrelated rays, with delays of the order of T_{chip}.

Chip matched filter (CMF)

We consider the transmission of a *DS signal* (10.9) for $U = 1$ on a *dispersive channel* as described by (10.48). The receiver that maximizes the ratio between the amplitude of the pulse associated with the desired signal sampled with sampling rate $1/T_{chip}$ and the standard deviation of the noise is obtained by the filter matched to the received pulse. We define

$$q_C(t) = (h_{Tx} * g_C * g_{AA})(t) \tag{10.50}$$

and let $g_M(t) = q_C^*(t_0 - t)$ be the corresponding matched filter. In practice, at the output of the filter g_{AA} an estimate of q_C with sampling period $T_c = T_{chip}/2$ is evaluated,[2] which yields the corresponding discrete-time matched filter with sampling period of the input signal equal to T_c and sampling period of the output signal equal to T_{chip} (see Figure 10.11).

If q_C is sparse, that is, it has a large support but only a few non-zero coefficients, for the realization of g_M we retain only the coefficients of q_C with larger amplitude; it is better to set to zero the remaining coefficients because their estimate is usually very noisy (see Appendix 3.A).

Figure 10.12a illustrates in detail the receiver of Figure 10.11 for a filter g_M with at most N_{MF} coefficients spaced of $T_c = T_{chip}/2$. If we now implement the despreader on every branch of the filter g_M, we obtain the structure of Figure 10.12b. We observe that typically only 3 or 4 branches are active, that is they have a coefficient $g_{M,i}$ different from zero.

Ideally, for an overall channel with N_{res} resolvable paths, we assume

$$q_C(t) = \sum_{i=1}^{N_{res}} q_{C,i}\, \delta(t - \tau_i) \tag{10.51}$$

hence

$$g_M(t) = \sum_{j=1}^{N_{res}} q_{C,j}^*\, \delta(t_0 - t - \tau_j) \tag{10.52}$$

Defining

$$t_{M,j} = t_0 - \tau_j \qquad j = 1, \ldots, N_{res} \tag{10.53}$$

the receiver scheme, analogous to that of Figure 10.12b, is illustrated in Figure 10.13.

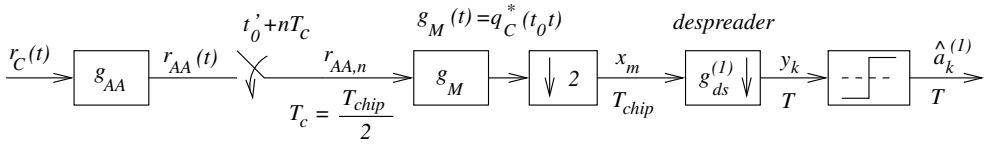


Figure 10.11. Chip matched filter receiver for a dispersive channel.

[2] To determine the optimum sampling phase t_0, usually r_{AA} is oversampled with a period T_Q such that $T_c/T_Q = 2$ or 4 for $T_c = T_{chip}/2$; among the 2 or 4 estimates of g_C obtained with sampling period T_c, the one with the largest energy is selected (see Observation 8.5 on page 641).

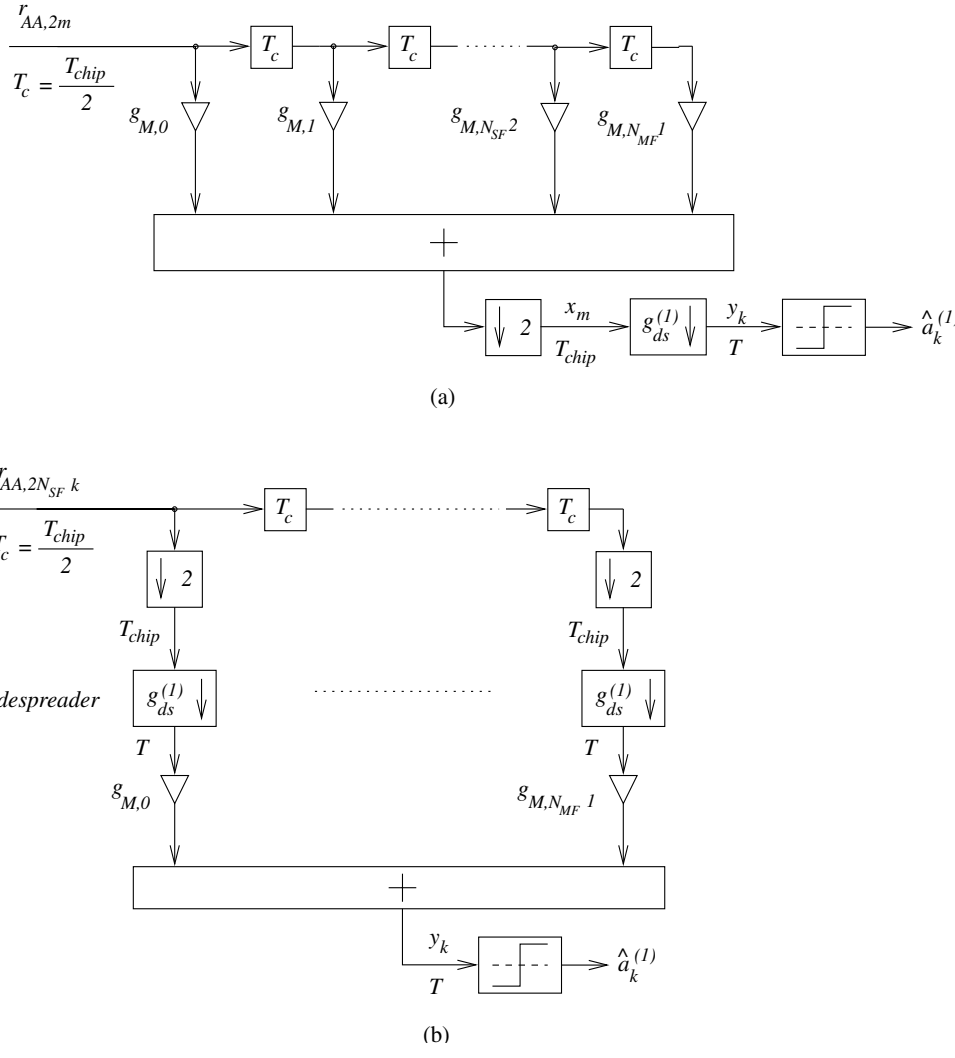


Figure 10.12. Two receiver structures: (a) chip matched filter with despreader, (b) rake.

To simplify the analysis, we assume that the spreading sequence is a PN sequence with N_{SF} sufficiently large, such that the following approximations hold: 1) the autocorrelation of the spreading sequence is a Kronecker delta: and 2) the delays $\{\tau_i\}$ are multiples of T_{chip}.

From (10.51), in the *absence of noise* the signal r_{AA} is given by

$$r_{AA}(t) = \sum_{n=1}^{N_{res}} q_{C,n} \sum_{i=-\infty}^{+\infty} a_i^{(1)} \sum_{\ell=0}^{N_{SF}-1} c_{\ell+iN_{SF}}^{(1)} \delta(t - \tau_n - (\ell + iN_{SF})\, T_{chip}) \tag{10.54}$$

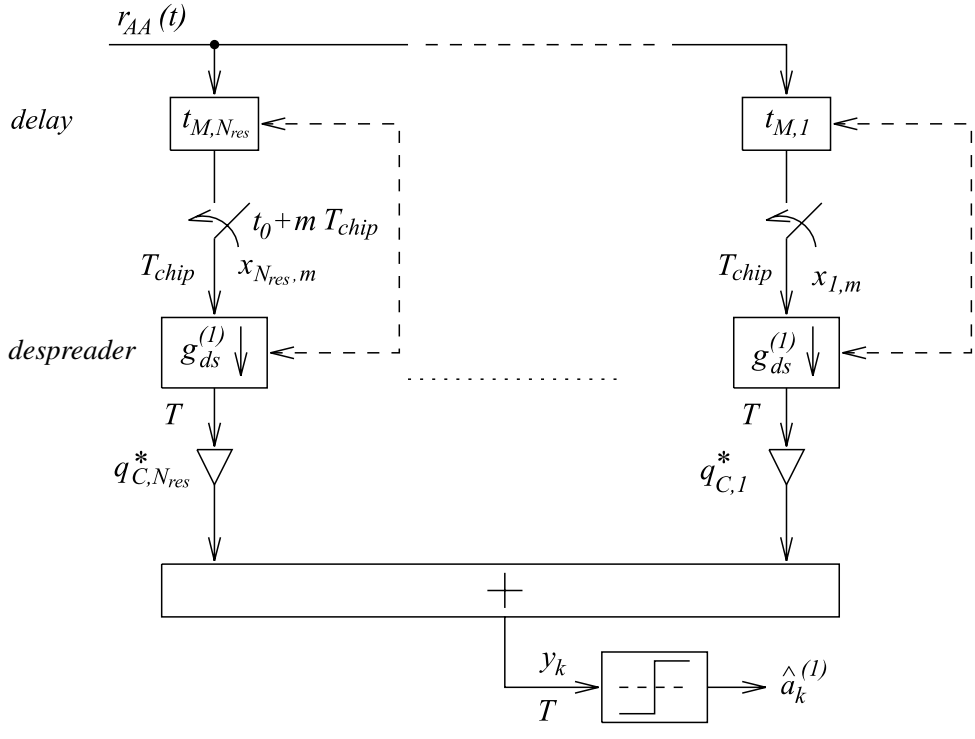


Figure 10.13. Rake receiver for a channel with N_{res} resolvable paths.

and the output of the sampler on branch j is given by[3]

$$x_{j,m} = \sum_{n=1}^{N_{res}} q_{C,n} \sum_{i=-\infty}^{+\infty} a_i^{(1)} \sum_{\ell=0}^{N_{SF}-1} c_{\ell+i\,N_{SF}}^{(1)}\, \delta_{m+\frac{\tau_j-\tau_n}{T_{chip}}-(\ell+i\,N_{SF})} \tag{10.55}$$

Correspondingly the despreader output, assuming $\mathfrak{r}_{c^{(1)}}(n_D) = \delta_{n_D}$ and the absence of noise, yields the signal $N_{SF}\, a_k^{(1)}\, q_{C,j}$. The contributions from the various branches are then combined according to the MRC technique (see Section 6.13) to yield the sample

$$y_k = \left(N_{SF} \sum_{n=1}^{N_{res}} |q_{C,n}|^2 \right) a_k^{(1)} \tag{10.56}$$

where $E_{q_C} = \sum_{n=1}^{N_{res}} |q_{C,n}|^2$ is the energy per chip of the overall channel impulse response.

The name rake originates from the structure of the receiver that is similar to a rake with N_{res} fingers. In practice, near the rake receiver a correlator estimates the delays, with precision $T_{chip}/2$, and the gains of the various channel rays. The rake is initialized with the

[3] Instead of using the Dirac delta in (10.51), a similar analysis assumes that 1) $g_{AA}(t) = h^*_{Tx}(-t)$, and 2) $\mathfrak{r}_{h_{Tx}}(t)$ is a Nyquist pulse. The result is the same as (10.55).

coefficients of rays with larger gain. The delays and the coefficients are updated whenever a change in the channel impulse response is observed. However, after the initialization has taken place, on each finger of the rake the estimates of the amplitude and of the delay of the corresponding ray may be refined by using the correlator of the despreader, as indicated by the dotted line in Figure 10.13. We note that if the channel is static, the structure of Figure 10.12a with $T_c = T_{chip}/2$ yields a sufficient statistic.

10.4 Interference

For a dispersive channel and in the case of U users, we evaluate the expression of the signal y_k at the decision point using the matched filter receiver of Figure 10.11.

Similarly to (10.50), we define

$$q_C^{(u)}(t) = (h_{Tx} * g_C^{(u)} * g_{AA})(t) \qquad u = 1, \ldots, U \tag{10.57}$$

and let

$$g_M^{(v)}(t) = q_C^{(v)*}(t_0 - t) \qquad v = 1, \ldots, U \tag{10.58}$$

be the corresponding matched filter. Moreover, we introduce the correlation between $q_C^{(u)}$ and $q_C^{(v)}$, expressed by

$$\mathtt{r}_{q_C^{(u)} q_C^{(v)}}(\tau) = (q_C^{(u)}(t) * q_C^{(v)*}(-t))(\tau) \tag{10.59}$$

Assuming without loss of generality that the desired user signal has the index $u = 1$, we refer to the receiver of Figure 10.11, where we have $g_M(t) = g_M^{(1)}(t) = q_C^{(1)*}(t_0 - t)$, and

$$x_m = \sum_{u=1}^{U} A^{(u)} \sum_{i=-\infty}^{+\infty} a_i^{(u)} \sum_{\ell=0}^{N_{SF}-1} c_{\ell+iN_{SF}}^{(u)} \, \mathtt{r}_{q_C^{(u)} q_C^{(1)}}((m - \ell - iN_{SF})\, T_{chip}) + \tilde{w}_m \tag{10.60}$$

where $\tilde{w}_m$ is given by (10.20).

At the despreader output we obtain

$$\begin{aligned} y_k &= \sum_{j=0}^{N_{SF}-1} x_{j+kN_{SF}} \, c_{j+kN_{SF}}^{(1)*} + w_k \\ &= \sum_{u=1}^{U} A^{(u)} \sum_{i=-\infty}^{+\infty} a_i^{(u)} \sum_{\ell=0}^{N_{SF}-1} \sum_{j=0}^{N_{SF}-1} c_{\ell+iN_{SF}}^{(u)} \\ &\qquad \mathtt{r}_{q_C^{(u)} q_C^{(1)}}((j - \ell + (k - i)\, N_{SF})\, T_{chip}) c_{j+kN_{SF}}^{(1)*} + w_k \end{aligned} \tag{10.61}$$

where w_k is defined in (10.28).

Introducing the change of index $n = \ell - j$ and recalling the definition of cross-correlation between two code sequences (10.26), the double equation in ℓ and j in (10.61) can be written as

$$\begin{aligned}&\sum_{n=-(N_{SF}-1)}^{-1} \mathtt{r}_{q_C^{(u)}\, q_C^{(1)}}((-n + (k-i)\, N_{SF})T_{chip})(N_{SF} - |n|)\mathtt{r}^*_{c^{(1)}\, c^{(u)}}(-n) \\ &+ \sum_{n=0}^{N_{SF}-1} \mathtt{r}_{q_C^{(u)}\, q_C^{(1)}}((-n + (k-i)\, N_{SF})T_{chip})(N_{SF} - |n|)\mathtt{r}_{c^{(u)}\, c^{(1)}}(n)\end{aligned} \tag{10.62}$$

where, to simplify the notation, we have assumed that the user code sequences are periodic of period N_{SF}.

The desired term in (10.61) is obtained for $u = 1$; as $\mathtt{r}^*_{c^{(1)}}(-n) = \mathtt{r}_{c^{(1)}}(n)$, it has the following expression:

$$A^{(1)} \sum_{i=-\infty}^{+\infty} a_i^{(1)} \sum_{n=-(N_{SF}-1)}^{N_{SF}-1} (N_{SF} - |n|)\, \mathtt{r}_{c^{(1)}}(n)\, \mathtt{r}_{q_C^{(1)}}((-n + (k-i)\, N_{SF})\, T_{chip}) \tag{10.63}$$

Consequently, if the code sequences are orthogonal, that is

$$\mathtt{r}_{c^{(1)}}(n) = \delta_n \tag{10.64}$$

and in the absence of ISI, that is

$$\mathtt{r}_{q_C^{(1)}}(i N_{SF}\, T_{chip}) = \delta_i\, E_{q_C^{(1)}} \tag{10.65}$$

where $E_{q_C^{(1)}}$ is the energy per chip of the overall pulse at the output of the filter g_{AA}, then the desired term (10.63) becomes

$$A^{(1)}\, N_{SF}\, E_{qC}\, a_k^{(1)} \tag{10.66}$$

which coincides with the case of an ideal AWGN channel (see (10.27)). Note that using the same assumptions we find the rake receiver behaves as an MRC (see (10.56)).

If (10.64) is not verified, as happens in practice, and if

$$\mathtt{r}_{q_C^{(1)}}(nT_{chip}) \neq \delta_n\, E_{qC} \tag{10.67}$$

the terms for $n \neq 0$ in (10.63) give rise to intersymbol interference, in this context also called *inter-path interference* (IPI). Usually the smaller the N_{SF}, the larger the IPI. We note, however, that if the overall pulse at the output of the CMF is a Nyquist pulse, that is

$$\mathtt{r}_{q_C^{(1)}}(nT_{chip}) = \delta_n\, E_{qC} \tag{10.68}$$

then there is no IPI, even if (10.64) is not verified.

With reference to (10.62) we observe that, in the multiuser case, if $\mathtt{r}_{c^{(u)}\, c^{(1)}}(n) \neq 0$ then y_k is affected by MUI, whose value increases as the cross-correlation between the pulses $q_C^{(u)}$ and $q_C^{(1)}$ increases.

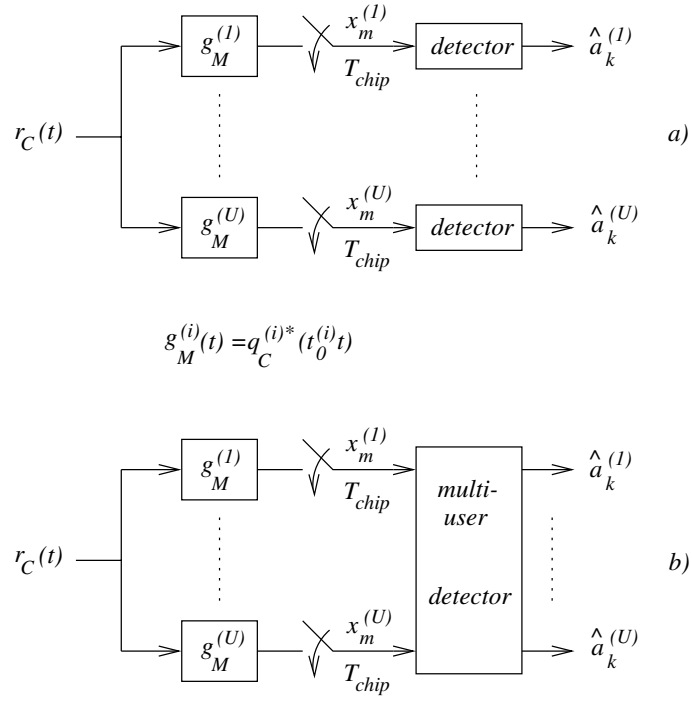


Figure 10.14. (a) Single-user receiver, and (b) multiuser receiver.

Detection strategies for multiple-access systems

For detection of the user messages in CDMA systems, we make a distinction between two classes of receivers: single-user and multiuser. In the first class the receivers focus on detecting the data from a single user, and the other user signals are considered as uncancellable interference. In the second class the receivers seek to simultaneously detect all U messages. The performance of the multiuser receivers is substantially better than that of the single-user receivers, achieved at the expense of a higher computational complexity. Using as front-end a filter bank, where the filters are matched to the channel impulse responses of the U users, the structures of single-user and multiuser receivers are exemplified in Figure 10.14.

10.5 Equalizers for single-user detection

We consider two equalizers for single-user detection.

Chip equalizer (CE)

To mitigate the interference in the signal sampled at the chip rate, after the CMF (see (10.68)) a ZF or an MSE equalizer can be used [10, 11, 12, 13]. As illustrated in Figure 10.15, let g_{CE} be the equalizer filter with output $\{\tilde{d}_m\}$. For an MSE criterion the cost

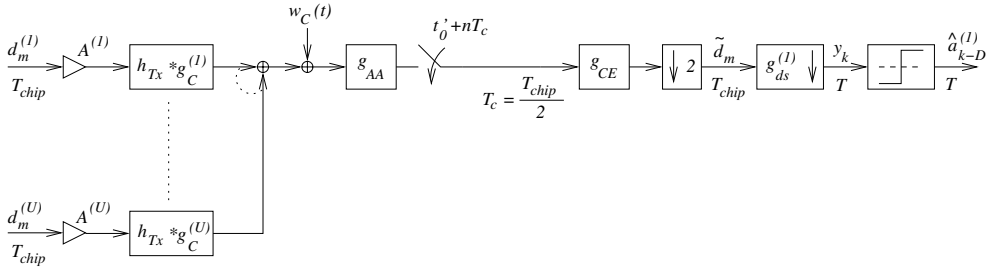


Figure 10.15. Receiver as a fractionally-spaced chip equalizer.

function is given by

$$J = E[|\tilde{d}_m - d_m|^2] \tag{10.69}$$

where $\{d_m\}$ is assumed i.i.d. We distinguish the two following cases:

1) *All code sequences are known.* This is the case that may occur for downlink transmission in wireless networks. Then $g_C^{(u)}(t) = g_C(t)$, $u = 1, \ldots, U$, and we assume

$$d_m = \sum_{u=1}^{U} d_m^{(u)} \tag{10.70}$$

that is, for the equalizer design, all user signals are considered as desired signals.

2) *Only the code sequence of the desired user signal is known.* In this case we need to assume

$$d_m = d_m^{(1)} \tag{10.71}$$

The other user signals are considered as white noise, with overall PSD N_i, that is added to w_C.

From the knowledge of $q_C^{(1)}$ and the overall noise PSD, the minimum of the cost function defined in (10.69) is obtained by following the same steps developed in Chapter 8. Obviously, if the level of interference is high, the solution corresponding to (10.71) yields a simple CMF, with low performance whenever the residual interference (MUI and IPI) at the decision point is high.

A better structure for single-user detection is obtained by the following approach.

Symbol equalizer (SE)

Recalling that we adopt the transmitter model of Figure 10.3, and that we are interested in the message $\{a_k^{(1)}\}$, the optimum receiver with linear filter $g_{SE}^{(1)}$ is illustrated in Figure 10.16.

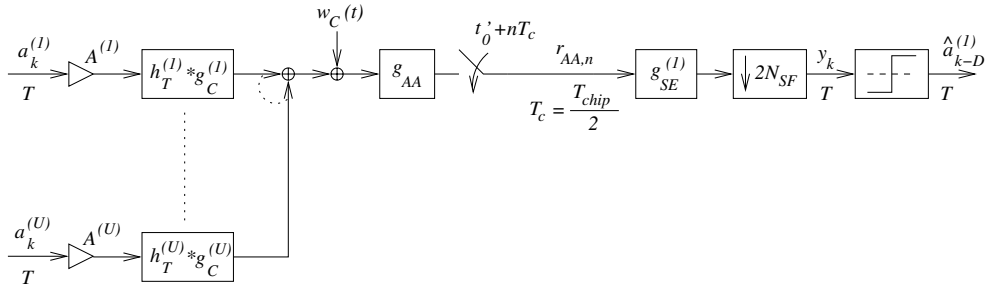


Figure 10.16. Receiver as a fractionally-spaced symbol equalizer.

The cost function is now given by [14, 15, 16]

$$J = E[|y_k - a_{k-D}|^2] \tag{10.72}$$

Note that $g_{SE}^{(1)}$, that includes also the function of despreading, depends on the code sequence of the desired user. Therefore the length of the code sequence is usually not larger than N_{SF}, otherwise we would find a different solution for every symbol period, even if $g_C^{(1)}$ is time invariant. Moreover, in this formulation the other user signals are seen as interference, and one of the tasks of g_{SE} is to mitigate the MUI.

In an adaptive approach, for example, using the LMS algorithm, the solution is simple to determine and does not require any particular *a priori* knowledge, except the training sequence in $\{a_k^{(1)}\}$ for initial convergence. On the other hand, using a direct approach we need to identify the autocorrelation of $r_{AA,n}$ and the cross-correlation between $r_{AA,n}$ and $a_{k-D}^{(1)}$. As usual these correlations are estimated directly or, assuming the messages $\{a_k^{(u)}\}$, $u = 1, \ldots, U$, are i.i.d. and independent of each other, we can determine them using the knowledge of the various pulses $\{h_T^{(u)}\}$ and $\{g_C^{(u)}\}$, that is the channel impulse responses and code sequences of all users; for the special case of downlink transmission, the knowledge of the code sequences is sufficient, as the channel is common to all user signals.

10.6 Block equalizer for multiuser detection

Multiuser detection techniques are essential for achieving near-optimum performance in communication systems where signals conveying the desired information are received in the presence of ambient noise plus multiple-user interference. The leitmotiv of developments in multiuser detection is represented by the reduction in complexity of practical receivers with respect to that of optimal receivers, which is known to increase exponentially with the number of active users and with the delay spread of the channel, while achieving near-optimum performance. A further element that is being recognized as essential to reap the full benefits of interference suppression is the joint application of multiuser detection with other techniques such as spatial-temporal processing and iterative decoding.

Here we first consider the simplest among multiuser receivers. It comprises a bank of U filters $g_T^{(u)}$, $u = 1, \ldots, U$, matched to the impulse responses[4]

$$q_T^{(u)}(t) = \sum_{\ell=0}^{N_{SF}-1} c_\ell^{(u)} \, q_C^{(u)}(t - \ell \, T_{chip}) \qquad u = 1, \ldots, U \tag{10.73}$$

where the functions $\{q_C^{(u)}(t)\}$ are defined in (10.57). Decisions taken by threshold detectors on the U output signals, sampled at the symbol rate, yield the detected user symbol sequences. It is useful to introduce this receiver, that we denote as MF, as, substituting the threshold detectors with more sophisticated detection devices, it represents the first stage of several multiuser receivers, as illustrated in general in Figure 10.17.

We introduce the following vector notation. The vector of symbols transmitted by U users in a symbol period T is expressed as

$$\mathbf{a}_k = [a_k^{(1)}, \ldots, a_k^{(U)}]^T \tag{10.74}$$

and the vector that carries the information on the codes and the channel impulse responses of the U users is expressed as

$$\mathbf{q}_T(t) = [q_T^{(1)}(t), \ldots, q_T^{(U)}(t)]^T \tag{10.75}$$

Joint detectors constitute an important class of multiuser receivers. They effectively mitigate both ISI and MUI, exploiting the knowledge of the vector $\mathbf{q}_T$. In particular we consider now *block linear receivers*: as the name suggests, a block linear receiver is a joint detector that recovers the information contained in a window of K symbol periods. Let

$$\begin{aligned} \mathbf{a} &= [\mathbf{a}_0^T, \ldots, \mathbf{a}_{K-1}^T]^T \\ &= [a_0^{(1)}, \ldots, a_0^{(U)}, \ldots, a_{K-1}^{(1)}, \ldots, a_{K-1}^{(U)}]^T \end{aligned} \tag{10.76}$$

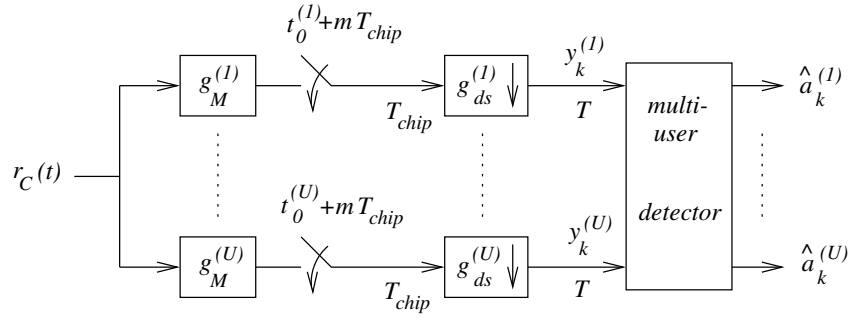


Figure 10.17. Receiver as MF and multiuser detector.

[4] We assume that the information on the power of the user signals is included in the impulse responses $g_C^{(u)}$, $u = 1, \ldots, U$, so that $A^{(u)} = 1$, $u = 1, \ldots, U$.

be the information transmitted by U users and let $\mathbf{y}$ be the corresponding vector of KU elements at the MF output. We define the following correlations:

$$\mathtt{r}_{q(u,v)}(k) = (q_T^{(u)}(t) * q_T^{(v)*}(-t))(\tau)|_{\tau=kT} \tag{10.77}$$

Assuming

$$\mathtt{r}_{q(u,v)}(k) = 0 \quad \text{for } |k| > \nu \tag{10.78}$$

with $\nu < K$, and following the approach in [17, 18, 19, 20], we introduce the $KU \times KU$ matrix,

$$\mathbf{T} = \left[\begin{array}{ccc|c|ccc|c}
\mathtt{r}_{q^{(1,1)}}(0) & \dots & \mathtt{r}_{q^{(1,U)}}(0) & \dots & \mathtt{r}_{q^{(1,1)}}(-\nu) & \dots & \mathtt{r}_{q^{(1,U)}}(-\nu) & \dots \\
\mathtt{r}_{q^{(2,1)}}(0) & \dots & \mathtt{r}_{q^{(2,U)}}(0) & \dots & \mathtt{r}_{q^{(2,1)}}(-\nu) & \dots & \mathtt{r}_{q^{(2,U)}}(-\nu) & \dots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathtt{r}_{q^{(U,1)}}(0) & \dots & \mathtt{r}_{q^{(U,U)}}(0) & \dots & \mathtt{r}_{q^{(U,1)}}(-\nu) & \dots & \mathtt{r}_{q^{(U,U)}}(-\nu) & \dots \\
\hline
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots \\
\hline
\mathtt{r}_{q^{(1,1)}}(\nu) & \dots & \mathtt{r}_{q^{(1,U)}}(\nu) & \dots & \mathtt{r}_{q^{(1,1)}}(0) & \dots & \mathtt{r}_{q^{(1,U)}}(0) & \dots \\
\mathtt{r}_{q^{(2,1)}}(\nu) & \dots & \mathtt{r}_{q^{(2,U)}}(\nu) & \dots & \mathtt{r}_{q^{(2,1)}}(0) & \dots & \mathtt{r}_{q^{(2,U)}}(0) & \dots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathtt{r}_{q^{(U,1)}}(\nu) & \dots & \mathtt{r}_{q^{(U,U)}}(\nu) & \dots & \mathtt{r}_{q^{(U,1)}}(0) & \dots & \mathtt{r}_{q^{(U,U)}}(0) & \dots \\
\hline
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots
\end{array}\right] \tag{10.79}$$

Let $\mathbf{w}$ be the vector of noise samples at the MF output. It is important to verify that its covariance matrix is $N_0\mathbf{T}$. Then the matrix $\mathbf{T}$ is Hermitian and, assuming that it is definite positive, the Cholesky decomposition (2.174) can be applied

$$\mathbf{T} = \mathbf{L}^H\mathbf{L} \tag{10.80}$$

where $\mathbf{L}^H$ is a lower triangular matrix with positive real elements on the main diagonal.

Using (10.76) and (10.79), we find that the vector $\mathbf{y}$ satisfies the linear relation

$$\mathbf{y} = \mathbf{T}\,\mathbf{a} + \mathbf{w} \tag{10.81}$$

Once the expression (10.81) is obtained, the vector $\mathbf{a}$ can be detected by well-known techniques [20].

Applying the zero-forcing criterion, at the decision point we get the vector

$$\begin{aligned} \mathbf{z} &= \mathbf{T}^{-1}\,\mathbf{y} \\ &= \mathbf{a} + \mathbf{T}^{-1}\,\mathbf{w} \end{aligned} \tag{10.82}$$

Equation (10.82) shows that the zero-forcing criterion completely eliminates both ISI and MUI, but it may enhance the noise.

Applying instead the MSE criterion to the signal $r_C(t)$ suitably sampled, leads to the solution (see (2.229))

$$\mathbf{z} = (\mathbf{T} + N_0\mathbf{I})^{-1}\,\mathbf{y} \tag{10.83}$$

Both approaches require the inversion of a $KU \times KU$ Hermitian matrix and therefore a large computational complexity. A scheme that is computationally efficient while maintaining comparable performance is described in [21]. A MMSE method with further reduced complexity operates on single output samples, that is for $K = 1$. However, the performance is lower because it does not exploit the correlation among the different observations. For the case $K = 1$, an alternative that yields performance near the optimum ML receiver is represented by a DFE structure (see Section 16.4).

10.7 Maximum likelihood multiuser detector

Correlation matrix approach

Using the notation introduced in the previous section, the multiuser signal $r_C(t)$ is expressed as

$$r_C(t) = \sum_{u=1}^{U} s_C^{(u)}(t) + w_C(t) \tag{10.84}$$

$$= \sum_{i=0}^{K-1} \mathbf{a}_i^T \, \mathbf{q}_T(t - iT) + w_C(t) \tag{10.85}$$

The log-likelihood associated with (10.84) is [22]

$$\ell_C = -\int \left| r_C(t) - \sum_{u=1}^{U} s_C^{(u)}(t) \right|^2 dt \tag{10.86}$$

Defining the matrices

$$\mathbf{Q}_{k_1-k_2} = \int \mathbf{q}_T^*(t - k_1T) \, \mathbf{q}_T^T(t - k_2T) \, dt \tag{10.87}$$

after several steps, (10.86) can be written as

$$\ell_C = \sum_{k=0}^{K-1} \mathtt{M}_C(k) \tag{10.88}$$

where the branch metric is given by

$$\mathtt{M}_C(k) = Re \left\{ \mathbf{a}_k^H \left[\mathbf{Q}_0 \, \mathbf{a}_k + \sum_{m=1}^{\nu} 2\mathbf{Q}_m \, \mathbf{a}_{k-m} - 2\mathbf{y}_k \right] \right\} \tag{10.89}$$

having assumed that

$$\mathbf{Q}_m = \mathbf{0} \qquad |m| > \nu \tag{10.90}$$

We note that the first two terms within the brackets in (10.89) can be computed off-line.

The sequence $\{\hat{\mathbf{a}}_k\}$ that maximizes (10.88) can be obtained using the Viterbi algorithm; the complexity of this scheme is, however, exceedingly large, because it requires $O(4^{2U\nu})$ branch metric computations per detected symbol, assuming QPSK modulation.

Whitening filter approach

We now derive an alternative formulation of the ML multiuser detector; for this reason it is convenient to express the MF output using the D transform [22]. Defining

$$\mathbf{Q}(D) = \sum_{k=-\nu}^{\nu} \mathbf{Q}_k D^k \tag{10.91}$$

the MF output can be written as

$$\mathbf{y}(D) = \mathbf{Q}(D)\,\mathbf{a}(D) + \mathbf{w}(D) \tag{10.92}$$

where $\mathbf{w}(D)$ is the noisy term with matrix spectral density $N_0\mathbf{Q}(D)$. Assuming that it does not have poles on the unit circle, $\mathbf{Q}(D)$ can be factorized in the form

$$\mathbf{Q}(D) = \mathbf{F}^H(D^{-1})\,\mathbf{F}(D) \tag{10.93}$$

where $\mathbf{F}(D)$ is minimum phase; in particular, $\mathbf{F}(D)$ has the form

$$\mathbf{F}(D) = \sum_{k=0}^{\nu} \mathbf{F}_k D^k \tag{10.94}$$

where $\mathbf{F}_0$ is a lower triangular matrix. Now let $\mathbf{\Gamma}(D) = [\mathbf{F}^H(D^{-1})]^{-1}$, an anticausal filter by construction. Applying $\mathbf{\Gamma}(D)$ to $\mathbf{y}(D)$ in (10.92), we obtain

$$\begin{aligned} \mathbf{z}(D) &= \mathbf{\Gamma}(D)\,\mathbf{y}(D) \\ &= \mathbf{F}(D)\,\mathbf{a}(D) + \mathbf{w}'(D) \end{aligned} \tag{10.95}$$

where the noisy term $\mathbf{w}'(D)$ is a white Gaussian process. Consequently, in the time domain (10.95) becomes

$$\mathbf{z}_k = \sum_{m=0}^{\nu} \mathbf{F}_m\,\mathbf{a}_{k-n} + \mathbf{w}'_k \tag{10.96}$$

With reference to [23], the expression (10.96) is an extension to the multidimensional case of Forney's MLSD approach. In fact, the log-likelihood can be expressed as the sum of branch metrics defined as

$$\begin{aligned} \mathtt{M}_E(k) &= \left\| \mathbf{z}_k - \sum_{m=0}^{\nu} \mathbf{F}_m\,\mathbf{a}_{k-m} \right\|^2 \\ &= \sum_{u=1}^{U} \left| z_k^{(u)} - \sum_{i=1}^{U} (F_0^{(u,i)}\,a_k^{(i)} + \cdots + F_\nu^{(u,i)}\,a_{k-\nu}^{(i)}) \right|^2 \end{aligned} \tag{10.97}$$

We note that, as $\mathbf{F}_0$ is a lower triangular matrix, the metric has a causal dependence also with regard to the ordering of the users.

For further study on multiuser detection techniques we refer the reader to [24, 25, 26].

Bibliography

[1] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread spectrum communications handbook*. New York: McGraw-Hill, 1994.

[2] R. C. Dixon, *Spread spectrum systems*. New York: John Wiley & Sons, 3rd ed., 1994.

[3] L. B. Milstein and M. K. Simon, "Spread spectrum communications", in *The Mobile Communications Handbook* (J. D. Gibson, ed.), ch. 11, pp. 152–165, New York: CRC/IEEE Press, 1996.

[4] J. G. Proakis, *Digital communications*. New York: McGraw-Hill, 3rd ed., 1995.

[5] R. Price and P. E. Green, "A communication technique for multipath channels", *IRE Proceedings*, vol. 46, pp. 555–570, Mar. 1958.

[6] A. J. Viterbi, *CDMA: Principles of spread-spectrum communication*. Reading, MA: Addison-Wesley, 1995.

[7] R. L. Peterson, R. E. Ziemer, and D. E. Borth, *Introduction to spread spectrum communications*. Englewood Cliffs, NJ: Prentice-Hall, 1995.

[8] "Wideband CDMA", *IEEE Communications Magazine*, vol. 36, pp. 46–95, Sept. 1998.

[9] G. Cherubini and L. B. Milstein, "Performance analysis of both hybrid and frequency–hopped phase–coherent spread–spectrum system. Part I and Part II", *IEEE Trans. on Communications*, vol. 37, pp. 600–622, June 1989.

[10] A. Klein, "Data detection algorithms specially designed for the downlink of CDMA mobile radio systems", in *Proc. 1997 IEEE Vehicular Technology Conference*, Phoenix, USA, pp. 203–207, May 4–7 1997.

[11] K. Li and H. Liu, "A new blind receiver for downlink DS-CDMA communications", *IEEE Communications Letters*, vol. 3, pp. 193–195, July 1999.

[12] S. Werner and J. Lilleberg, "Downlink channel decorrelation in CDMA systems with long codes", in *Proc. 1999 IEEE Vehicular Technology Conference*, Houston, USA, pp. 1614–1617, May 16–20 1999.

[13] K. Hooli, M. Latva-aho, and M. Juntti, "Multiple access interference suppression with linear chip equalizers in WCDMA downlink receivers", in *Proc. 1999 IEEE Global Telecommunications Conference*, Rio de Janeiro, Brazil, pp. 467–471, Dec. 5–9 1999.

[14] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA", *IEEE Trans. on Communications*, vol. 42, pp. 3178–3188, Dec. 1994.

[15] S. L. Miller, "An adaptive direct-sequence code-division multiple-access receiver for multiuser interference rejection", *IEEE Trans. on Communications*, vol. 43, pp. 1746–1755, Feb./Mar./Apr. 1995.

[16] P. B. Rapajic and B. S. Vucetic, "Adaptive receiver structures for asynchronous CDMA systems", *IEEE Journal on Selected Areas in Communications*, vol. 12, pp. 685–697, May 1994.

[17] A. Klein and P. W. Baier, "Linear unbiased data estimation in mobile radio systems applying CDMA", *IEEE Journal on Selected Areas in Communications*, vol. 11, pp. 1058–1066, Sept. 1993.

[18] J. Blanz, A. Klein, M. Naßhan, and A. Steil, "Performance of a cellular hybrid C/TDMA mobile radio system applying joint detection and coherent receiver antenna diversity", *IEEE Journal on Selected Areas in Communications*, vol. 12, pp. 568–579, May 1994.

[19] G. K. Kaleh, "Channel equalization for block transmission systems", *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 110–120, Jan. 1995.

[20] A. Klein, G. K. Kaleh, and P. W. Baier, "Zero forcing and minimum mean-square-error equalization for multiuser detection in code-division multiple-access channels", *IEEE Trans. on Vehicular Technology*, vol. 45, pp. 276–287, May 1996.

[21] N. Benvenuto and G. Sostrato, "Joint detection with low computational complexity for hybrid TD-CDMA systems", *IEEE Journal on Selected Areas in Communications*, vol. 19, pp. 245–253, Jan. 2001.

[22] G. E. Bottomley and S. Chennakeshu, "Unification of MLSE receivers and extension to time-varying channels", *IEEE Trans. on Communications*, vol. 46, pp. 464–472, Apr. 1998.

[23] A. Duel-Hallen, "A family of multiuser decision feedback detectors for asynchronous code-division multiple access channels", *IEEE Trans. on Communications*, vol. 43, pp. 421–434, Feb./Mar./Apr. 1995.

[24] S. Verdù, *Multiuser detection*. Cambridge: Cambridge University Press, 1998.

[25] "Multiuser detection techniques with application to wired and wireless communications systems I", *IEEE Journal on Selected Areas in Communications*, vol. 19, Aug. 2001.

[26] "Multiuser detection techniques with application to wired and wireless communications systems II", *IEEE Journal on Selected Areas in Communications*, vol. 20, Feb. 2002.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 11

Channel codes

Forward error correction (FEC) is a widely used technique to achieve reliable data transmission. The redundancy introduced by an encoder for the transmission of data in coded form allows the decoder at the receiver to detect and partially correct errors. An alternative transmission technique, known as *automatic repeat query* or *request* (ARQ), consists in detecting the errors (usually by a *check-sum* transmitted with the data, see page 875) and requesting the retransmission of a data packet whenever it is received with errors.

The FEC technique presents two advantages with respect to the ARQ technique.

1. In systems that make use of the ARQ technique the data packets do not necessarily have to be retransmitted until they are received without errors; however, for large values of the error probability, the aggregate traffic of the link is higher.

2. In systems that make use of the FEC technique the receiver does not have to request the retransmission of data packets, thus making possible the use of a simplex link (see Section 6.13); this feature represents a strong point in many applications like TDMA and video satellite links, where a central transmitter broadcasts to receive-only terminals, which are unable to make a possible retransmission request. The FEC technique is also particularly useful in various satellite communication applications, in which the long round-trip delay of the link would cause serious traffic problems whenever the ARQ technique would be used.

We distinguish two broad classes of FEC techniques, each with numerous subclasses, employing *block codes* or *convolutional codes*.

All error correction techniques add redundancy, in the form of additional bits, to the information bits that must be transmitted. Redundancy makes the correction of errors possible and for the classes of codes considered in this chapter represents the coding overhead. The effectiveness of a coding technique is expressed in terms of the *coding gain*, G_{code}, given by the difference between the signal-to-noise ratios, in dB, that are required to achieve a certain bit error probability for transmission without and with coding (see Definition 6.2 on page 508). The overhead is expressed in terms of the *code rate*, R_c, given by the ratio between the number of information bits and the number of code bits that are transmitted. The transmission bit rate is inversely proportional to R_c, and is larger than that necessary for uncoded data. If one of the modulation techniques of Chapter 6 is employed, the modulation rate is also larger. In Chapter 12 methods to

transmit coded sequences of symbols without an increase in the modulation rate will be discussed.

For further study on the topic of error correcting codes we refer to [1, 2, 3].

11.1 System model

With reference to the model of a transmission system with coding, illustrated in Figure 6.20, we introduce some fundamental parameters.

A block code is composed of a set of vectors of given length called *code words*; the length of a code word is defined as the number of vector elements, indicated by n_0. The elements of a code word are chosen from an alphabet of q elements: if the alphabet consists of two elements, for example 0 and 1, the code is a *binary code* and we refer to the elements of each code word as *bits*; if, on the other hand, the elements of a code word are chosen from an alphabet having q elements ($q > 2$), the code is nonbinary. It is interesting to note that if q is a power of two, that is $q = 2^b$, where b is a positive integer, each q-ary element has an equivalent binary representation of b bits and therefore a nonbinary code word of length N can be mapped to a binary code word of length $n_0 = bN$.

There are 2^{n_0} possible code words in a binary code of length n_0. From these 2^{n_0} possible code words, we choose 2^{k_0} words ($k_0 < n_0$) to form a code. Thus a block of k_0 information bits is mapped to a code word of length n_0 chosen from a set of 2^{k_0} code words; the resulting block code is indicated as (n_0, k_0) code and the ratio $R_c = k_0/n_0$ is the code rate.[1]

Observation 11.1
The code rate R_c is related to the *encoder-modulator rate* R_I (6.93) by the following relation

$$R_I = \frac{k_0}{n_0}\frac{\log_2 M}{I} = R_c\frac{\log_2 M}{I} \tag{11.1}$$

where M is the number of symbols of the I-dimensional constellation adopted by the bit-mapper.

Because the number of bits per unit of time produced by the encoder is larger than that produced by the source, two transmission strategies are possible.

Transmission for a given bit rate of the information message. With reference to Figure 6.20, from the relation

$$k_0\, T_b = n_0\, T_{cod} \tag{11.2}$$

we obtain

$$\frac{1}{T_{cod}} = \frac{1}{R_c}\,\frac{1}{T_b} \tag{11.3}$$

[1] In this chapter a block code will sometimes be indicated also by the notation (n, k).

note that the bit rate at the modulator input is increased in the presence of the encoder. For a given modulator with M symbols, that is using the same bit mapper, this implies an increase of the modulation rate given by

$$\frac{1}{T'} = \frac{1}{T_{cod}} \log_2 M = \frac{1}{T} \frac{1}{R_c} \tag{11.4}$$

and therefore an increase of the bandwidth of the transmission channel by a factor $1/R_c$. Moreover, for the same transmitted power, from (6.105) in the presence of the encoder the signal-to-noise ratio becomes

$$\Gamma' = \Gamma R_c \tag{11.5}$$

i.e. it decreases by a factor R_c with respect to the case of transmission of an uncoded message. Therefore, for a given information message bit rate $1/T_b$, the system operates with a lower Γ': consequently the receiver is prone to introduce more bit errors at the decoder input. In spite of this, for a suitable choice of the code, in many cases the decoder produces a detected message $\{\hat{b}_\ell\}$ affected by fewer errors with respect to the case of transmission of an uncoded message. We note that the energy per information bit of the encoded message $\{c_m\}$ is equal to that of $\{b_\ell\}$. In fact, for a given bit mapper,

$$L'_b = \frac{k_0}{n_0} \log_2 M = R_c L_b \tag{11.6}$$

Assuming the same transmitted power, from (6.97) and (11.4) we get

$$E_{s'_{Ch}} = R_c E_{s_{Ch}} \tag{11.7}$$

Therefore (6.99) yields for the encoded message $\{c_m\}$ an energy per bit of information equal to

$$\frac{E_{s'_{Ch}}}{L'_b} = \frac{E_{s_{Ch}}}{L_b} = E_b \tag{11.8}$$

Since $\Gamma' \neq \Gamma$, a comparison between the performance of the two systems, with and without coding, is made for the same E_b/N_0. In this case the coding gain, in dB, is given by

$$10 (\log_{10} \Gamma - \log_{10} \Gamma' + \log_{10} R_c) \tag{11.9}$$

Transmission for a given modulation rate. For given transmitted power and given transmission channel bandwidth, Γ remains unchanged in the presence of the encoder. Therefore, there are three possibilities.

1. The bit rate of the information message decreases by a factor R_c and becomes

$$\frac{1}{T'_b} = \frac{1}{T_b} R_c \tag{11.10}$$

2. The source emits information bits in packets and each packet is followed by additional bits generated by the encoder, forming a code word; the resulting bits are transmitted at the rate
$$\frac{1}{T_{cod}} = \frac{1}{T_b} \tag{11.11}$$
3. A block of m information bits is mapped to a transmitted symbol using a constellation with cardinality $M > 2^m$. In this case transmission occurs without decreasing the bit rate of the information message.

In the first two cases, for the same number of bits of the information message we have an increase in the duration of the transmission by a factor $1/R_c$.

For a given bit error probability in the sequence $\{\hat{b}_\ell\}$, we expect that in the presence of coding a smaller Γ is required to achieve a certain error probability as compared to the case of transmission of an uncoded message; this reduction corresponds to the coding gain (see Definition 6.2 on page 508).

11.2 Block codes

We give the following general definition.[2]

Definition 11.1
The *Hamming distance* between two vectors $\mathbf{v}_1$ and $\mathbf{v}_2$, $d^H(\mathbf{v}_1, \mathbf{v}_2)$, is given by the number of elements in which the two vectors differ.

11.2.1 Theory of binary codes with group structure

Properties

A binary block code of length n is a subset containing M_c of the 2^n possible binary sequences of length n, also called code words. The only requirement on the code words is that they are all of the same length.

Definition 11.2
The *minimum Hamming distance* of a block code, to which we will refer in this chapter simply as the *minimum distance*, is denoted by d^H_{min} and coincides with the smallest number of positions in which any two code words differ.

An example of a block code with $n = 4$, $M_c = 4$ and $d^H_{min} = 2$ is given by (11.22).

For the binary symmetric channel model (6.90), assuming that the binary code word $\mathbf{c}$ of length n is transmitted, we observe at the receiver[3]
$$\mathbf{z} = \mathbf{c} \oplus \mathbf{e} \tag{11.12}$$

[2] The material presented in Sections 11.2 and 11.3 is largely based on lectures given at the University of California, San Diego, by Professor Jack K. Wolf [4], whom the authors gratefully acknowledge.

[3] In Figure 6.20, $\mathbf{z}$ is indicated as $\tilde{\mathbf{c}}$.

where $\oplus$ denotes the modulo 2 sum of respective vector components; for example $(0111) \oplus (0010) = (0101)$. In (11.12), $\mathbf{e}$ is the binary error vector whose generic component is equal to 1 if the channel has introduced an error in the corresponding bit of $\mathbf{c}$, and 0 otherwise. We note that $\mathbf{z}$ can assume all the 2^n possible combinations of n bits.

With reference to Figure 6.20, the function of the decoder consists in associating with each possible value $\mathbf{z}$ a code word. A commonly adopted criterion is to associate $\mathbf{z}$ with the code word $\hat{\mathbf{c}}$ that is closest according to the Hamming distance. From this code word the k_0 information bits, which form the sequence $\{\hat{b}_l\}$, are recovered by inverse mapping.

Interpreting the code words as points in an n-dimensional space where the distance between points is given by the Hamming distance, we obtain the following properties.

1. A binary block code with minimum distance d_{min}^H can *correct* all patterns of

$$t = \left\lfloor \frac{d_{min}^H - 1}{2} \right\rfloor \tag{11.13}$$

   or fewer errors, where $\lfloor x \rfloor$ denotes the integer value of x.

2. A binary block code with minimum distance d_{min}^H can *detect* all patterns of $(d_{min}^H - 1)$ or fewer errors.

3. In a *binary erasure channel*, the transmitted binary symbols are detected using a ternary alphabet {0, 1, *erasure*}; a symbol is detected as *erasure* if the reliability of a binary decision is low. In the absence of errors, a binary block code with minimum distance d_{min}^H can fill in $(d_{min}^H - 1)$ erasures.

4. Seeking a relation among n, M_c, and d_{min}^H we find that, for fixed n and odd d_{min}^H, M_c is upper bounded by[4]

$$M_c \leq M_{UB} = \left\lfloor 2^n \Big/ \left\{ 1 + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{\left\lfloor \frac{d_{min}^H - 1}{2} \right\rfloor} \right\} \right\rfloor \tag{11.15}$$

5. For fixed n and d_{min}^H, it is always possible to find a code with M_c^* words where

$$M_c^* = \left\lceil 2^n \Big/ \left\{ 1 + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{d_{min}^H - 1} \right\} \right\rceil \tag{11.16}$$

   where $\lceil x \rceil$ denotes the smallest integer greater than or equal to x. We will now consider a procedure for finding such a code.

---

[4] We recall that the number of binary sequences of length n with m 'ones' is equal to

$$\binom{n}{m} = \frac{n!}{m!(n-m)!} \tag{11.14}$$

where $n! = n(n-1)\cdots 1$.

Step 1: choose any code word of length n and exclude from future choices that word and all words that differ from it in $(d_{min}^H - 1)$ or fewer positions. The total number of words excluded from future choices is

$$N_c(n, d_{min}^H - 1) = 1 + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{d_{min}^H - 1} \tag{11.17}$$

Step i: choose a word not previously excluded and exclude from future choices all words previously excluded plus the chosen word and those that differ from it in $(d_{min}^H - 1)$ or fewer positions.

Continue this procedure until there are no more words available to choose from. At each step, if still not excluded, at most $N_c(n, d_{min}^H - 1)$ additional words are excluded; therefore after step i, when i code words have been chosen, at most $i N_c(n, d_{min}^H - 1)$ words have been excluded. Then, if $2^n / N_c(n, d_{min}^H - 1)$ is an integer, we can choose at least that number of code words; if it is not an integer, we can choose at least a number of code words equal to the next largest integer.

Definition 11.3
A *binary code with group structure* is a binary block code for which the following conditions are verified:

1. the all zero word is a code word (zero code word);
2. the modulo 2 sum of any two code words is also a code word.

Definition 11.4
The *weight* of any binary vector $\mathbf{x}$, denoted as $w(\mathbf{x})$, is the number of ones in the vector.

Property 1 of a group code. The minimum distance of the code d_{min}^H is given by

$$d_{min}^H = \min\ w(\mathbf{c}) \tag{11.18}$$

where $\mathbf{c}$ can be any non-zero code word.

Proof. The sum of any two distinct code words is a non-zero word. The weight of the resulting word is equal to the number of positions in which the two original words differ. Because two words at the minimum distance differ in d_{min}^H positions, there is a word of weight d_{min}^H. If there were a non-zero word of weight less than d_{min}^H, it would be different from the zero word in less than d_{min}^H positions.

Property 2 of a group code. If all code words in a group code are written as rows of an $M_c \times n$ matrix, then every column is either zero or consists of half zeros and half ones.

Proof. An all zero column is possible if all code words have a zero in that column. Suppose in column i there are m 1s and $(M_c - m)$ 0s. Choose one of the words with a 1 in that column and add it to all words that have a 1 in that column, including the word itself: this

operation produces m words with a 0 in that column, hence $(M_c - m) \geq m$. Now we add that word to each word that has a 0 in that column: this produces $(M_c - m)$ words with a 1 in that column, hence $(M_c - m) \leq m$. Therefore $M_c - m = m$ or $m = M_c/2$.

Corollary 11.1
From Property 2 it turns out that the number of code words M_c must be even for a binary group code.

Corollary 11.2
Excluding codes of no interest from the transmission point of view, for which all code words have a 0 in a given position, from Property 2 the *average* weight of a code word is equal to $n/2$.

Parity check matrix

Let $\mathbf{H}$ be a binary $r \times n$ matrix, which is called parity check matrix, of the form

$$\mathbf{H} = [\mathbf{A}\ \mathbf{B}] \tag{11.19}$$

where $\mathbf{B}$ is an $r \times r$ matrix with $\det[\mathbf{B}] \neq 0$, i.e. the columns of $\mathbf{B}$ are linearly independent.

A *binary parity check code* is a code consisting of all binary vectors $\mathbf{c}$ that are solutions of the equation

$$\mathbf{Hc} = \mathbf{0} \tag{11.20}$$

The matrix product in (11.20) is computed using the modulo 2 arithmetic.

Example 11.2.1
Let the matrix $\mathbf{H}$ be given by

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix} \tag{11.21}$$

There are four code words in the binary parity check code corresponding to the matrix $\mathbf{H}$; they are

$$\mathbf{c}_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \mathbf{c}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} \qquad \mathbf{c}_2 = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} \qquad \mathbf{c}_3 = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \tag{11.22}$$

Property 1 of a parity check code. A parity check code is a group code.

Proof. The all zero word is always a code word, as

$$\mathbf{H0} = \mathbf{0} \tag{11.23}$$

Suppose that $\mathbf{c}_1$ and $\mathbf{c}_2$ are code words; then $\mathbf{Hc}_1 = \mathbf{0}$ and $\mathbf{Hc}_2 = \mathbf{0}$. It follows that

$$\mathbf{H}(\mathbf{c}_1 \oplus \mathbf{c}_2) = \mathbf{Hc}_1 \oplus \mathbf{Hc}_2 = \mathbf{0} \oplus \mathbf{0} = \mathbf{0} \tag{11.24}$$

Therefore $\mathbf{c}_1 \oplus \mathbf{c}_2$ is also a code word.

Property 2 of a parity check code. The code words corresponding to the parity check matrix $\mathbf{H} = [\mathbf{A}\ \mathbf{B}]$ are identical to the code words corresponding to the parity check matrix $\tilde{\mathbf{H}} = [\mathbf{B}^{-1}\mathbf{A}, \mathbf{I}] = [\tilde{\mathbf{A}}\ \mathbf{I}]$, where $\mathbf{I}$ is the $r \times r$ identity matrix.

Proof. Let $\mathbf{c} = \begin{bmatrix} \mathbf{c}_1^{n-r} \\ \mathbf{c}_{n-r+1}^{n} \end{bmatrix}$ be a code word corresponding to the matrix $\mathbf{H} = [\mathbf{A}\ \mathbf{B}]$, where $\mathbf{c}_1^{n-r}$ are the first $(n-r)$ components of the vector and $\mathbf{c}_{n-r+1}^{n}$ are the last r components of the vector. Then

$$\mathbf{Hc} = \mathbf{Ac}_1^{n-r} \oplus \mathbf{Bc}_{n-r+1}^{n} = \mathbf{0} \tag{11.25}$$

Multiplying by $\mathbf{B}^{-1}$ we get

$$\mathbf{B}^{-1}\mathbf{Ac}_1^{n-r} \oplus \mathbf{Ic}_{n-r+1}^{n} = \mathbf{0} \tag{11.26}$$

or $\tilde{\mathbf{H}}\mathbf{c} = \mathbf{0}$.

From Property 2 we see that parity check matrices of the form $\tilde{\mathbf{H}} = [\tilde{\mathbf{A}}\ \mathbf{I}]$ are not less general than parity check matrices of the form $\mathbf{H} = [\mathbf{A}\ \mathbf{B}]$, where $\det[\mathbf{B}] \neq 0$. In general, we can consider any $r \times n$ matrix as a parity check matrix, provided that some set of r columns has a non-zero determinant. If we are not concerned with the order by which the elements of a code word are transmitted, then such a code would be equivalent to a code formed by a parity check matrix of the form

$$\mathbf{H} = [\mathbf{A}\ \mathbf{I}] \tag{11.27}$$

The form of the matrix (11.27) is called *canonical or systematic form*. We assume that the last r columns of $\mathbf{H}$ have a non-zero determinant and therefore that the parity check matrix can be expressed in canonical form.

Property 3 of a parity check code. There are exactly $2^{n-r} = 2^k$ code words in a parity check code.

Proof. Referring to the proof of Property 2, we find that

$$\mathbf{c}_{n-r+1}^{n} = \mathbf{Ac}_1^{n-r} \tag{11.28}$$

For each of the $2^{n-r} = 2^k$ possible binary vectors $\mathbf{c}_1^{n-r}$ it is possible to compute the corresponding vector $\mathbf{c}_{n-r+1}^{n}$. Each of these code words is unique as all of them differ in the first $(n-r) = k$ positions. Assume that there are more than 2^k code words; then at least two will agree in the first $(n-r) = k$ positions. But from (11.28) we find that these two code words also agree in the last r positions and therefore they are identical.

The code words have the following structure

$$\mathbf{c} = [m_0 \dots m_{k-1}, p_0 \dots p_{r-1}]^T \tag{11.29}$$

where the first $k = (n - r)$ bits are called *information bits* and the last r bits are called *parity check bits*. As mentioned in Section 11.1, a parity check code that has code words of length n that are obtained by encoding k information bits is an (n, k) code.

Property 4 of a parity check code. A code word of weight w exists if and only if the modulo 2 sum of w columns of $\mathbf{H}$ equals $\mathbf{0}$.

Proof. $\mathbf{c}$ is a code word if and only if $\mathbf{Hc} = \mathbf{0}$. Let $\mathbf{h}_i$ be the i-th column of $\mathbf{H}$ and let c_j be the j-th component of $\mathbf{c}$. Therefore, if $\mathbf{c}$ is a code word, then

$$\sum_{j=1}^{n} \mathbf{h}_j c_j = \mathbf{0} \tag{11.30}$$

If $\mathbf{c}$ is a code word of weight w, then there are exactly w non-zero components of $\mathbf{c}$, for example $c_{j_1}, c_{j_2}, \dots, c_{j_w}$. Consequently $\mathbf{h}_{j_1} \oplus \mathbf{h}_{j_2} \oplus \dots \oplus \mathbf{h}_{j_w} = \mathbf{0}$, thus a code word of weight w implies that the sum of w columns of $\mathbf{H}$ equals $\mathbf{0}$. Conversely, if $\mathbf{h}_{j_1} \oplus \mathbf{h}_{j_2} \oplus \dots \oplus \mathbf{h}_{j_w} = \mathbf{0}$ then $\mathbf{Hc} = \mathbf{0}$, where $\mathbf{c}$ is a binary vector with elements equal to 1 in positions $j_1, j_2, \dots, j_w$.

From Property 1 of a group code and also from Properties 1 and 4 of a parity check code we obtain the following property.

Property 5 of a parity check code. A parity check code has minimum distance d_{min}^H if some modulo 2 sum of d_{min}^H columns of $\mathbf{H}$ is equal to $\mathbf{0}$, but no modulo 2 sum of fewer than d_{min}^H columns of $\mathbf{H}$ is equal to $\mathbf{0}$.

Property 5 may be considered as the fundamental property of parity check codes, as it forms the basis for the design of almost all such codes. An important exception is constituted by low-density parity check codes, which will be discussed in Section 11.7. A limit on the number of parity check bits required for a given block length n and given d_{min}^H derives directly from this property.

Property 6 of a parity check code. A binary parity check code exists of block length n and minimum distance d_{min}^H, having no more than r^* parity check bits, where

$$r^* = \left\lfloor \log_2 \left(\sum_{i=0}^{d_{min}^H - 2} \binom{n-1}{i} \right) \right\rfloor + 1 \tag{11.31}$$

Proof. The proof derives from the following exhaustive construction procedure of the parity check matrix of the code.

Step 1: choose as the first column of $\mathbf{H}$ any non-zero vector with r^* components.

Step 2: choose as the second column of $\mathbf{H}$ any non-zero vector different from the first.

Step 3: choose as the i-th column of $\mathbf{H}$ any vector distinct from all vectors obtained by modulo 2 sum of $(d^H_{min} - 2)$ or fewer previously chosen columns.

Clearly such a procedure will result in a matrix $\mathbf{H}$ where no set of $(d^H_{min} - 1)$ or fewer columns of $\mathbf{H}$ sum to $\mathbf{0}$. However, we must show that we can indeed continue this process for n columns. After applying this procedure for $(n-1)$ columns, there will be at most

$$N_c(n-1, d^H_{min} - 2) = 1 + \binom{n-1}{1} + \binom{n-1}{2} + \cdots + \binom{n-1}{d^H_{min} - 2} \tag{11.32}$$

distinct vectors that are forbidden for the choice of the last column, but there are 2^{r^*} vectors to choose from; observing (11.31) and (11.32) we get $2^{r^*} > N_c(n-1, d^H_{min} - 2)$. Thus n columns can always be chosen where no set of $(d^H_{min} - 1)$ or fewer columns sums to zero. From Property 5, the code therefore has minimum distance at least d^H_{min}.

Code generator matrix

Using (11.28), we can write

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1^{n-r} \\ \mathbf{c}_{n-r+1}^{n} \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ \mathbf{A} \end{bmatrix} \mathbf{c}_1^{n-r} = \mathbf{G}^T \mathbf{c}_1^{n-r} \tag{11.33}$$

where $\mathbf{G} = [\mathbf{I}\ \mathbf{A}^T]$ is a $k \times n$ binary matrix, and $\mathbf{I}$ is the $k \times k$ identity matrix. Taking the transpose of (11.33), we obtain

$$\mathbf{c}^T = (\mathbf{c}_1^{n-r})^T\ \mathbf{G} \tag{11.34}$$

thus the code words, considered now as row vectors, are given as all linear combinations of the rows of $\mathbf{G}$, which is called the *generator matrix* of the code. A parity check code can be specified by giving its parity check matrix $\mathbf{H}$ or its generator matrix $\mathbf{G}$.

Example 11.2.2
Consider the parity check code (7,4) with the parity check matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} = [\mathbf{A}\ \mathbf{I}] \tag{11.35}$$

Expressing a general code word according to (11.29), to every 4 information bits 3 parity check bits are added, related to the information bits by the equations (see (11.28))

$$\begin{aligned} p_0 &= m_0 \oplus m_1 \oplus m_3 \\ p_1 &= m_0 \oplus m_1 \oplus m_2 \\ p_2 &= m_0 \oplus m_2 \oplus m_3 \end{aligned} \tag{11.36}$$

The generator matrix of this code is given by

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} = [\mathbf{I}\ \mathbf{A}^T] \tag{11.37}$$

There are 16 code words consisting of all linear combinations of the rows of $\mathbf{G}$. By inspection, we find that the minimum weight of a non-zero code word is 3; hence, from (11.18) the code has $d_{min}^H = 3$ and therefore is a single error correcting code.

Decoding of binary parity check codes

Conceptually the simplest method for decoding a block code is to compare the received block of n bits with each code word and choose that code word that differs from the received word in the minimum number of positions; in case several code words satisfy this condition, choose amongst them at random. Although the simplest conceptually, the described method is out of the question practically because we usually employ codes with very many code words. It is, however, instructive to consider the application of this method, suitably modified, to decode group codes.

Cosets

The 2^n possible binary sequences of length n are partitioned into 2^r sets, called *cosets*, by a group code with $2^k = 2^{n-r}$ code words; this partitioning is done as follows:

Step 1: choose the first set as the set of code words $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{2^k}$.

Step 2: choose any vector, say, $\boldsymbol{\eta}_2$, that is not a code word; then choose the second set as $\mathbf{c}_1 \oplus \boldsymbol{\eta}_2, \mathbf{c}_2 \oplus \boldsymbol{\eta}_2, \ldots, \mathbf{c}_{2^k} \oplus \boldsymbol{\eta}_2$.

Step i: choose any vector, say, $\boldsymbol{\eta}_i$, not included in any previous set; choose the i-th set, i.e. coset, as $\mathbf{c}_1 \oplus \boldsymbol{\eta}_i, \mathbf{c}_2 \oplus \boldsymbol{\eta}_i, \ldots, \mathbf{c}_{2^k} \oplus \boldsymbol{\eta}_i$. The partitioning continues until all 2^n vectors are used.

Note that each coset contains 2^k vectors; if we show that no vector can appear in more than one coset, we will have demonstrated that there are $2^r = 2^{n-k}$ cosets.

Property 1 of cosets. Every binary vector of length n appears in one and only one coset.

Proof. Every vector appears in at least one coset as the partitioning stops only when all vectors are used. Suppose that a vector appeared twice in one coset; then for some value of the index i we have $\mathbf{c}_{j_1} \oplus \boldsymbol{\eta}_i = \mathbf{c}_{j_2} \oplus \boldsymbol{\eta}_i$, or $\mathbf{c}_{j_1} = \mathbf{c}_{j_2}$, that is a contradiction as all code words are unique. Suppose that a vector appears in two cosets; then $\mathbf{c}_{j_1} \oplus \boldsymbol{\eta}_{i_1} = \mathbf{c}_{j_2} \oplus \boldsymbol{\eta}_{i_2}$, where we assume $i_2 > i_1$. Then $\boldsymbol{\eta}_{i_2} = \mathbf{c}_{j_1} \oplus \mathbf{c}_{j_2} \oplus \boldsymbol{\eta}_{i_1} = \mathbf{c}_{j_3} \oplus \boldsymbol{\eta}_{i_1}$, that is a contradiction as $\boldsymbol{\eta}_{i_2}$ would have appeared in a previous coset, against the hypothesis.

Example 11.2.3
Consider partitioning the 2^4 binary vectors of length 4 into cosets using the group code with code words 0000, 0011, 1100, 1111, as follows:

$$\begin{array}{rcl} & 0\,0\,0\,0 \quad 0\,0\,1\,1 \quad 1\,1\,0\,0 \quad 1\,1\,1\,1 \\ \boldsymbol{\eta}_2 & = & 0\,0\,0\,1 \quad 0\,0\,1\,0 \quad 1\,1\,0\,1 \quad 1\,1\,1\,0 \\ \boldsymbol{\eta}_3 & = & 0\,1\,1\,1 \quad 0\,1\,0\,0 \quad 1\,0\,1\,1 \quad 1\,0\,0\,0 \\ \boldsymbol{\eta}_4 & = & 1\,0\,1\,0 \quad 1\,0\,0\,1 \quad 0\,1\,1\,0 \quad 0\,1\,0\,1 \end{array} \tag{11.38}$$

The vectors $\boldsymbol{\eta}_1 = \mathbf{0}, \boldsymbol{\eta}_2, \boldsymbol{\eta}_3, \ldots, \boldsymbol{\eta}_{2^r}$, are called *coset leaders*; the partitioning (11.38) is called *coset table* or *decoding table*.

Property 2 of cosets. Suppose that instead of choosing $\boldsymbol{\eta}_i$ as the coset leader of the i-th coset, we choose another element of that coset; the new coset formed by using this new coset leader contains exactly the same vectors as the old coset.

Proof. Assume that the new coset leader is $\boldsymbol{\eta}_i \oplus \mathbf{c}_{j_1}$, and that $\mathbf{z}$ is an element of the new coset; then $\mathbf{z} = \boldsymbol{\eta}_i \oplus \mathbf{c}_{j_1} \oplus \mathbf{c}_{j_2} = \boldsymbol{\eta}_i \oplus \mathbf{c}_{j_3}$, so $\mathbf{z}$ is an element of the old coset. As the new and the old cosets both contain 2^k vectors and all vectors in a coset are unique, every element of the new coset belongs to the old coset and vice versa.

Example 11.2.4
Suppose that in the previous example we had chosen the third coset leader as 0100; then the table (11.38) would be

$$\begin{array}{rcl} & 0\,0\,0\,0 \quad 0\,0\,1\,1 \quad 1\,1\,0\,0 \quad 1\,1\,1\,1 \\ \boldsymbol{\eta}_2 & = & 0\,0\,0\,1 \quad 0\,0\,1\,0 \quad 1\,1\,0\,1 \quad 1\,1\,1\,0 \\ \boldsymbol{\eta}_3 & = & 0\,1\,0\,0 \quad 0\,1\,1\,1 \quad 1\,0\,0\,0 \quad 1\,0\,1\,1 \\ \boldsymbol{\eta}_4 & = & 1\,0\,1\,0 \quad 1\,0\,0\,1 \quad 0\,1\,1\,0 \quad 0\,1\,0\,1 \end{array} \tag{11.39}$$

Two conceptually simple decoding methods

Assume that each *coset leader* is chosen as the *minimum weight vector in its coset*; in case several vectors in a coset have the same minimum weight, choose any one of them as the coset leader.

Then a second method of decoding, using the decoding table, is as follows:

Step 1: locate the received vector in the coset table.

Step 2: choose the code word that appears as the first vector in the column containing the received vector.

Proposition 11.1
Decoding using the decoding table decodes to the closest code word to the received word; in case several code words are at the same smallest distance from the received word, it decodes to one of these closest words.

Proof. Assume that the received word is the j-th vector in the i-th coset. The received word, given by $\mathbf{z} = \mathbf{c}_j \oplus \boldsymbol{\eta}_i$, is corrected to the code word $\mathbf{c}_j$ and the distance between the received word and the j-th code word is $w(\boldsymbol{\eta}_i)$. Suppose that another code word, say, $\mathbf{c}_k$, is closer to the received vector: then

$$w(\mathbf{c}_k \oplus \mathbf{c}_j \oplus \boldsymbol{\eta}_i) < w(\boldsymbol{\eta}_i) \tag{11.40}$$

or

$$w(\mathbf{c}_\ell \oplus \boldsymbol{\eta}_i) < w(\boldsymbol{\eta}_i) \tag{11.41}$$

but this cannot be as $w(\boldsymbol{\eta}_i)$ is assumed to be the minimum weight vector in its coset and $\mathbf{c}_\ell \oplus \boldsymbol{\eta}_i$ is in that coset.

We note that the coset leaders determine the only error patterns that can be corrected by the code. Coset leaders, moreover, have many other interesting properties: for example, if a code has minimum distance d^H_{min}, all binary n-tuple of weight less than or equal to $\left\lfloor \frac{d^H_{min}-1}{2} \right\rfloor$ are coset leaders.

Definition 11.5
A code for which coset leaders are all vectors of weight t or less, and no others, is called a *perfect* t-error correcting code. A code for which coset leaders are all vectors of weight t or less, and some vectors of weight $t+1$ but not all, and no others, is called *quasi-perfect* t-error correcting code.

The *perfect binary codes* are:

1. codes given by the repetition of n bits, with n odd: these codes contain only two code words, 000 ... 0 (all zeros) and 111 ... 1 (all ones), and correct $t = (n-1)/2$ errors ($d^H_{min} = n$);
2. Hamming codes: these codes correct $t = 1$ errors ($d^H_{min} = 3$) and have $n = 2^r - 1$, $k = n - r$, $r > 1$; the columns of the matrix $\mathbf{H}$ are given by all non-zero vectors of length r;
3. Golay code: $t = 3$ ($d^H_{min} = 7$), $n = 23$, $k = 12$, $r = 11$.

The following modification of the decoding method dealt with in this section will be useful later on:

Step $1'$*:* locate the received vector in the coset table and identify the coset leader of the coset containing that vector.

Step $2'$*:* add the coset leader to the received vector to find the decoded code word.

Syndrome decoding

A third method of decoding is based on the concept of syndrome. Among the methods described in this section, syndrome decoding is the only method of practical value for a code with a large number of code words.

Definition 11.6
For any parity check matrix $\mathbf{H}$, we define the *syndrome* $\mathbf{s}(\mathbf{z})$ of a binary vector $\mathbf{z}$ of length n as

$$\mathbf{s}(\mathbf{z}) = \mathbf{H}\mathbf{z} \tag{11.42}$$

We note that the syndrome is a vector of length r, whereas $\mathbf{z}$ is a vector of length n. Therefore many vectors will have the same syndrome. All code words have an all zero syndrome and these are the only vectors with this property. This property of the code words is a special case of the following:

Property 3 of cosets. All vectors in the same coset have the same syndrome; vectors in different cosets have distinct syndromes.

Proof. Assume that $\mathbf{z}_1$ and $\mathbf{z}_2$ are in the same coset, say, the i-th: then $\mathbf{z}_1 = \boldsymbol{\eta}_i \oplus \mathbf{c}_{j_1}$ and $\mathbf{z}_2 = \boldsymbol{\eta}_i \oplus \mathbf{c}_{j_2}$. Moreover $\mathbf{s}(\mathbf{z}_1) = \mathbf{H}\mathbf{z}_1 = \mathbf{H}(\boldsymbol{\eta}_i \oplus \mathbf{c}_{j_1}) = \mathbf{H}\boldsymbol{\eta}_i \oplus \mathbf{H}\mathbf{c}_{j_1} = \mathbf{H}\boldsymbol{\eta}_i \oplus \mathbf{0} = \mathbf{s}(\boldsymbol{\eta}_i)$. Similarly $\mathbf{s}(\mathbf{z}_2) = \mathbf{s}(\boldsymbol{\eta}_i)$, so $\mathbf{s}(\mathbf{z}_1) = \mathbf{s}(\mathbf{z}_2) = \mathbf{s}(\boldsymbol{\eta}_i)$: this proves the first part of the property.

Now assume that $\mathbf{z}_1$ and $\mathbf{z}_2$ are in different cosets, say, the i_1-th and i_2-th: then $\mathbf{z}_1 = \boldsymbol{\eta}_{i_1} \oplus \mathbf{c}_{j_1}$ and $\mathbf{z}_2 = \boldsymbol{\eta}_{i_2} \oplus \mathbf{c}_{j_2}$, so $\mathbf{s}(\mathbf{z}_1) = \mathbf{s}(\boldsymbol{\eta}_{i_1})$ and $\mathbf{s}(\mathbf{z}_2) = \mathbf{s}(\boldsymbol{\eta}_{i_2})$. If $\mathbf{s}(\mathbf{z}_1) = \mathbf{s}(\mathbf{z}_2)$ then $\mathbf{s}(\boldsymbol{\eta}_{i_1}) = \mathbf{s}(\boldsymbol{\eta}_{i_2})$, which implies $\mathbf{H}\boldsymbol{\eta}_{i_1} = \mathbf{H}\boldsymbol{\eta}_{i_2}$. Consequently $\mathbf{H}(\boldsymbol{\eta}_{i_1} \oplus \boldsymbol{\eta}_{i_2}) = \mathbf{0}$, or $\boldsymbol{\eta}_{i_1} \oplus \boldsymbol{\eta}_{i_2}$ is a code word, say $\mathbf{c}_{j_3}$. Then $\boldsymbol{\eta}_{i_2} = \boldsymbol{\eta}_{i_1} \oplus \mathbf{c}_{j_3}$, which implies that $\boldsymbol{\eta}_{i_1}$ and $\boldsymbol{\eta}_{i_2}$ are in the same coset, that is a contradiction. Thus the assumption that $\mathbf{s}(\mathbf{z}_1) = \mathbf{s}(\mathbf{z}_2)$ is incorrect.

From Property 3 we see that there is a one-to-one relation between cosets and syndromes; this leads to the *third method of decoding*, which proceeds as follows:

Step $1''$*:* compute the syndrome of the received vector; this syndrome identifies the coset in which the received vector is. Identify then the leader of that coset.

Step $2''$*:* add the coset leader to the received vector to find the decoded code word.

Example 11.2.5
Consider the parity check matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \tag{11.43}$$

The coset leaders and their respective syndromes obtained using (11.42) are reported in Table 11.1.

Suppose that the vector $\mathbf{z} = 000111$ is received. To decode we first compute the syndrome $\mathbf{H}\mathbf{z} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$, then by Table 11.1 we identify the coset leader as $\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$, and obtain the decoded code word (1 0 0 1 1 1).

Table 11.1 Coset leaders and respective syndromes for Example 11.2.5.

Coset leader	Syndrome
000000	000
000001	001
000010	010
000100	100
001000	011
010000	101
100000	111
100001	110

The advantage of syndrome decoding over the other decoding methods previously described is that there is no need to memorize the entire decoding table at the receiver. The first part of Step $1''$, namely computing the syndrome, is trivial. The second part of Step $1''$, namely identifying the coset leader corresponding to that syndrome, is the difficult part of the procedure; in general it requires a RAM with 2^r memory locations, addressed by the syndrome of r bits and containing the coset leaders of n bits. Overall the memory bits are $n2^r$.

There is also an algebraic method to identify the coset leader. In fact, this problem is equivalent to finding the minimum set of columns of the parity check matrix which sum to the syndrome. In other words, we must find the vector $\mathbf{z}$ of minimum weight such that $\mathbf{Hz} = \mathbf{s}$.

For a single error correcting Hamming code, all coset leaders are of weight 1 or 0, so a non-zero syndrome corresponds to a single column of $\mathbf{H}$ and the correspondence between syndrome and coset leader is simple. For a code with coset leaders of weight 0, 1, or 2, the syndrome is either 0, a single column of $\mathbf{H}$, or the sum of two columns, etc.

For a particular class of codes that will be considered later, the structure of the construction of $\mathbf{H}$ will allow identification of the coset leader starting from the syndrome by using algebraic procedures. In general, each class of codes leads to a different technique to perform this task.

Property 7 of parity check codes. There are exactly 2^r correctable error vectors for a parity check code with r parity check bits.

Proof. Correctable error vectors are given by the coset leaders and there are $2^{n-k} = 2^r$ of them, all of which are distinct. On the other hand, there are 2^r distinct syndromes and each corresponds to a correctable error vector.

For a binary symmetric channel (see Definition 6.1) we should correct all error vectors of weight i, $i = 0, 1, 2, \ldots$, until we exhaust the capability of the code. Specifically, we should try to use a *perfect code* or a *quasi-perfect code*. For a quasi-perfect t-error correcting code, the coset leaders consist of all error vectors of weight $i = 0, 1, 2, \ldots, t$, and some vectors of weight $t + 1$.

Nonbinary parity check codes are discussed in Appendix 11.A.

11.2.2 Fundamentals of algebra

The calculation of parity check bits from information bits involves solving linear equations. This procedure is particularly easy for binary codes since we use modulo-2 arithmetic. An obvious question is whether or not the concepts of the previous section generalize to codes with symbols taken from alphabets with a larger cardinality, say, alphabets with q symbols. We will see that the answer can be yes or no according to the value of q; furthermore, even if the answer is yes we might not be able to use modulo q arithmetic.

Consider the equation for the unknown x

$$ax = b \tag{11.44}$$

where a and b are known coefficients, and all values are from the finite alphabet $\{0, 1, 2, \ldots, q-1\}$. First, we need to introduce the concept of multiplication, which is normally given in the form of a *multiplication table*, as the one given in Table 11.2 for the three elements $\{0, 1, 2\}$.

Table 11.2 allows us to solve (11.44) for any values of a and b, except $a = 0$. For example, the solution to equation $2x = 1$ is $x = 2$, as from the multiplication table we find $2 \cdot 2 = 1$.

Let us now consider the case of an alphabet with four elements. A multiplication table for the four elements $\{0, 1, 2, 3\}$, resulting from the modulo 4 arithmetic, is given in Table 11.3.

Note that the equation $2x = 2$ has two solutions, $x = 1$ and $x = 3$, and equation $2x = 1$ has no solution. It is possible to construct a multiplication table that allows the equation (11.44) to be solved uniquely for x, provided that $a \neq 0$, as shown in Table 11.4.

Table 11.2 Multiplication table for 3 elements (modulo 3 arithmetic).

$\cdot$	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

Table 11.3 Multiplication table for an alphabet with four elements (modulo 4 arithmetic).

$\cdot$	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	0	2
3	0	3	2	1

Table 11.4 Multiplication table for an alphabet with four elements.

·	0	1	2	3
0	0	0	0	0
1	0	1	2	3
2	0	2	3	1
3	0	3	1	2

Note that Table 11.4 is not obtained using modulo 4 arithmetic. For example, $2x = 3$ has the solution $x = 2$, and $2x = 1$ has the solution $x = 3$.

Modulo q arithmetic

Consider the elements $\{0, 1, 2, \ldots, q-1\}$, where q is a positive integer larger than or equal to 2. We define two operations for combining pairs of elements from this set. The first, denoted by $\oplus$, is called *modulo q addition* and is defined as

$$c = a \oplus b = \begin{cases} a+b & \text{if } 0 \leq a+b < q \\ a+b-q & \text{if } a+b \geq q \end{cases} \tag{11.45}$$

Here $a+b$ is the ordinary addition operation for integers that may produce an integer not in the set. In this case q is subtracted from $a+b$ and $a+b-q$ is always an element in the set $\{0, 1, 2, \ldots, q-1\}$. The second operation, denoted by $\otimes$, is called *modulo q multiplication* and is defined as

$$d = a \otimes b = \begin{cases} ab & \text{if } 0 \leq ab < q \\ ab - \left\lfloor \dfrac{ab}{q} \right\rfloor q & \text{if } ab \geq q \end{cases} \tag{11.46}$$

Note that $ab - \left\lfloor \frac{ab}{q} \right\rfloor q$, is the *remainder* or *residue* of the division of ab by q, and is always an integer in the set $\{0, 1, 2, \ldots, q-1\}$. Often we will omit the notation $\otimes$ and write $a \otimes b$ simply as ab.

We recall that special names are given to sets which possess certain properties with respect to operations. Consider the general set G that contains the elements $\{\alpha, \beta, \gamma, \delta, \ldots\}$, and two operations for combining elements from the set. We denote the first operation $\triangle$ (addition), and the second operation $\Diamond$ (multiplication). Often we will omit the notation $\Diamond$ and write $a \Diamond b$ simply as ab. The properties we are interested in are:

1. *Existence of additive identity*. For every $\alpha \in G$, there exists an element $\emptyset \in G$, called *additive identity*, such that $\alpha \triangle \emptyset = \emptyset \triangle \alpha = \alpha$.

2. *Existence of additive inverse*. For every $\alpha \in G$, there exists an element $\beta \in G$, called *additive inverse of* α, and indicated by $-\alpha$, such that $\alpha \triangle \beta = \beta \triangle \alpha = \emptyset$.

3. *Additive closure*. For every $\alpha, \beta \in G$, not necessarily distinct, $\alpha \triangle \beta \in G$.

4. *Additive associative law*. For every $\alpha, \beta, \gamma \in G$, $\alpha \triangle (\beta \triangle \gamma) = (\alpha \triangle \beta) \triangle \gamma$.

5. *Additive commutative law*. For every $\alpha, \beta \in G$, $\alpha \triangle \beta = \beta \triangle \alpha$.

6. *Multiplicative closure*. For every $\alpha, \beta \in G$, not necessarily distinct, $\alpha \diamondsuit \beta \in G$.

7. *Multiplicative associative law*. For every $\alpha, \beta, \gamma \in G$, $\alpha \diamondsuit (\beta \diamondsuit \gamma) = (\alpha \diamondsuit \beta) \diamondsuit \gamma$.

8. *Distributive law*. For every $\alpha, \beta, \gamma \in G$, $\alpha \diamondsuit (\beta \triangle \gamma) = (\alpha \diamondsuit \beta) \triangle (\alpha \diamondsuit \gamma)$ and $(\alpha \triangle \beta) \diamondsuit \gamma = (\alpha \diamondsuit \gamma) \triangle (\beta \diamondsuit \gamma)$.

9. *Multiplicative commutative law*. For every $\alpha, \beta \in G$, $\alpha \diamondsuit \beta = \beta \diamondsuit \alpha$.

10. *Existence of multiplicative identity*. For every $\alpha \in G$, there exists an element $I \in G$, called *multiplicative identity*, such that $\alpha \diamondsuit I = I \diamondsuit \alpha = \alpha$.

11. *Existence of multiplicative inverse*. For every $\alpha \in G$, except the element $\emptyset$, there exists an element $\delta \in G$, called *multiplicative inverse of* α, and indicated with α^{-1}, such that $\alpha \diamondsuit \delta = \delta \diamondsuit \alpha = I$.

Any set G for which Properties 1−4 hold is called a *group with respect to* $\triangle$. If G has a finite number of elements, then G is called *finite group* and the number of elements of G is called the *order of* G.

Any set G for which Properties 1−5 hold is called an *Abelian group with respect to* $\triangle$.

Any set G for which Properties 1−8 hold is called a *ring with respect to the operations* $\triangle$ *and* $\diamondsuit$.

Any set G for which Properties 1−9 hold is called a *commutative ring with respect to the operations* $\triangle$ *and* $\diamondsuit$.

Any set G for which Properties 1−10 hold is called a *commutative ring with identity*.

Any set G for which Properties 1−11 hold is called a *field*.

It can be seen that the set $\{0, 1, 2, \ldots, q-1\}$ is a commutative ring with identity with respect to the operations of addition $\oplus$ defined in (11.45) and multiplication $\otimes$ defined in (11.46). We will show by the next three properties that this set satisfies also Property 11 if and only if q is a prime: in other words, we will show that the set $\{0, 1, 2, \ldots, q-1\}$ is a *field* with respect to the modulo q addition and modulo q multiplication if and only if q is a *prime*.

Finite fields are called *Galois fields*; a field of q elements is usually denoted as $GF(q)$.

Property 11a of modulo q arithmetic. If q is not a prime, each factor of q (less than q and greater than 1) does not have a multiplicative inverse.

Proof. Let $q = ab$, where $1 < a, b < q$; then, observing (11.46), $a \otimes b = 0$. Assume that a has a multiplicative inverse a^{-1}; then $a^{-1} \otimes (a \otimes b) = a^{-1} \otimes 0 = 0$. Now, from $a^{-1} \otimes (a \otimes b) = 0$ it is $1 \otimes b = 0$; this implies $b = 0$, which is a contradiction as $b > 1$. Similarly we show that b does not have a multiplicative inverse.

Property 11b of modulo q arithmetic. If q is a prime and $a \otimes b = 0$, then $a = 0$, or $b = 0$, or $a = b = 0$.

Proof. Assume $a \otimes b = 0$ and $a, b > 0$; then $ab = Kq$, where $K < \min(a, b)$. If $1 < a \leq q - 1$ and a has no factors in common with q, then it must divide K; but this is impossible as $K < \min(a, b)$. The only other possibility is that $a = 1$, but then $a \otimes b \neq 0$ as $ab < q$.

Property 11c of modulo q arithmetic. If q is a prime, all non-zero elements of the set $\{0, 1, 2, \ldots, q - 1\}$ have multiplicative inverse.

Proof. Assume the converse, that is the element j, with $1 \leq j \leq q - 1$, does not have a multiplicative inverse; then there must be two distinct elements $a, b \in \{0, 1, 2, \ldots, q - 1\}$ such that $a \otimes j = b \otimes j$. This is a consequence of the fact that the product $i \otimes j$ can only assume values in the set $\{0, 2, 3, \ldots, q - 1\}$, as by assumption $i \otimes j \neq 1$; then

$$(a \otimes j) \oplus (q - (b \otimes j)) = 0 \tag{11.47}$$

On the other hand, $q - (b \otimes j) = (q - b) \otimes j$, and

$$(a \oplus (q - b)) \otimes j = 0 \tag{11.48}$$

But $j \neq 0$ and consequently, by Property 11b, we have $a \oplus (q - b) = 0$. This implies $a = b$, which is a contradiction.

Definition 11.7
An *ideal* I is a subset of elements of a ring R such that:

1. I is a subgroup of the additive group R, that is the elements of I form a group with respect to the addition defined in R;
2. for any element a of I and any element r of R, ar and ra are in I.

Polynomials with coefficients from a field

We consider the set of polynomials in one variable; as this set is interesting in two distinct applications, to avoid confusion we will use a different notation for the two cases.

The first application permits to extend our knowledge of finite fields. We have seen in Section 11.2.2 how to construct a field with a prime number of elements. Polynomials allow us to construct fields in which the number of elements is given by a power of a prime; for this purpose we will use polynomials in the variable y.

The second application introduces an alternative method to describe code words. We will consider *cyclic codes*, a subclass of parity check codes, and in this context will use polynomials in the single variable x.

Consider any two polynomials with coefficients from the set $\{0, 1, 2, \ldots, p - 1\}$, where p is a prime:

$$\begin{aligned} g(y) &= g_0 + g_1 y + g_2 y^2 + \cdots + g_m y^m \\ f(y) &= f_0 + f_1 y + f_2 y^2 + \cdots + f_n y^n \end{aligned} \tag{11.49}$$

We assume that $g_m \neq 0$ and $f_n \neq 0$. We define m as *degree of the polynomial* $g(y)$, and we write $m = \deg(g(y))$; in particular, if $g(y) = a$, $a \in \{0, 1, 2, \ldots, p-1\}$, we say that $\deg(g(y)) = 0$. Similarly, it is $n = \deg(f(y))$. If $g_m = 1$, we say that $g(y)$ is a *monic polynomial*.

Assume $m \leq n$: then the *addition among polynomials* is defined as

$$f(y) + g(y) = (f_0 \oplus g_0) + (f_1 \oplus g_1)y + (f_2 \oplus g_2)y^2 + \cdots + (f_m \oplus g_m)\, y^m + \cdots + f_n y^n \tag{11.50}$$

Example 11.2.6
Let $p = 5$, $f(y) = 1+3y+2y^4$, and $g(y) = 4+3y+3y^2$; then $f(y)+g(y) = y+3y^2+2y^4$.

Note that

$$\deg(f(y) + g(y)) \leq \max(\deg(f(y)), \deg(g(y)))$$

Multiplication among polynomials is defined as usual

$$f(y)\, g(y) = d_0 + d_1 y + \cdots + d_{m+n}\, y^{m+n} \tag{11.51}$$

where the arithmetic to perform operations with the various coefficients is modulo p,

$$d_i = (f_0 \otimes g_i) \oplus (f_1 \otimes g_{i-1}) \oplus \cdots \oplus (f_{i-1} \otimes g_1) \oplus (f_i \otimes g_0) \tag{11.52}$$

Example 11.2.7
Let $p = 2$, $f(y) = 1 + y + y^3$, and $g(y) = 1 + y^2 + y^3$; then $f(y)\, g(y) = 1 + y + y^2 + y^3 + y^4 + y^5 + y^6$.

Note that

$$\deg(f(y)\, g(y)) = \deg(f(y)) + \deg(g(y))$$

Definition 11.8
If $f(y)\, g(y) = d(y)$, we say that $f(y)$ divides $d(y)$, and $g(y)$ divides $d(y)$. We say that $p(y)$ is an *irreducible polynomial* if and only if, assuming another polynomial $a(y)$ divides $p(y)$, then $a(y) = a \in \{0, 1, \ldots, p-1\}$ or $a(y) = k\, p(y)$, with $k \in \{0, 1, \ldots, p-1\}$.

The concept of an irreducible polynomial plays the same role in the theory of polynomials as does the concept of a prime number in the number theory.

The concept of modulo in the arithmetic of polynomials

We define a modulo arithmetic for polynomials, analogously to the modulo q arithmetic for integers. We choose a polynomial $q(y) = q_0 + q_1 y + \cdots + q_m y^m$ with coefficients that are elements of the field $\{0, 1, 2, \ldots, p-1\}$. We consider the set $\mathcal{P}$ of all polynomials of degree less than m with coefficients from the field $\{0, 1, 2, \ldots, p-1\}$; this set consists of p^m polynomials.

Example 11.2.8
Let $p = 2$ and $q(y) = 1 + y + y^3$; then the set $\mathcal{P}$ consists of 2^3 polynomials, $\{0, 1, y, y + 1, y^2, y^2 + 1, y^2 + y, y^2 + y + 1\}$.

Example 11.2.9
Let $p = 3$ and $q(y) = 2y^2$; then the set $\mathcal{P}$ consists of 3^2 polynomials, $\{0, 1, 2, y, y + 1, y+2, 2y, 2y + 1, 2y + 2\}$.

We now define two operations among polynomials of the set $\mathcal{P}$, namely modulo $q(y)$ addition, denoted by $\triangle$, and modulo $q(y)$ multiplication, denoted by $\Diamond$. Modulo $q(y)$ addition is defined for every pair of polynomials $a(y)$ and $b(y)$ from the set $\mathcal{P}$ as

$$a(y) \triangle b(y) = a(y) + b(y) \tag{11.53}$$

where $a(y) + b(y)$ is defined in (11.50).

The definition of modulo $q(y)$ multiplication requires the knowledge of the *Euclidean division algorithm*.

Euclidean division algorithm. For every pair of polynomials $\alpha(y)$ and $\beta(y)$ with coefficients from some field, and $\deg(\beta(y)) \geq \deg(\alpha(y)) > 0$, there exists a unique pair of polynomials $q(y)$ and $r(y)$ such that

$$\beta(y) = q(y)\,\alpha(y) + r(y) \tag{11.54}$$

where $0 \leq \deg(r(y)) < \deg(\alpha(y))$; polynomials $q(y)$ and $r(y)$ are called, respectively, *quotient polynomial* and *remainder or residue polynomial*. In a notation analogous to that used for integers we can write

$$q(y) = \left\lfloor \frac{\beta(y)}{\alpha(y)} \right\rfloor \tag{11.55}$$

and

$$r(y) = \beta(y) - \left\lfloor \frac{\beta(y)}{\alpha(y)} \right\rfloor \alpha(y) \tag{11.56}$$

Example 11.2.10
Let $p = 2$, $\beta(y) = y^4 + 1$, and $\alpha(y) = y^3 + y + 1$; then $y^4 + 1 = y(y^3 + y + 1) + y^2 + y + 1$, so $q(y) = y$ and $r(y) = y^2 + y + 1$.

We define modulo $q(y)$ multiplication, denoted by $\Diamond$, for polynomials $a(y)$ and $b(y)$ in the set $\mathcal{P}$ as

$$a(y) \Diamond b(y) = \begin{cases} a(y)\,b(y) & \text{if } \deg(a(y)\,b(y)) < \deg(q(y)) \\ a(y)\,b(y) - \left\lfloor \dfrac{a(y)\,b(y)}{q(y)} \right\rfloor q(y) & \text{otherwise} \end{cases} \tag{11.57}$$

It is easier to think of (11.57) as a typical multiplication operation for polynomials whose coefficients are given according to (11.52). If in this multiplication there are terms of degree greater than or equal to $\deg(q(y))$, then we use the relation $q(y) = 0$ to lower the degree.

Example 11.2.11
Let $p = 2$ and $q(y) = 1 + y + y^3$; then $(y^2 + 1)\Diamond(y + 1) = y^3 + y^2 + y + 1 = (-1 - y) + y^2 + y + 1 = (1 + y) + y^2 + y + 1 = y^2$.

It can be shown that the set of polynomials with coefficients from some field and degree less than $\deg(q(y))$ is a commutative ring with identity with respect to the operations modulo $q(y)$ addition and modulo $q(y)$ multiplication. We now find under what conditions this set of polynomials and operations forms a field.

Property 11a of modular polynomial arithmetic. If $q(y)$ is not irreducible, then the factors of $q(y)$, of degree greater than zero and less than $\deg(q(y))$, do not have multiplicative inverses.

Proof. Let $q(y) = a(y)\,b(y)$, where $0 < \deg(a(y)), \deg(b(y)) < \deg(q(y))$; then $a(y)\Diamond b(y) = 0$. Assume $a(y)$ has a multiplicative inverse, $a^{-1}(y)$; then, from $a^{-1}(y)\Diamond(a(y)\Diamond b(y)) = a^{-1}(y)\Diamond 0 = 0$ it is $(a^{-1}(y)\Diamond a(y))\Diamond b(y) = 0$, then $1\Diamond b(y) = 0$, or $b(y) = 0$. The last equation is a contradiction as by assumption $\deg(b(y)) > 0$. Similarly, we show that $b(y)$ does not have a multiplicative inverse.

We give without proof the following properties.

Property 11b of modular polynomial arithmetic. If $q(y)$ is irreducible and $a(y)\Diamond b(y) = 0$, then $a(y) = 0$, or $b(y) = 0$, or $a(y) = b(y) = 0$.

Property 11c of modular polynomial arithmetic. If $q(y)$ is irreducible, all non-zero elements of the set of polynomials $\mathcal{P}$ of degree less than $\deg(q(y))$ have multiplicative inverses.

We now have that the set of polynomials with coefficients from some field and degree less than $\deg(q(y))$ forms a field, with respect to the operations of modulo $q(y)$ addition and modulo $q(y)$ multiplication, if and only if $q(y)$ is irreducible.

Furthermore it can be shown that there exists at least one irreducible polynomial of degree m, for every $m \geq 1$, with coefficients from a generic field $\{0, 1, 2, \ldots, p - 1\}$. We now have a method of generating a field with p^m elements.

Example 11.2.12
Let $p = 2$ and $q(y) = y^2 + y + 1$; we have that $q(y)$ is irreducible. Consider the set $\mathcal{P}$ with elements $\{0, 1, y, y + 1\}$. The addition and multiplication tables for these elements modulo $y^2 + y + 1$ are given in Table 11.5 and Table 11.6, respectively.

Table 11.5 Modulo y^2+y+1 addition table for $p=2$.

$\triangle$	0	1	y	$y+1$
0	0	1	y	$y+1$
1	1	0	$y+1$	y
y	y	$y+1$	0	1
$y+1$	$y+1$	y	1	0

Table 11.6 Modulo y^2+y+1 multiplication table for $p=2$.

$\Diamond$	0	1	y	$y+1$
0	0	0	0	0
1	0	1	y	$y+1$
y	0	y	$y+1$	1
$y+1$	0	$y+1$	1	y

Devices to sum and multiply elements in a finite field

For the $GF(p^m)$ obtained by an irreducible polynomial of degree m,

$$q(y)=\sum_{i=0}^{m} q_i\, y^i \qquad q_i \in GF(p) \tag{11.58}$$

let $a(y)$ and $b(y)$ be two elements of $\mathcal{P}$:

$$a(y)=\sum_{i=0}^{m-1} a_i\, y^i \qquad a_i \in GF(p) \tag{11.59}$$

and

$$b(y)=\sum_{i=0}^{m-1} b_i\, y^i \qquad b_i \in GF(p) \tag{11.60}$$

The device to perform the addition (11.53),

$$s(y)=\sum_{i=0}^{m-1} s_i\, y^i = (a(y)+b(y)) \bmod q(y) \tag{11.61}$$

is illustrated in Figure 11.1. The implementation of a device to perform the multiplication is slightly more complicated, as illustrated in Figure 11.2, where T_c is the period of the clock applied to the shift-register (SR) with m elements, and all operations are modulo p.

Let us define

$$d(y)=\sum_{i=0}^{m-1} d_i\, y^i = (a(y)\, b(y)) \bmod q(y) \tag{11.62}$$

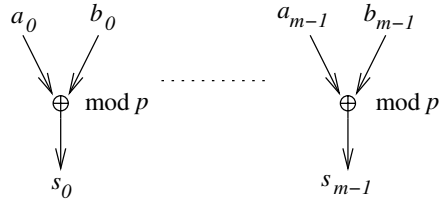


Figure 11.1. Device for the sum of two elements $(a_0, \ldots, a_{m-1})$ and $(b_0, \ldots, b_{m-1})$ of $GF(p^m)$.

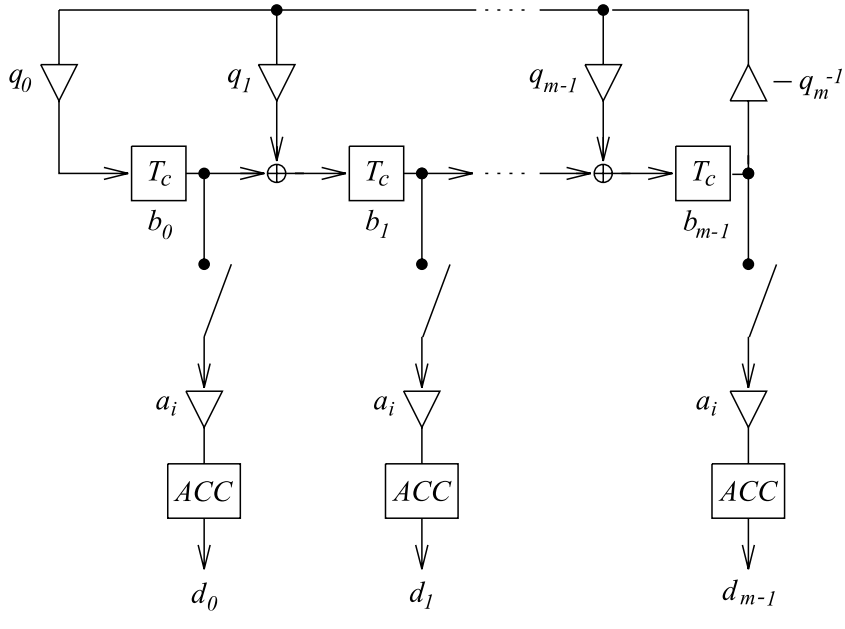


Figure 11.2. Device for the multiplication of two elements $(a_0, \ldots, a_{m-1})$ and $(b_0, \ldots, b_{m-1})$ of $GF(p^m)$. T_c is the clock period, and ACC denotes an accumulator. All additions and multiplications are modulo p.

The device is based on the following decomposition

$$\begin{aligned} a(y)\, b(y) \bmod q(y) &= \sum_{i=0}^{m-1} a_i\, y^i\, b(y) \bmod q(y) \\ &= a_0\, b(y) \\ &\quad + a_1 (y\, b(y)) \bmod q(y) \\ &\quad \vdots \\ &\quad + a_{m-1} (y^{m-1}\, b(y)) \bmod q(y) \end{aligned} \tag{11.63}$$

where additions and multiplications are modulo p. Now, using the identity $\sum_{i=0}^{m} q_i\, y^i = 0 \bmod q(y)$, note that the following relation holds:

$$\begin{aligned} y\, b(y) &= b_0 y + b_1 y^2 + \cdots + b_{m-2} y^{m-1} + b_{m-1} y^m \\ &= (-b_{m-1} q_m^{-1} q_0) + (b_0 - b_{m-1} q_m^{-1} q_1)\, y + \cdots + (b_{m-2} - b_{m-1} q_m^{-1} q_{m-1})\, y^{m-1} . \end{aligned} \tag{11.64}$$

The term $(y^i\, b(y)) \bmod q(y)$ is thus obtained by initializing the SR of Figure 11.2 to the sequence $(b_0, \ldots, b_{m-1})$, and by applying i clock pulses; the desired result is then contained in the shift register.

Observing (11.63), we find that it is necessary to multiply each element of the SR by a_i and accumulate the result; after multiplications by all coefficients $\{a_i\}$ have been performed, the final result is given by the content of the accumulators. Note that in the binary case, for $p = 2$, the operations of addition and multiplication are carried out by XOR and AND functions, respectively.

Remarks on finite fields

1. We have seen how to obtain finite fields with p (p a prime) elements, given by $\{0, 1, \ldots, p-1\}$, or p^m elements, using the Property 11c. These fields are also known as Galois fields and are usually denoted by $GF(p)$ or $GF(p^m)$. It can be shown that there are no other fields with a finite number of elements. Moreover, all fields with the same number of elements are identical, that is all finite fields are generated by the procedures discussed in the previous sections.

2. The field, from which the coefficients of the irreducible polynomial are chosen, is called the *ground field*; the field generated using the arithmetic of polynomials is called the *extension field.*

3. Every row of the addition table contains each field element once and only once; the same is true for the columns.

4. Every row of the multiplication table, except the row corresponding to the element 0, contains each field element once and only once; the same is true for the columns.

5. If we multiply any non-zero element by itself we get a non-zero element of the field (perhaps itself). As there are only $(q-1)$ non-zero elements, we must eventually find a situation for $j > i$ such that an element α multiplied by itself j times will equal α multiplied by itself i times, that is

$$\underbrace{\alpha \otimes \alpha \otimes \alpha \otimes \cdots \otimes \alpha}_{j \text{ times}} = \underbrace{\alpha \otimes \alpha \otimes \cdots \otimes \alpha}_{i \text{ times}} = \beta \tag{11.65}$$

We observe that

$$\underbrace{\alpha \otimes \alpha \otimes \cdots \otimes \alpha}_{j-i \text{ times}} \otimes \underbrace{\alpha \otimes \alpha \otimes \cdots \otimes \alpha}_{i \text{ times}} = \underbrace{\alpha \otimes \alpha \otimes \cdots \otimes \alpha}_{j \text{ times}} \tag{11.66}$$

Substituting (11.65) in (11.66), and observing that β has a multiplicative inverse, we can multiply from the right by this inverse to obtain

$$\alpha^{j-i} = \underbrace{\alpha \otimes \alpha \otimes \cdots \otimes \alpha}_{j-i \text{ times}} = 1 \tag{11.67}$$

Definition 11.9
For every non-zero field element, α, the *order of* α is the smallest integer ℓ such that $\alpha^{\ell} = 1$.

Example 11.2.13
Consider the field with elements $\{0, 1, 2, 3, 4\}$, and modulo 5 arithmetic. Then

element	*order*
1	1
2	4
3	4
4	2

(11.68)

Example 11.2.14
Consider the field $GF(2^2)$ with 4 elements, $\{0, 1, y, y+1\}$, and addition and multiplication modulo $y^2 + y + 1$. Then

element	*order*
1	1
y	3
$y+1$	3

(11.69)

6. An element from the field $GF(q)$ is said to be *primitive* if it has order $q - 1$. For fields generated by arithmetic modulo a polynomial $q(y)$, if the field element y is primitive we say that $q(y)$ is a *primitive irreducible polynomial*.

A property of finite fields that we give without proof is that every finite field has at least one primitive element; we note that once a primitive element has been identified, every other non-zero field element can be obtained by multiplying the primitive element by itself an appropriate number of times. A list of primitive polynomials for the ground field $GF(2)$ is given in Table 11.7.

Example 11.2.15
For the field $GF(4)$ generated by the polynomial arithmetic modulo $q(y) = y^2 + y + 1$, for the ground field $GF(2)$, y is a primitive element (see (11.69)); thus $y^2 + y + 1$ is a primitive polynomial.

7. The order of every non-zero element of $GF(q)$ must divide $(q - 1)$.

Table 11.7 List of primitive polynomials $q(y)$ of degree m for the ground field $GF(2)$.

m		m	
2	$1 + y + y^2$	14	$1 + y + y^6 + y^{10} + y^{14}$
3	$1 + y + y^3$	15	$1 + y + y^{15}$
4	$1 + y + y^4$	16	$1 + y + y^3 + y^{12} + y^{16}$
5	$1 + y^2 + y^5$	17	$1 + y^3 + y^{17}$
6	$1 + y + y^6$	18	$1 + y^7 + y^{18}$
7	$1 + y^3 + y^7$	19	$1 + y + y^2 + y^5 + y^{19}$
8	$1 + y^2 + y^3 + y^4 + y^8$	20	$1 + y^3 + y^{20}$
9	$1 + y^4 + y^9$	21	$1 + y^2 + y^{21}$
10	$1 + y^3 + y^{10}$	22	$1 + y + y^{22}$
11	$1 + y^2 + y^{11}$	23	$1 + y^5 + y^{23}$
12	$1 + y + y^4 + y^6 + y^{12}$	24	$1 + y + y^2 + y^7 + y^{24}$
13	$1 + y + y^3 + y^4 + y^{13}$		

Proof. Every non-zero element β can be written as the power of a primitive element α_p; this implies that there is some $i < (q-1)$ such that

$$\beta = \underbrace{\alpha_p \otimes \alpha_p \otimes \cdots \otimes \alpha_p}_{i \text{ times}} = \alpha_p^i \tag{11.70}$$

Note that from the definition of a primitive element we get $\alpha_p^{q-1} = 1$, but $\alpha_p^j \neq 1$ for $j < (q-1)$; furthermore there exists an integer ℓ such that $\beta^\ell = \alpha_p^{i\ell} = 1$. Consequently $(\ell)(i)$ is a multiple of $(q-1)$ and it is exactly the smallest multiple of i that is a multiple of $(q-1)$, thus $(i)(\ell) = \text{l.c.m.}(i, q-1)$, i.e. the least common multiple of i and $(q-1)$. We recall that

$$\text{l.c.m.}(a, b) = \frac{ab}{\text{g.c.d.}(a, b)} \tag{11.71}$$

where g.c.d.(a, b) is the greatest common divisor of a and b. Thus

$$(i)(\ell) = \frac{(i)(q-1)}{\text{g.c.d.}(i, q-1)} \tag{11.72}$$

and

$$\frac{q-1}{\text{g.c.d.}(i, q-1)} = \ell \tag{11.73}$$

Example 11.2.16
Let α_p be a primitive element of $GF(16)$; from (11.73) the orders of the non-zero field elements are:

field element $\beta = \alpha^i$	g.c.d.$(i, q-1)$	*order of field element* $\dfrac{q-1}{\text{g.c.d.}(i, q-1)}$
α_p	1	15
α_p^2	1	15
α_p^3	3	5
α_p^4	1	15
α_p^5	5	3
α_p^6	3	5
α_p^7	1	15
α_p^8	1	15
α_p^9	3	5
α_p^{10}	5	3
α_p^{11}	1	15
α_p^{12}	3	5
α_p^{13}	1	15
α_p^{14}	1	15
α_p^{15}	15	1

(11.74)

8. A ground field can itself be generated as an extension field. For example $GF(16)$ can be generated by taking an irreducible polynomial of degree 4 with coefficients from $GF(2)$, which we would call $GF(2^4)$, or by taking an irreducible polynomial of degree 2 with coefficients from $GF(4)$, which we would call $GF(4^2)$. In either case we would have the same field, except for the names of the elements.

Example 11.2.17
Consider the field $GF(2^3)$ generated by the primitive polynomial $q(y) = 1 + y + y^3$, with ground field $GF(2)$. As $q(y)$ is a primitive polynomial, each element of $GF(2^3)$, except the zero element, can be expressed as a power of y. Recalling the polynomial representation $\mathcal{P}$, we may attach to each polynomial a vector representation, with m components on $GF(p)$ given by the coefficients of the powers of the variable y. The three representations are reported in Table 11.8.

Roots of a polynomial

Consider a polynomial of degree m with coefficients that are elements of some field. We will use the variable x, as the polynomials are now considered for a purpose that is not that of generating a finite field. In fact, the field of the coefficients may itself have a polynomial representation.

Table 11.8 Three equivalent representations of the elements of $GF(2^3)$.

Exponential	Polynomial	Binary $(y^0 y^1 y^2)$
0	0	0 0 0
1	1	1 0 0
y	y	0 1 0
y^2	y^2	0 0 1
y^3	$1+y$	1 1 0
y^4	$y+y^2$	0 1 1
y^5	$1+y+y^2$	1 1 1
y^6	$1+y^2$	1 0 1

Consider, for example, a polynomial in x with coefficients from $GF(4)$. We immediately see that it is not worth using the notation $\{0, 1, y, y+1\}$ to identify the 4 elements of $GF(4)$, as the notation $\{0, 1, \alpha, \beta\}$ would be much simpler. For example, a polynomial of degree three with coefficients from $GF(4)$ is given by $f(x) = \alpha x^3 + \beta x^2 + 1$.

Given any polynomial $f(x)$, we say that γ is a *root of the equation* $f(x) = 0$ or, more simply, that it is a *root of* $f(x)$, if and only if $f(\gamma) = 0$. The definition is more complicated than it appears, as we must know the meaning of the two members of the equation $f(\gamma) = 0$. For example, we recall that the fundamental theorem of algebra states that every polynomial of degree m has exactly m roots, not necessarily distinct. If we take the polynomial $f(x) = x^2 + x + 1$ with coefficients from $\{0, 1\}$, what are its roots? As $f(0) = f(1) = 1$, we have that neither 0 nor 1 are roots.

Before proceeding, we recall a similar situation that we encounter in ordinary algebra. The polynomial $x^2 + 3$, with coefficients in the field of real numbers, has two roots in the field of complex numbers; however, no roots exist in the field of real numbers; therefore the polynomial does not have factors whose coefficients are real numbers. Thus we would say that the polynomial is irreducible, yet even the irreducible polynomial has complex-valued roots and can be factorized.

This situation is due to the fact that, if we have a polynomial $f(x)$ with coefficients from some field, the roots of the polynomial are either from that field or from an extension field of that field. For example, take the polynomial $f(x) = x^2 + x + 1$ with coefficients from $GF(2)$, and consider the extension field $GF(4)$ with elements $\{0, 1, \alpha, \beta\}$ that obey the addition and the multiplication rules given in Table 11.9 and Table 11.10, respectively. Then $f(\alpha) = f(x)|_{x=\alpha} = \alpha^2 \triangle \alpha \triangle 1 = (\beta \triangle \alpha) \triangle 1 = 1 \triangle 1 = 0$, thus α is a root. Similarly we find $f(\beta) = f(x)|_{x=\beta} = \beta^2 \triangle \beta \triangle 1 = (\alpha \triangle \beta) \triangle 1 = 1 \triangle 1 = 0$, thus the two roots of $f(x)$ are α and β. We can factor $f(x)$ into two factors, each of which is a polynomial in x with coefficients from $GF(4)$. For this purpose we consider $(x \triangle - \alpha) \diamondsuit (x \triangle - \beta) = (x \triangle \alpha) \diamondsuit (x \triangle \beta)$; leaving out the notations $\triangle$ and $\diamondsuit$ for $+$ and $\times$ we get

$$(x \triangle \alpha) \diamondsuit (x \triangle \beta) = x^2 + (\alpha + \beta)\, x + \alpha\beta = x^2 + x + 1 \tag{11.75}$$

Table 11.9 Addition table for the elements of $GF(4)$.

$\triangle$	0	1	α	β
0	0	1	α	β
1	1	0	β	α
α	α	β	0	1
β	β	α	1	0

Table 11.10 Multiplication table for the elements of $GF(4)$.

$\Diamond$	0	1	α	β
0	0	0	0	0
1	0	1	α	β
α	0	α	β	1
β	0	β	1	α

Thus if we use the operations defined in $GF(4)$, $(x+\alpha)$ and $(x+\beta)$ are factors of x^2+x+1; it remains that x^2+x+1 is irreducible as it has no factors with coefficients from $GF(2)$.

Property 1 of the roots of a polynomial. If γ is a root of $f(x)=0$, then $(x-\gamma)$, that is $(x+(-\gamma))$, is a factor of $f(x)$.

Proof. Using the Euclidean division algorithm, we divide $f(x)$ by $(x-\gamma)$ to get

$$f(x) = Q(x)\,(x-\gamma) + r(x) \tag{11.76}$$

where $\deg(r(x)) < \deg(x-\gamma) = 1$. Therefore

$$f(x) = Q(x)\,(x-\gamma) + r_0 \tag{11.77}$$

But $f(\gamma)=0$, so

$$f(\gamma) = 0 = Q(\gamma)\,(\gamma-\gamma) + r_0 = r_0 \tag{11.78}$$

therefore

$$f(x) = Q(x)\,(x-\gamma) \tag{11.79}$$

Property 2 of the roots of a polynomial. If $f(x)$ is an arbitrary polynomial with coefficients from $GF(p)$, p a prime, and β is a root of $f(x)$, then β^p is also a root of $f(x)$.

Proof. We consider the polynomial $f(x) = f_0+f_1x+f_2x^2+\cdots+f_mx^m$, where $f_i \in GF(p)$, and form the power $(f(x))^p$. It results

$$(f(x))^p = (f_0 + f_1x + f_2x^2 + \cdots + f_mx^m)^p = f_0^p + f_1^p x^p + \cdots + f_m^p x^{mp} \tag{11.80}$$

as the cross-terms contain a factor p, which is the same as 0 in $GF(p)$. On the other hand, for $f_i \neq 0$, $f_i^p = f_i$, as from Property 7 on page 853 the order of any non-zero element divides $p-1$; the equation is true also if f_i is the zero element. Therefore

$$(f(x))^p = f(x^p) \tag{11.81}$$

If β is a root of $f(x) = 0$, then $f(\beta) = 0$, and $f^p(\beta) = 0$. But $f^p(\beta) = f(\beta^p)$, so that $f(\beta^p) = 0$; therefore β^p is also a root of $f(x)$.

A more general form of the property just introduced, that we will give without proof, is expressed by the following property.

Property 2a of the roots of a polynomial. If $f(x)$ is an arbitrary polynomial having coefficients from $GF(q)$, with q a prime or a power of a prime, and β is a root of $f(x) = 0$, then β^q is also a root of $f(x) = 0$,

Example 11.2.18
Consider the polynomial $x^2 + x + 1$ with coefficients from $GF(2)$. We already have seen that α, element of $GF(4)$, is a root of $x^2 + x + 1 = 0$. Therefore α^2 is also a root; but $\alpha^2 = \beta$, so β is a second root. The polynomial has degree two, thus it has two roots and they are α and β, as previously seen. Note also that β^2 is also a root, but $\beta^2 = \alpha$.

Minimum function

Definition 11.10
Let β be an element of an extension field of $GF(q)$; the *minimum function of* β, $m_\beta(x)$, is the monic polynomial of least degree with coefficients from $GF(q)$ such that $m_\beta(x)|_{x=\beta} = 0$.

We now list some properties of the minimum function.

1. The minimum function is unique.

Proof. Assume there were two minimum functions, of the same degree and monic, $m_\beta(x)$ and $m'_\beta(x)$. Form the new polynomial $(m_\beta(x) - m'_\beta(x))$ whose degree is less than the degree of $m_\beta(x)$ and $m'_\beta(x)$; but $(m_\beta(x) - m'_\beta(x))|_{x=\beta} = 0$, so we have a new polynomial, whose degree is less than that of the minimum function, that admits β as root. Multiplying by a constant we can thus find a monic polynomial with this property, but this cannot be since the minimum function is the monic polynomial of least degree for which β is a root.

2. The minimum function is irreducible.

Proof. Assume the converse were true, that is $m_\beta(x) = a(x)\,b(x)$; then $m_\beta(x)|_{x=\beta} = a(\beta)\,b(\beta) = 0$. Then either $a(\beta) = 0$ or $b(\beta) = 0$, so that β is a root of a polynomial of degree less than the degree of $m_\beta(x)$. By making this polynomial monic we arrive at a contradiction.

3. Let $f(x)$ be any polynomial with coefficients from $GF(q)$, and let $f(x)|_{x=\beta} = 0$; then $f(x)$ is divisible by $m_\beta(x)$.

Proof. Use the Euclidean division algorithm to yield

$$f(x) = Q(x)\, m_\beta(x) + r(x) \tag{11.82}$$

where $\deg(r(x)) < \deg(m_\beta(x))$. Then we have that

$$f(\beta) = Q(\beta)\, m_\beta(\beta) + r(\beta) \tag{11.83}$$

but as $f(\beta) = 0$ and $m_\beta(\beta) = 0$, then $r(\beta) = 0$. As $\deg(r(x)) < \deg(m_\beta(x))$, the only possibility is $r(x) = 0$; thus $f(x) = Q(x)\, m_\beta(x)$.

4. Let $f(x)$ be any irreducible monic polynomial with coefficients from $GF(q)$ for which $f(\beta) = 0$, where β is an element of some extension field of $GF(q)$; then $f(x) = m_\beta(x)$.

Proof. From Property 3 $f(x)$ must be divisible by $m_\beta(x)$, but $f(x)$ is irreducible, so it is only trivially divisible by $m_\beta(x)$, that is $f(x) = K\, m_\beta(x)$: but $f(x)$ and $m_\beta(x)$ are both monic polynomials, therefore $K = 1$.

We now introduce some interesting propositions.

1. Let β be an element of $GF(q^m)$, with q prime; then the polynomial $F(x)$, defined as

$$F(x) = \prod_{i=0}^{m-1} (x - \beta^{q^i}) = (x - \beta)\,(x - \beta^q)\,(x - \beta^{q^2}) \cdots (x - \beta^{q^{m-1}}) \tag{11.84}$$

has all its coefficients from $GF(q)$.

Proof. Observing Property 7 on page 853, we have that the order of β divides $q^m - 1$, therefore $\beta^{q^m} = \beta$. Thus we can express $F(x)$ as

$$F(x) = \prod_{i=1}^{m} (x - \beta^{q^i}) \tag{11.85}$$

therefore

$$F(x^q) = \prod_{i=1}^{m} (x^q - \beta^{q^i}) = \prod_{i=1}^{m} (x - \beta^{q^{i-1}})^q = \prod_{j=0}^{m-1} (x - \beta^{q^j})^q = (F(x))^q \tag{11.86}$$

Consider now the expression $F(x) = \sum_{i=0}^{m} f_i\, x^i$; then

$$F(x^q) = \sum_{i=0}^{m} f_i\, x^{iq} \tag{11.87}$$

and

$$(F(x))^q = \left(\sum_{i=0}^{m} f_i\, x^i \right)^q = \sum_{i=0}^{m} f_i^q\, x^{iq} \tag{11.88}$$

Equating like coefficients in (11.87) and (11.88) we get $f_i^q = f_i$; hence f_i is a root of the equation $x^q - x = 0$. But on the basis of Property 7 on page 853 the q elements from $GF(q)$ all satisfy the equation $x^q - x = 0$, and this equation only has q roots; therefore the coefficients f_i are elements from $GF(q)$.

2. If $g(x)$ is an irreducible polynomial of degree m with coefficients from $GF(q)$, and $g(\beta) = 0$, where β is an element of some extension field of $GF(q)$, then $\beta, \beta^q, \beta^{q^2}, \ldots, \beta^{q^{m-1}}$ are all the roots of $g(x)$.

Proof. At least one root of $g(x)$ is in $GF(q^m)$; this follows by observing that, if we form $GF(q^m)$ using the arithmetic modulo $g(y)$, then y will be a root of $g(x) = 0$. From Proposition 1, if β is an element from $GF(q^m)$ then $F(x) = \prod_{i=0}^{m-1}(x - \beta^{q^i})$ has all coefficients from $GF(q)$; thus $F(x)$ has degree m, and $F(\beta) = 0$. As $g(x)$ is irreducible, we know that $g(x) = K\, m_\beta(x)$; but as $F(\beta) = 0$, and $F(x)$ and $g(x)$ have the same degree, then $F(x) = K_1 m_\beta(x)$, and therefore $g(x) = K_2 F(x)$. As $\beta, \beta^q, \beta^{q^2}, \ldots, \beta^{q^{m-1}}$, are all roots of $F(x)$, then they must also be all the roots of $g(x)$.

3. Let $g(x)$ be a polynomial with coefficients from $GF(q)$ which is also irreducible in this field. Moreover, let $g(\beta) = 0$, where β is an element of some extension field of $GF(q)$; then the degree of $g(x)$ equals the smallest integer k such that

$$\beta^{q^k} = \beta \tag{11.89}$$

Proof. We have that $\deg(g(x)) \geq k$ as $\beta, \beta^q, \beta^{q^2}, \ldots, \beta^{q^{k-1}}$, are all roots of $g(x)$ and by assumption are distinct. Assume that $\deg(g(x)) > k$; from Proposition 2, we know that β must be at least a double root of $g(x) = 0$, and therefore $g'(x) = (d/dx)g(x) = 0$ must also have β as a root. As $g(x)$ is irreducible we have that $g(x) = K\, m_\beta(x)$, but $m_\beta(x)$ must divide $g'(x)$; we get a contradiction because $\deg(g'(x)) < \deg(g(x))$.

Methods to determine the minimum function

1. Direct calculation.

Example 11.2.19
Consider the field $GF(2^3)$ obtained by taking the polynomial arithmetic modulo the irreducible polynomial $y^3 + y + 1$ with coefficients from $GF(2)$; the field elements are $\{0, 1, y, y+1, y^2, y^2+1, y^2+y, y^2+y+1\}$. Assume we want to find the minimum function of $\beta = (y+1)$. If $(y+1)$ is a root, also $(y+1)^2 = y^2 + 1$ and $(y+1)^4 = y^2 + y + 1$ are roots. Note that $(y+1)^8 = (y+1) = \beta$, thus the minimum function is

$$\begin{aligned} m_{y+1}(x) &= (x - \beta)\,(x - \beta^2)\,(x - \beta^4) \\ &= (x + (y+1))(x + (y^2+1))(x + (y^2+y+1)) \\ &= x^3 + x^2 + 1 \end{aligned} \tag{11.90}$$

2. Solution of the system of the coefficient equations.

Example 11.2.20
Consider the field $GF(2^3)$ of the previous example; as $(y+1)$, $(y+1)^2 = y^2+1$, $(y+1)^4 = y^2+y+1$, $(y+1)^8 = y+1$, the minimum function has degree three; as the minimum function is monic and irreducible, we have

$$m_{y+1}(x) = m_3x^3 + m_2x^2 + m_1x + m_0 = x^3 + m_2x^2 + m_1x + 1 \tag{11.91}$$

As $m_{y+1}(y+1) = 0$, then

$$(y+1)^3 + m_2(y+1)^2 + m_1(y+1) + 1 = 0 \tag{11.92}$$

that can be written as

$$y^2(1+m_2) + ym_1 + (m_2 + m_1 + 1) = 0 \tag{11.93}$$

As all coefficients of the powers of y must be zero, we get a system of equations in the unknown m_1 and m_2, whose solution is given by $m_1 = 0$ and $m_2 = 1$. Substitution of this solution in (11.91) yields

$$m_{y+1}(x) = x^3 + x^2 + 1 \tag{11.94}$$

3. Using the minimum function of the multiplicative inverse.

Definition 11.11
The *reciprocal polynomial* of any polynomial $m_\alpha(x) = m_0 + m_1x + m_2x^2 + \cdots + m_Kx^K$ is defined by $m^\alpha(x) = m_0x^K + m_1x^{K-1} + \cdots + m_{K-1}x + m_K$.

We use the following proposition that we give without proof.

The minimum function of the multiplicative inverse of a given element is equal to the reciprocal of the minimum function of the given element. In formulae: let $\alpha\beta = 1$, then $m_\beta(x) = m^\alpha(x)$.

Example 11.2.21
Consider the field $GF(2^6)$ obtained by taking the polynomial arithmetic modulo the irreducible polynomial $y^6 + y + 1$ with coefficients from $GF(2)$; the polynomial $y^6 + y + 1$ is primitive, thus from Property 7 on page 853 any non-zero field element can be written as a power of the primitive element y. From Proposition 2, we have that the minimum function of y is also the minimum function of y^2, y^4, y^8, y^{16}, y^{32}, the minimum function of y^3 is also the minimum function of y^6, y^{12}, y^{24}, y^{48}, y^{33}, and so forth. We list in Table 11.11 the powers of y that have the same minimum function.

Given the minimum function of y^{11}, $m_{y^{11}} = x^6 + x^5 + x^3 + x^2 + 1$, we want to find the minimum function of y^{13}. From Table 11.11 we note that y^{13} has the same minimum function as y^{52}; furthermore we note that y^{52} is the multiplicative inverse of y^{11}, as $(y^{11})(y^{52}) = y^{63} = 1$. Therefore the minimum function of y^{13} is the reciprocal polynomial of $m_{y^{11}}$, given by $m_{y^{13}} = x^6 + x^4 + x^3 + x + 1$.

Table 11.11 Powers of a primitive element in $GF(2^6)$ with the same minimum function.

1	2	4	8	16	32
3	6	12	24	48	33
5	10	20	40	17	34
7	14	28	56	49	35
9	18	36			
11	22	44	25	50	37
13	26	52	41	19	38
15	30	60	57	51	39
21	42				
23	46	29	58	53	43
27	54	45			
31	62	61	59	55	47

Properties of the minimum function

1. Let β be an element of order n in an extension field of $GF(q)$, and let $m_\beta(x)$ be the minimum function of β with coefficients from $GF(q)$; then $x^n - 1 = m_\beta(x)\, b(x)$, but $x^i - 1 \neq m_\beta(x)\, b(x)$ for $i < n$.

Proof. We show that β is a root of $x^n - 1$, as $\beta^n - 1 = 0$, but from Property 3 of the minimum function (see page 858) we know that $m_\beta(x)$ divides any polynomial $f(x)$ such that $f(\beta) = 0$; this proves the first part.

Assume that $x^i - 1 = m_\beta(x)\, b(x)$ for some $i < n$: then

$$x^i - 1|_{x=\beta} = m_\beta(x)\, b(x)|_{x=\beta} = 0 \tag{11.95}$$

so $\beta^i - 1 = 0$ for $i < n$. But from Definition 11.9 of the order of β (see page 852), n is the smallest integer such that $\beta^n = 1$, hence we get a contradiction.

2. Let $\beta_1, \beta_2, \ldots, \beta_L$ be elements of some extension field of $GF(q)$, and let $\ell_1, \ell_2, \ldots, \ell_L$ be the orders of these elements, respectively. Moreover, let $m_{\beta_1}(x)$, $m_{\beta_2}(x), \ldots, m_{\beta_L}(x)$ be the minimum functions of these elements with coefficients from $GF(q)$, and let $g(x)$ be the smallest monic polynomial with coefficients from $GF(q)$ that has $\beta_1, \beta_2, \ldots, \beta_L$ as roots: then

a) $g(x) = \text{l.c.m.}(m_{\beta_1}(x), m_{\beta_2}(x), \ldots, m_{\beta_L}(x))$;

b) if the minimum functions are all distinct, that is they do not have factor polynomials in common, then $g(x) = m_{\beta_1}(x)\, m_{\beta_2}(x) \ldots m_{\beta_L}(x)$;

c) if $n = \text{l.c.m.}(\ell_1, \ell_2, \ldots, \ell_L)$, then $x^n - 1 = h(x)\, g(x)$, and $x^i - 1 \neq h(x)\, g(x)$ for $i < n$.

Proof.

a) Noting that $g(x)$ must be divisible by each of the minimum functions, it must be the smallest degree monic polynomial divisible by $m_{\beta_1}(x), m_{\beta_2}(x), \ldots, m_{\beta_L}(x)$, but this is just the definition of the least common multiple.

b) If all the minimum functions are distinct, as each is irreducible, the least common multiple is given by the product of the polynomials.

c) As n is a multiple of the order of each element, $\beta_j^n - 1 = 0$, for $j = 1, 2, \ldots, L$; then $x^n - 1$ must be divisible by $m_{\beta_j}(x)$, for $j = 1, 2, \ldots, L$, and therefore it must be divisible by the least common multiple of these polynomials. Assume now that $g(x)$ divides $x^i - 1$ for $i < n$; then $\beta_j^i - 1 = 0$ for each $j = 1, 2, \ldots, L$, and thus i is a multiple of $\ell_1, \ell_2, \ldots, \ell_L$. But n is the smallest integer multiple of $\ell_1, \ell_2, \ldots, \ell_L$, hence we get a contradiction.

We note that if the extension field is $GF(q^k)$ and $L = q^k - 1 = n$, then $g(x) = x^n - 1$ and $h(x) = 1$.

11.2.3 Cyclic codes

In Section 11.2.1 we dealt with the theory of binary group codes. We now discuss a special class of linear codes. These codes, called *cyclic codes*, are based upon polynomial algebra and lead to particularly efficient implementations for encoding and decoding.

The algebra of cyclic codes

We consider polynomials with coefficients from some field $GF(q)$; in particular we consider the polynomial $x^n - 1$, and assume it can be factorized as

$$x^n - 1 = g(x)\, h(x) \tag{11.96}$$

Many such factorizations are possible for a given polynomial $x^n - 1$; we will consider any one of them. We denote the degrees of $g(x)$ and $h(x)$ as r and k, respectively; thus $n = k + r$. The choice of the symbols n, k and r is intentional, as they assume the same meaning as in the previous sections.

The polynomial arithmetic modulo $q(x) = x^n - 1$ is particularly important in the discussion of cyclic codes.

Proposition 11.2
Consider the set of all polynomials of the form $c(x) = a(x)\, g(x)$ modulo $q(x)$, as $a(x)$ ranges over all polynomials of all degrees with coefficients from $GF(q)$. This set must be finite as there are at most q^n remainder polynomials that can be obtained by dividing a polynomial by $x^n - 1$. Now we show that there are exactly q^k distinct polynomials.

Proof. There are at least q^k distinct polynomials $a(x)$ of degree less than or equal to $k - 1$, and each such polynomial leads to a distinct polynomial $a(x)\, g(x)$. In fact, as the degree

of $a(x)\,g(x)$ is less than $r + k = n$, no reduction modulo $x^n - 1$ is necessary for these polynomials.

Now let $a(x)$ be a polynomial of degree greater than or equal to k. To reduce the polynomial $a(x)\,g(x)$ modulo $x^n - 1$, we divide by $x^n - 1$ and keep the remainder; thus

$$a(x)\,g(x) = Q(x)\,(x^n - 1) + r(x) \tag{11.97}$$

where $0 \leq \deg(r(x)) < n$. By using (11.96), we can express $r(x)$ as

$$r(x) = (a(x) - h(x)\,Q(x))g(x) = a'(x)\,g(x) \tag{11.98}$$

As $r(x)$ is of degree less than n, $a'(x)$ is of degree less than k, but we have already considered all polynomials of this form; therefore $r(x)$ is one of the q^k polynomials determined in the first part of the proof.

Example 11.2.22
Let $g(x) = x + 1$, $GF(q) = GF(2)$, and $n = 4$; then all polynomials $a(x)\,g(x)$ modulo $x^4 - 1 = x^4 + 1$ are given by

$a(x)$	$a(x)\,g(x) \bmod (x^4 - 1)$	code word
0	0	0000
1	$x + 1$	1100
x	$x^2 + x$	0110
$x + 1$	$x^2 + 1$	1010
x^2	$x^3 + x^2$	0011
$x^2 + 1$	$x^3 + x^2 + x + 1$	1111
$x^2 + x$	$x^3 + x$	0101
$x^2 + x + 1$	$x^3 + 1$	1001

(11.99)

We associate with any polynomial of degree less than n and coefficients from $GF(q)$ a vector of length n with components equal to the coefficients of the polynomial, that is

$$f(x) = f_0 + f_1 x + f_2 x^2 + \cdots + f_{n-1}x^{n-1} \longleftrightarrow \mathbf{f} = (f_0, f_1, f_2, \ldots, f_{n-1}) \tag{11.100}$$

Note that in the definition f_{n-1} does not need to be non-zero.

We can now define cyclic codes. The code words will be the vectors associated with a set of polynomials; alternatively, we speak of the polynomials themselves as being code words or code polynomials (see (11.99)).

Definition 11.12
Choose a field $GF(q)$, a positive integer n and a polynomial $g(x)$ with coefficients from $GF(q)$ such that $x^n - 1 = g(x)\,h(x)$; furthermore, let $\deg(g(x)) = r = n - k$. Words of a cyclic code are the vectors of length n that are associated with all multiples of $g(x)$ reduced modulo $x^n - 1$. In formulae: $c(x) = a(x)\,g(x) \bmod (x^n - 1)$, for $a(x)$ polynomial with coefficients from $GF(q)$.

The polynomial $g(x)$ is called a *generator polynomial*.

Properties of cyclic codes

1. In a cyclic code there are q^k code words, as shown in the previous section.

2. A cyclic code is a linear code.

Proof. The all zero word is a code word as $0\,g(x) = 0$; any multiple of a code word is a code word, as if $a_1(x)\,g(x)$ is a code word so is $\alpha a_1(x)\,g(x)$. Let $a_1(x)\,g(x)$ and $a_2(x)\,g(x)$ be two code words; then

$$\alpha_1\,a_1(x)\,g(x) + \alpha_2\,a_2(x)\,g(x) = (\alpha_1\,a_1(x) + \alpha_2\,a_2(x))g(x) = a_3(x)\,g(x) \tag{11.101}$$

is a code word.

3. Every cyclic permutation of a code word is a code word.

Proof. It is enough to show that if $c(x) = c_0 + c_1x + \cdots + c_{n-2}x^{n-2} + c_{n-1}x^{n-1}$ corresponds to a code word, then also $c_{n-1} + c_0x + \cdots + c_{n-3}x^{n-2} + c_{n-2}x^{n-1}$ corresponds to a code word. But if $c(x) = a(x)\,g(x) = c_0 + c_1x + \cdots + c_{n-2}x^{n-2} + c_{n-1}x^{n-1} \bmod(x^n - 1)$, then $xc(x) = xa(x)\,g(x) = c_{n-1} + c_0x + \cdots + c_{n-3}x^{n-2} + c_{n-2}x^{n-1} \bmod(x^n - 1)$.

Example 11.2.23
Let $GF(q) = GF(2)$, $g(x) = x + 1$, and $n = 4$. From the previous example we obtain the code words, which can be grouped by the number of cyclic shifts.

code polynomials	code words	cyclic shifts
0	0000	1
$1 + x$	1100	4
$x + x^2$	0110	
$x^2 + x^3$	0011	
$1 + x^3$	1001	
$1 + x + x^2 + x^3$	1111	1
$1 + x^2$	1010	2
$x + x^3$	0101	

(11.102)

4. $c(x)$ is a code polynomial if and only if $c(x)\,h(x) = 0 \bmod(x^n - 1)$.

Proof. If $c(x)$ is a code polynomial, then $c(x) = a(x)g\,(x) \bmod(x^n - 1)$, but $h(x)\,c(x) = h(x)\,a(x)\,g(x) = a(x)\,(g(x)h(x)) = a(x)(x^n - 1) = 0 \bmod(x^n - 1)$.

Assume now $h(x)\,c(x) = 0 \bmod(x^n - 1)$; then $h(x)\,c(x) = Q(x)(x^n - 1) = Q(x)\,h(x)\,g(x)$, or $c(x) = Q(x)\,g(x)$, therefore $c(x)$ is a code polynomial.

5. Let $x^n - 1 = g(x)\,h(x)$, where $g(x) = g_0 + g_1x + \cdots + g_rx^r$ and $h(x) = h_0 + h_1x + \cdots + h_kx^k$; then the code corresponding to all multiples of $g(x)$ modulo $x^n - 1$ has the

generator matrix

$$\mathbf{G} = \begin{bmatrix} g_0 & g_1 & g_2 & \dots & g_r & 0 & 0 & \dots & 0 \\ 0 & g_0 & g_1 & \dots & g_{r-1} & g_r & 0 & \dots & 0 \\ & & & & & & & & \\ 0 & 0 & 0 & \dots & & & & \dots & g_r \end{bmatrix} \tag{11.103}$$

and parity check matrix

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & \dots & 0 & h_k & h_{k-1} & \dots & h_1 & h_0 \\ 0 & 0 & \dots & h_k & h_{k-1} & h_{k-2} & \dots & h_0 & 0 \\ & & & & & & & & \\ h_k & h_{k-1} & \dots & \dots & & & \dots & 0 & 0 \end{bmatrix} \tag{11.104}$$

Proof. We show that **G** is the generator matrix. The first row of **G** corresponds to the polynomial $g(x)$, the second to $xg(x)$ and the last row to $x^{k-1}\,g(x)$, but the code words are all words of the form

$$(a_0 + a_1 x + \cdots + a_{k-1}x^{k-1})g(x) = a_0\,g(x) + a_1(xg(x)) + \cdots + a_{k-1}(x^{k-1}g(x)) \tag{11.105}$$

But (11.105) expresses all code words as linear combinations of the rows of **G**, therefore **G** is the generator matrix of the code.

To show that **H** is the parity check matrix, we consider the product $c(x)\,h(x)$. If we write

$$c(x) = c_0 + c_1 x + \cdots + c_{n-1}x^{n-1} \tag{11.106}$$

and

$$h(x) = h_0 + h_1 x + \cdots + h_{k-1}x^{k-1} + h_k x^k + \cdots + h_{n-1}x^{n-1} \tag{11.107}$$

below this point where $h_{k+1} = h_{k+2} = \cdots = h_{n-1} = 0$, we get

$$d(x) = c(x)\,h(x) = d_0 + d_1 x + \cdots + d_{2n-2}x^{2n-2} \tag{11.108}$$

where

$$d_i = \begin{cases} \displaystyle\sum_{j=0}^{i} c_j\,h_{i-j} & \text{if } 0 \le i \le n-1 \\ \displaystyle\sum_{j=i-(n-1)}^{n-1} c_j\,h_{i-j} & \text{if } n \le i \le 2n-2 \end{cases} \tag{11.109}$$

We consider reducing $d(x)$ modulo $x^n - 1$, and denote the result as $\hat{d}(x) = \hat{d}_0 + \hat{d}_1 x + \cdots + \hat{d}_{n-1}x^{n-1}$; then $\hat{d}_i = d_i + d_{n+i}$, $i = 0, 1, 2, \dots, n-1$. If $c(x)\,h(x) = 0 \bmod(x^n - 1)$, then $\hat{d}_i = 0$, $i = 0, 1, 2, \dots, n-1$, therefore we get

$$\sum_{j=0}^{i} c_j\,h_{i-j} + \sum_{j=i+1}^{n-1} c_j\,h_{n+i-j} = 0 \qquad i = 0, 1, 2, \dots, n-1 \tag{11.110}$$

For $i = n - 1$, (11.110) becomes

$$\sum_{j=0}^{n-1} c_j \, h_{n-1-j} = 0 \tag{11.111}$$

or $[h_{n-1}\ h_{n-2}\ \ldots\ h_1\ h_0]\,[c_0\ c_1\ \ldots\ c_{n-1}]^T = 0$.

For $i = n - 2$, (11.110) becomes

$$\sum_{j=0}^{n-2} c_j \, h_{n-2-j} + c_{n-1} \, h_{n-1} = 0 \tag{11.112}$$

or $[h_{n-2}\ h_{n-3}\ \ldots\ h_0\ h_{n-1}]\,[c_0\ c_1\ \ldots\ c_{n-1}]^T = 0$.

After r steps, for $i = n - r$, (11.110) becomes

$$\sum_{j=0}^{n-r} c_j \, h_{n-r-j} + \sum_{j=n-r+1}^{n-1} c_j \, h_{2n-r-j} = 0 \tag{11.113}$$

or $[h_{n-r}\ h_{n-r-1}\ \ldots\ h_{n-r+2}\ h_{n-r+1}]\,[c_0\ c_1\ \ldots\ c_{n-1}]^T = 0$. The r equations can be written in matrix form as

$$\begin{bmatrix} h_{n-1} & h_{n-2} & \ldots & h_1 & h_0 \\ h_{n-2} & h_{n-3} & \ldots & h_0 & h_{n-1} \\ \vdots & & & & \vdots \\ h_{n-r} & h_{n-r-1} & \ldots & h_{n-r+2} & h_{n-r+1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{11.114}$$

therefore all code words are solutions of the equation $\mathbf{Hc} = \mathbf{0}$, where $\mathbf{H}$ is given by (11.104). It still remains to be shown that all solutions of the equation $\mathbf{Hc} = \mathbf{0}$ are code words. As $h_{n-1} = h_{n-2} = \cdots = h_{n-r+1} = 0$, and $h_0 \neq 0$, from (11.104) $\mathbf{H}$ has rank r, and can be written as $\mathbf{H} = [\mathbf{A}\ \mathbf{B}]$, where $\mathbf{B}$ is an $r \times r$ matrix with non-zero determinant; therefore

$$\begin{bmatrix} c_k \\ c_{k+1} \\ \vdots \\ c_{n-1} \end{bmatrix} = -\mathbf{B}^{-1}\,\mathbf{A} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{k-1} \end{bmatrix} \tag{11.115}$$

so there are $q^k = q^{n-r}$ solutions of the equation $\mathbf{Hc} = \mathbf{0}$. As there are q^k code words, all solutions of the equation $\mathbf{Hc} = \mathbf{0}$ are the code words in the cyclic code.

Example 11.2.24

Let $q = 2$ and $n = 7$. As $x^7 - 1 = x^7 + 1 = (x^3 + x + 1)(x^3 + x^2 + 1)(x + 1)$, we can choose $g(x) = x^3 + x + 1$ and $h(x) = (x^3 + x^2 + 1)(x + 1) = x^4 + x^2 + x + 1$; thus the

matrices **G** and **H** of this code are given by

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \tag{11.116}$$

$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \end{bmatrix} \tag{11.117}$$

Note that the columns of **H** are all possible non-zero vectors of length 3, so the code is a Hamming single error correcting (7,4) code.

6. In a code word, any string of r consecutive symbols, even taken cyclically, can identify the check positions.

Proof. From (11.115) it follows that the last r positions can be check positions. Now, if we cyclically permute every code word of m positions, the resultant words are themselves code words; thus the r check positions can be cyclically permuted anywhere in the code words.

7. As the r check positions can be the first r positions, a simple *encoding method* in canonical form is given by the following steps.

Step 1: represent the k information bits by the coefficients of the polynomial $m(x) = m_0 + m_1 x + \cdots + m_{k-1} x^{k-1}$.

Step 2: multiply $m(x)$ by x^r to obtain $x^r\, m(x)$.

Step 3: divide $x^r\, m(x)$ by $g(x)$ to obtain the remainder $r(x) = r_0 + r_1 x + \cdots + r_{r-1}\, x^{r-1}$.

Step 4: form the code word $c(x) = (x^r\, m(x) - r(x))$; note that the coefficients of $(-r(x))$ are the parity check bits.

Proof. To show that $(x^r\, m(x) - r(x))$ is a code word, we must prove that it is a multiple of $g(x)$: from Step 3 we obtain

$$x^r\, m(x) = Q(x)\, g(x) + r(x) \tag{11.118}$$

so that

$$(x^r\, m(x) - r(x)) = Q(x)\, g(x) \tag{11.119}$$

Example 11.2.25
Let $g(x) = 1 + x + x^3$, for $q = 2$ and $n = 7$. We report in Table 11.12 the message words $(m_0, \ldots, m_3)$ and the corresponding code words $(c_0, \ldots, c_6)$ obtained by the generator polynomial according to Definition 11.12 on page 863 for $a(x) = m(x)$; the same code in canonical form, obtained by (11.119), is reported in Table 11.13.

Table 11.12 (7,4) binary cyclic code, generated by $g(x) = 1 + x + x^3$.

Message $(m_0 m_1 m_2 m_3)$	Code polynomial $c(x) = m(x)\, g(x) \bmod x^7 - 1$	Code $(c_0 c_1 c_2 c_3 c_4 c_5 c_6)$
0000	$0g(x) = 0$	0000000
1000	$1g(x) = 1 + x + x^3$	1101000
0100	$xg(x) = x + x^2 + x^4$	0110100
1100	$(1 + x)g(x) = 1 + x^2 + x^3 + x^4$	1011100
0010	$x^2 g(x) = x^2 + x^3 + x^5$	0011010
1010	$(1 + x^2)g(x) = 1 + x + x^2 + x^5$	1110010
0110	$(x + x^2)g(x) = x + x^3 + x^4 + x^5$	0101110
1110	$(1 + x + x^2)g(x) = 1 + x^4 + x^5$	1000110
0001	$x^3 g(x) = x^3 + x^4 + x^6$	0001101
1001	$(1 + x^3)g(x) = 1 + x + x^4 + x^6$	1100101
0101	$(x + x^3)g(x) = x + x^2 + x^3 + x^6$	0111001
1101	$(1 + x + x^3)g(x) = 1 + x^2 + x^6$	1010001
0011	$(x^2 + x^3)g(x) = x^2 + x^4 + x^5 + x^6$	0010111
1011	$(1 + x^2 + x^3)g(x) = 1 + x + x^2 + x^3 + x^4 + x^5 + x^6$	1111111
0111	$(x + x^2 + x^3)g(x) = x + x^5 + x^6$	0100011
1111	$(1 + x + x^2 + x^3)g(x) = 1 + x^3 + x^5 + x^6$	1001011

Table 11.13 (7,4) binary cyclic code in canonical form, generated by $g(x) = 1 + x + x^3$.

Message $(m_0 m_1 m_2 m_3)$	Code polynomial $r(x) = x^r\, m(x) \bmod g(x)$ $c(x) = x^r\, m(x) - r(x)$	Code $(c_0 c_1 c_2 c_3 c_4 c_5 c_6)$
0000	0	0000000
1000	$1 + x + x^3$	1101000
0100	$x + x^2 + x^4$	0110100
1100	$1 + x^2 + x^3 + x^4$	1011100
0010	$1 + x + x^2 + x^5$	1110010
1010	$x^2 + x^3 + x^5$	0011010
0110	$1 + x^4 + x^5$	1000110
1110	$x + x^3 + x^4 + x^5$	0101110
0001	$1 + x^2 + x^6$	1010001
1001	$x + x^2 + x^3 + x^6$	0111001
0101	$1 + x + x^4 + x^6$	1100101
1101	$x^3 + x^4 + x^6$	0001101
0011	$x + x^5 + x^6$	0100011
1011	$1 + x^3 + x^5 + x^6$	1001011
0111	$x^2 + x^4 + x^5 + x^6$	0010111
1111	$1 + x + x^2 + x^3 + x^4 + x^5 + x^6$	1111111

Encoding method using a shift register of length r

We show that the steps of the encoding procedure can be accomplished by a *linear shift register with r stages*. We begin by showing how to divide $m_0 x^r$ by the generator polynomial $g(x)$ and obtain the remainder. As

$$g(x) = g_r x^r + g_{r-1} x^{r-1} + \cdots + g_1 x + g_0 \tag{11.120}$$

then

$$x^r = -g_r^{-1}(g_{r-1} x^{r-1} + g_{r-2} x^{r-2} + \cdots + g_1 x + g_0) \bmod g(x) \tag{11.121}$$

and

$$m_0 x^r = -m_0 g_r^{-1}(g_{r-1} x^{r-1} + g_{r-2} x^{r-2} + \cdots + g_1 x + g_0) \bmod g(x) \tag{11.122}$$

is the remainder after dividing $m_0 x^r$ by $g(x)$.

We now consider the scheme illustrated in Figure 11.3, where multiplications and additions are in $GF(q)$, and T_c denotes the clock period with which the message symbols $\{m_i\}$, $i = k-1, \ldots, 1, 0$, are input to the shift register. In the *binary case*, the storage elements are flip flops, the addition is the modulo 2 addition, and multiplication by g_i is performed by a switch that is open or closed depending upon whether $g_i = 0$ or 1, respectively. Note that if m_0 is input, the storage elements of the shift register will contain the coefficients of the remainder upon dividing $m_0 x^r$ by $g(x)$.

Let us suppose we want to compute the remainder upon dividing $m_1 x^{r+1}$ by $g(x)$. We could first compute the remainder of the division of $m_1 x^r$ by $g(x)$, by presenting m_1 at the input, then multiplying the remainder by x, and again reduce the result modulo $g(x)$. But once the remainder of the first division is stored in the shift register, multiplication by x and division by $g(x)$ are obtained simply by clocking the register once with no input. In fact, if the shift register contains the polynomial

$$b(x) = b_0 + b_1 x + \cdots + b_{r-1} x^{r-1} \tag{11.123}$$

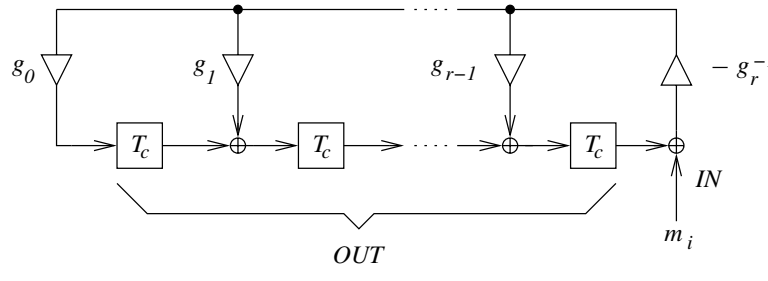


Figure 11.3. Scheme of an encoder for cyclic codes using a shift register with *r* elements.

and we multiply by x and divide by $g(x)$, we obtain

$$\begin{aligned} x\,b(x) &= b_0 x + b_1 x^2 + \cdots + b_{r-1} x^r \\ &= b_0 x + b_1 x^2 + \cdots + b_{r-2} x^{r-1} \\ &\quad + b_{r-1}(-g_r^{-1}(g_{r-1} x^{r-1} + \cdots + g_1 x + g_0)) \bmod g(x) \\ &= -b_{r-1} g_r^{-1} g_0 + (b_0 - b_{r-1} g_r^{-1} g_1)\, x + \cdots \\ &\quad + (b_{r-2} - b_{r-1} g_r^{-1} g_{r-1})\, x^{r-1} \bmod g(x) \end{aligned} \tag{11.124}$$

that is just the result obtained by clocking the register once.

Finally, we note that superimposition holds in computing remainders; in other words, if $m_0 x^r = r_1(x) \bmod g(x)$ and $m_1 x^{r+1} = r_2(x) \bmod g(x)$, then $m_0 x^r + m_1 x^{r+1} = r_1(x) + r_2(x) \bmod g(x)$. Therefore, to compute the remainder upon dividing $m_0 x^r + m_1 x^{r+1}$ by $g(x)$ using the scheme of Figure 11.3, we would first input m_1 and then next input m_0 to the shift register.

Hence, to compute the remainder upon dividing $x^r m(x) = m_0 x^r + m_1 x^{r+1} + \cdots + m_{k-1} x^{n-1}$ by $g(x)$ we input the symbols $m_{k-1}, m_{k-2}, \ldots, m_1, m_0$ to the device of Figure 11.3; after the last symbol, m_0, enters, the coefficients of the desired remainder will be contained in the storage elements. From (11.119) we note that the parity check bits are the inverse elements (with respect to addition) of the values contained in the register.

In general for an input $z(x)$, polynomial with n coefficients, after n clock pulses the device of Figure 11.3 yields $x^r z(x) \bmod g(x)$.

Encoding method using a shift register of length k

It is also possible to accomplish the encoding procedure for cyclic codes by using a shift register with k stages. Again we consider the first r positions of the code word as the parity check bits, $p_0, p_1, \ldots, p_{r-1}$; utilizing the first row of the parity check matrix we obtain

$$h_k\, p_{r-1} + h_{k-1}\, m_0 + \cdots + h_1\, m_{k-2} + h_0\, m_{k-1} = 0 \tag{11.125}$$

or

$$p_{r-1} = -h_k^{-1}(h_{k-1}\, m_0 + h_{k-2}\, m_1 + \cdots + h_1\, m_{k-2} + h_0\, m_{k-1}) \tag{11.126}$$

Similarly, using the second row we obtain

$$p_{r-2} = -h_k^{-1}\,(h_{k-1}\, p_{r-1} + h_{k-2}\, m_0 + \cdots + h_1\, m_{k-3} + h_0\, m_{k-2}) \tag{11.127}$$

and so forth.

Let us consider the scheme of Figure 11.4 and assume that the register initially contains the symbols $m_0, m_1, \ldots, m_{k-1}$. After one clock pulse m_{k-1} will appear at the output, all information symbols will have moved by one place to the right and the parity check symbol p_{r-1} will appear in the first left-most storage element; after the second clock pulse, m_{k-2} will appear at the output, all symbols contained in the storage elements will move one place to the right and the parity check symbol p_{r-2} will appear in the left-most storage element. It is easy to verify that, if we apply n clock pulses to the device, the output will be given by the k message symbols followed by the r parity check bits.

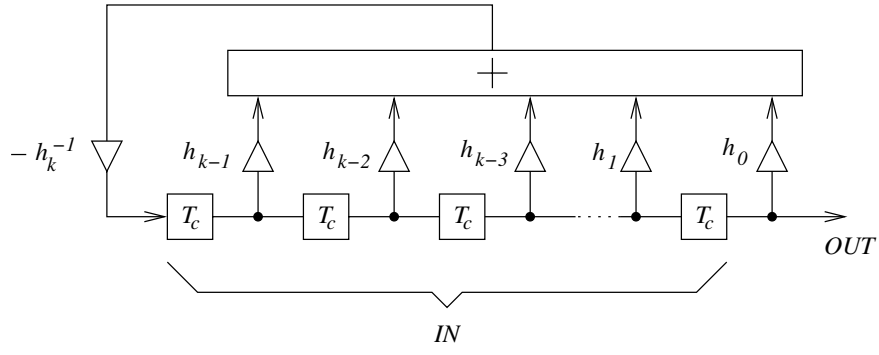


Figure 11.4. Scheme of an encoder for cyclic codes using a shift register with k elements.

Hard decoding of cyclic codes

We discover (see page 839) that all vectors in the same coset of the decoding table have the same syndrome and that vectors in different cosets have different syndromes.

Proposition 11.3
All polynomials corresponding to vectors in the same coset have the same remainder if they are divided by $g(x)$; polynomials corresponding to vectors in different cosets have different remainders if they are divided by $g(x)$.

Proof. Let $a_j(x)\,g(x)$, $j = 0, 1, 2, \ldots, q^k - 1$, be the code words, and $\eta_i(x)$, $i = 0, 1, 2, \ldots, q^r - 1$, be the coset leaders. Assume $z_1(x)$ and $z_2(x)$ are two arbitrary polynomials of degree $n - 1$: if they are in the same coset, say, the i-th, then

$$z_1(x) = \eta_i(x) + a_{j_1}(x)\,g(x) \tag{11.128}$$

and

$$z_2(x) = \eta_i(x) + a_{j_2}(x)\,g(x) \tag{11.129}$$

As upon dividing $a_{j_1}\,g(x)$ and $a_{j_2}\,g(x)$ by $g(x)$ we get 0 as a remainder, the division of $z_1(x)$ and $z_2(x)$ by $g(x)$ gives the same remainder, namely the polynomial $r_i(x)$, where

$$\eta_i(x) = Q(x)\,g(x) + r_i(x) \qquad \deg(r_i(x)) < \deg(g(x)) = r \tag{11.130}$$

Now assume $z_1(x)$ and $z_2(x)$ are in different cosets, say, the i_1-th and i_2-th cosets, but have the same remainder, say, $r_0(x)$, if they are divided by $g(x)$; then the coset leaders $\eta_{i_1}(x)$ and $\eta_{i_2}(x)$ of these cosets must give the same remainder $r_0(x)$ if they are divided by $g(x)$, i.e.

$$\eta_{i_1}(x) = Q_1(x)\,g(x) + r_0(x) \tag{11.131}$$

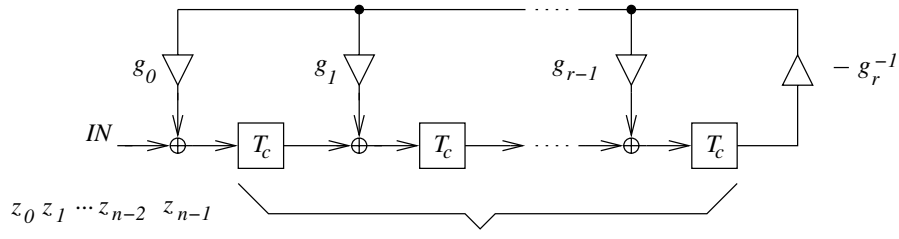


Figure 11.5. Device to compute the division of the polynomial $z(x) = z_0 + z_1 x + \cdots + z_{n-1}x^{n-1}$ by $g(x)$. After n clock pulses the r storage elements contain the remainder $r_0, r_1, \ldots, r_{r-1}$.

and

$$\eta_{i_2}(x) = Q_2(x)\, g(x) + r_0(x) \tag{11.132}$$

therefore we get

$$\eta_{i_2}(x) = \eta_{i_1}(x) + (Q_2(x) - Q_1(x))g(x) = \eta_{i_1}(x) + Q_3(x)\, g(x) \tag{11.133}$$

This implies that $\eta_{i_1}(x)$ and $\eta_{i_2}(x)$ are in the same coset, which is a contradiction.

This result leads to the following decoding method for cyclic codes.

Step 1: compute the remainder upon dividing the received polynomial $z(x)$ of degree $n-1$ by $g(x)$, for example, by the device of Figure 11.5 (see (11.124)), by presenting at the input the sequence of received symbols, and applying n clock pulses. The remainder identifies the coset leader of the coset where the received polynomial is located.

Step 2: subtract the coset leader from the received polynomial to obtain the decoded code word.

Hamming codes

Hamming codes are binary cyclic single error correcting codes. We consider cyclic codes over $GF(2)$, where $g(x)$ is an irreducible polynomial of degree r such that $g(x)$ divides $x^{2^r-1} - 1$, but not $x^\ell - 1$ for $\ell < 2^r - 1$.

To show that $g(x)$ is a primitive irreducible polynomial, we choose $n = 2^r - 1$, thus $x^n - 1 = g(x)\, h(x)$ and the corresponding cyclic code has parameters $n = 2^r - 1$, r, and $k = 2^r - 1 - r$.

Proposition 11.4
This code has minimum distance $d^H_{min} = 3$ and therefore is a single error correcting code.

Proof. We first prove that $d^H_{min} \geq 3$ by showing that all single error polynomials have distinct, non-zero remainders if they are divided by $g(x)$.

Assume that $x^i = 0 \bmod g(x)$, for some $0 \leq i \leq n-1$; then $x^i = Q(x)\, g(x)$, which is impossible since $g(x)$ is not divisible by x.

Now assume that x^i and x^j give the same remainder upon division by $g(x)$, and that $0 \leq i < j \leq n-1$; then

$$x^j - x^i = x^i(x^{j-i} - 1) = Q(x)\, g(x) \tag{11.134}$$

but $g(x)$ does not divide x^i, so it must divide $(x^{j-i} - 1)$. But $0 < j - i \leq n-1$ and by assumption $g(x)$ does not divide this polynomial. Hence $d^H_{min} \geq 3$.

By the limit (11.15) we know that for a code with fixed n and k the following inequality holds:

$$2^k \left[1 + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{\left\lfloor \frac{d^H_{min} - 1}{2} \right\rfloor} \right] \leq 2^n \tag{11.135}$$

As $n = 2^r - 1$ and $k = n - r$, we have

$$\left[1 + \binom{2^r - 1}{1} + \binom{2^r - 1}{2} + \cdots + \binom{2^r - 1}{\left\lfloor \frac{d^H_{min} - 1}{2} \right\rfloor} \right] \leq 2^r \tag{11.136}$$

but

$$1 + \binom{2^r - 1}{1} = 2^r \tag{11.137}$$

and therefore $d^H_{min} \leq 3$.

We have seen in the previous section how to implement an encoder for a cyclic code. We consider now the decoder device of Figure 11.6, whose operations are described as follows.

1. Initially all storage elements of the register contain zeros and the switch SW is in position 0. The received n-bit word $\mathbf{z} = (z_0, \ldots, z_{n-1})$ is sequentially clocked into the lower register, with n storage elements, and into the feedback register, with r storage elements, whose content is denoted by $r_0, r_1, \ldots, r_{r-1}$.

2. After n clock pulses, the behavior of the decoder depends on the value of v: if $v = 0$, the switch SW remains in the position 0 and both registers are clocked once. This procedure is repeated until $v = 1$, which occurs for $r_0 = r_1 = \cdots = r_{r-2} = 0$; then SW moves to position 1 and the content of the last stage of the feedback shift register is added modulo 2 to the content of the last stage of the lower register; both registers are then clocked until the n bits of the entire word are obtained at the output of the decoder. Overall, $2n$ clock pulses are needed.

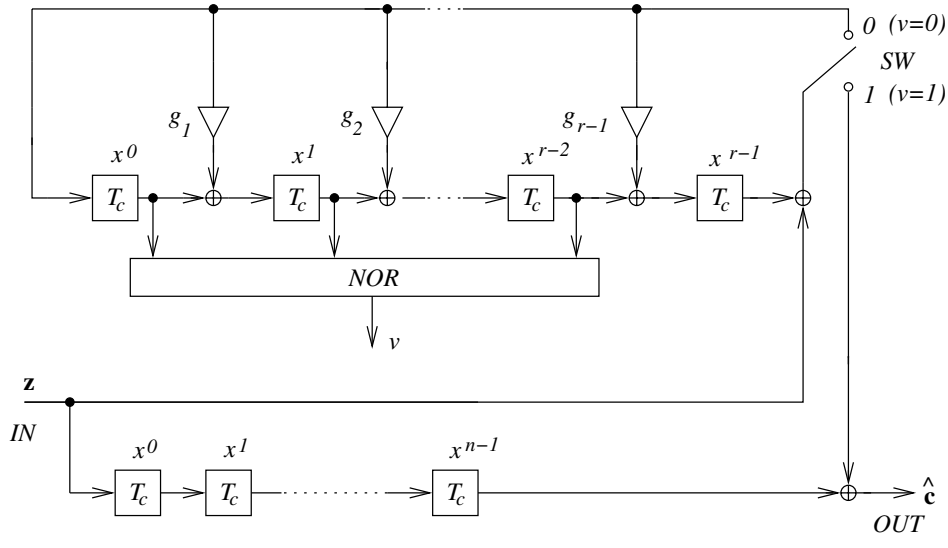


Figure 11.6. Scheme of a decoder for binary cyclic single error correcting codes (Hamming codes). All operations are in $GF(2)$.

We now illustrate the procedure of the scheme of Figure 11.6. First of all we note that for the first n clocks the device coincides with that of Figure 11.3, hence the content of the shift register is given by

$$r(x) = x^r\, z(x) \bmod g(x) \tag{11.138}$$

We consider two cases.

1. *The received word is correct,* $z(x) = c(x)$. After the first n clock pulses, from (11.138) we have

$$r(x) = x^r\, c(x) = x^r\, a(x)\, g(x) = 0 \bmod g(x) \tag{11.139}$$

and thus

$$v = 1 \quad \text{and} \quad r_{r-1} = 0 \tag{11.140}$$

In the successive n clock pulses we have

$$\hat{c}_i = z_i + 0 \qquad i = 0, \ldots, n-1 \tag{11.141}$$

therefore $\hat{\mathbf{c}} = \mathbf{c}$.

2. *The received word is affected by one error,* $z(x) = c(x) + x^i$. In other words we assume that there is a single error in the i-th bit, $0 \le i \le n-1$.

After the first n clock pulses, it is

$$r(x) = x^r\, x^i \bmod g(x) \tag{11.142}$$

If $i = n - 1$, we have

$$\begin{aligned} r(x) &= x^r\, x^{n-1} \\ &= x^n\, x^{r-1} \\ &= (x^n - 1)\, x^{r-1} + x^{r-1} \\ &= h(x)\, g(x)\, x^{r-1} + x^{r-1} \bmod g(x) \\ &= x^{r-1} \end{aligned} \tag{11.143}$$

and consequently

$$r_{r-1} = 1 \quad \text{and} \quad r_{r-2} = \cdots = r_0 = 0 \quad (v = 1) \tag{11.144}$$

This leads to switching SW, therefore during the last n clock pulses we have

$$\hat{c}_{n-1} = z_{n-1} + 1 \quad \hat{c}_i = z_i + 0 \qquad i = n - 2, \ldots, 0 \tag{11.145}$$

Therefore the bit in the last stage of the buffer is corrected.

If $i = n - j$, then we have

$$r(x) = x^{r-j} \tag{11.146}$$

thus only at the $(n + j - 1)$-th clock pulse the condition (11.144) that forces to switch SW from 0 to 1 occurs; therefore, at the next clock pulse the received bit in error will be corrected.

Burst error detection

We assume that a burst error occurs in the received word and that this burst affects $\ell \leq n-k$ consecutive bits, that is the error pattern is

$$\mathbf{e} = (0, 0, 0, \ldots, 0, \overset{\text{bit } j}{1}, \ldots, \ldots, \overset{\text{bit } (j+\ell-1)}{1}, 0, \ldots, 0) \tag{11.147}$$

where within the two '1's the values can be either '0' or '1'.

Then we can write the vector $\mathbf{e}$ in polynomial form,

$$e(x) = x^j\, B(x) \tag{11.148}$$

where $B(x)$ is a polynomial of degree $\ell - 1 \leq n - k - 1$. Thus $e(x)$ is divisible by the generator polynomial $g(x)$ if $B(x)$ is divisible by $g(x)$, as x is not a factor of $g(x)$; but $B(x)$ has a degree at most equal to $(n - k - 1)$, lower than the degree of $g(x)$, equal to $n - k$; therefore $e(x)$ cannot be a code word. We have then that all burst errors of length ℓ less than or equal to $r = n - k$ are detectable by (n, k) cyclic codes. This result leads to the introduction of the *cyclic redundancy check* (CRC) codes.

11.2.4 Simplex cyclic codes

We consider a class of cyclic codes over $GF(q)$ such that the Hamming distance between every pair of distinct code words is a constant; this is equivalent to stating that the weight

Table 11.14 Parameters of some simplex binary codes.

n	k	r	d_{min}
7	3	4	4
15	4	11	8
31	5	26	16
63	6	57	32
127	7	120	64

of all non-zero code words is equal to the same constant. We show that in the binary case, for these codes the non-zero code words are related to the PN sequences of Appendix 3.A.

Let $n = q^k - 1$, and $x^n - 1 = g(x)\, h(x)$, where we choose $h(x)$ as a *primitive* polynomial of degree k; then the resultant code has minimum distance

$$d_{min}^H = (q-1)\, q^{k-1} \tag{11.149}$$

The parameters of some binary codes in this class are listed in Table 11.14.

To show that these codes have minimum distance given by (11.149), first we prove the following:

Property. All non-zero code words have the same weight.

Proof. We begin by showing that

$$x^i\, g(x) \neq x^j\, g(x) \mod(x^n - 1) \qquad 0 \leq i < j \leq n-1 \tag{11.150}$$

Assume the converse is true, that is $x^i\, g(x) = x^j\, g(x) \mod(x^n - 1)$; then

$$x^i (x^{j-i} - 1)\, g(x) = Q(x)\, g(x)\, h(x) \tag{11.151}$$

or

$$x^i (x^{j-i} - 1) = Q(x)\, h(x) \tag{11.152}$$

But this is impossible since $h(x)$ is a primitive polynomial of degree k and cannot divide $(x^{j-i} - 1)$, as $(j - i) < n = (q^k - 1)$.

Relation (11.150) implies that all cyclic shifts of the code polynomial $g(x)$ are unique, but there are $n = (q^k - 1)$ cyclic shifts. Furthermore we know that there are only q^k code words and one is the all-zero word; therefore all cyclic shifts of $g(x)$ are all the non-zero code words and they all have the same weight.

Recall Property 2 of a group code (see page 832), that is if all code words of a linear code are written as rows of a matrix, every column is either formed by all zeros, or it consists of each field element repeated an equal number of times. If we apply this result to

a simplex code, we find that no column can be all zero as the code is cyclic, so the sum of the weights of all code words is given by

$$\text{sum of weights} = n(q-1)\frac{q^k}{q} = (q^k-1)(q-1)q^{k-1} \tag{11.153}$$

But there are $(q^k - 1)$ non-zero code words, all of the same weight; the weight of each word is then given by

$$\text{weight of non-zero code words} = (q-1)q^{k-1} \tag{11.154}$$

Therefore the minimum weight of the non-zero code words is given by

$$d_{min}^H = (q-1)q^{k-1} \tag{11.155}$$

Example 11.2.26
Let $q = 2$, $n = 15$, and $k = 4$; hence $r = 11$, and $d_{min}^H = 8$. Choose $h(x)$ as a primitive irreducible polynomial of degree 4 over $GF(2)$, $h(x) = x^4 + x + 1$.

The generator polynomial $g(x)$ is obtained by dividing $x^{15} - 1$ by $h(x) = x^4 + x + 1$ in $GF(2)$, obtaining

$$g(x) = x^{11} + x^8 + x^7 + x^5 + x^3 + x^2 + x + 1 \tag{11.156}$$

Given an extension field $GF(2^k)$ and $n = 2^k - 1$, from Property 2 on page 861, $x^n - 1$ is given by the l.c.m. of the minimum functions of the elements of the extension field. As $h(x)$ is a primitive polynomial, $g(x)$ is therefore given by the l.c.m. of the minimum functions of the elements $1, \alpha^3, \alpha^5, \alpha^7$, from $GF(2^4)$. By a table similar to Table 11.11, obtained for $GF(2^6)$, and using one of the three methods to determine the minimum function (see page 859), it turns out that the generator polynomial for this code is given by

$$g(x) = (x+1)(x^4 + x^3 + x^2 + x + 1)(x^2 + x + 1)(x^4 + x^3 + 1) \tag{11.157}$$

Relation to PN sequences

We consider a periodic binary sequence of period L, given by $\ldots, p(-1), p(0), p(1), \ldots$, with $p(\ell) \in \{0, 1\}$. We define the normalized autocorrelation function of this sequence as

$$\mathtt{r}_p(m) = \frac{1}{L}\left[L - 2\sum_{\ell=0}^{L-1}(p(\ell) \oplus p(\ell - m))\right] \tag{11.158}$$

Note that with respect to (3.302), now $p(\ell) \in \{0, 1\}$ rather than $p(\ell) \in \{-1, 1\}$.

Theorem 11.1
If the periodic binary sequence $\{p(\ell)\}$ is formed by repeating any non-zero code word of a simplex binary code of length $L = n = 2^k - 1$, then

$$\mathtt{r}_p(m) = \begin{cases} 1 & m = 0, \pm L, \pm 2L, \ldots \\ -\dfrac{1}{L} & \text{otherwise} \end{cases} \tag{11.159}$$

Proof. We recall that for a simplex binary code all non-zero code words

a) have weight 2^{k-1},

b) are cyclic permutations of the same code word.

As the code is linear, the Hamming distance between any code word and a cyclic permutation of this word is 2^{k-1}; this means that for the periodic sequence formed by repeating any non-zero code word we obtain

$$\sum_{\ell=0}^{L-1}(p(\ell) \oplus p(\ell - m)) = \begin{cases} 0 & m = 0, \pm L, \pm 2L, \ldots \\ 2^{k-1} & \text{otherwise} \end{cases} \tag{11.160}$$

Substitution of (11.160) in (11.158) yields

$$\mathfrak{r}_p(m) = \begin{cases} 1 & m = 0, \pm L, \pm 2L, \ldots \\ \dfrac{2^k - 1 - 2^k}{2^k - 1} = -\dfrac{1}{2^k - 1} & \text{otherwise} \end{cases} \tag{11.161}$$

If we recall the implementation of Figure 11.4, we find that the generation of such sequences is easy. We just need to determine the shift register associated with $h(x)$, load it with anything except all zeros, and let it run. For example, choosing $h(x) = x^4 + x + 1$, we get the PN sequence of Figure 3.41, as illustrated in Figure 11.7, where $L = n = 2^4 - 1 = 15$.

11.2.5 BCH codes

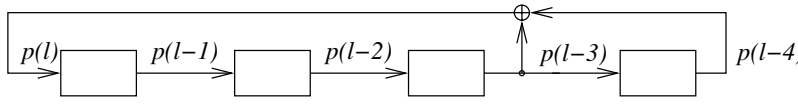
An alternative method to specify the code polynomials

Definition 11.13
Suppose we arbitrarily choose L *elements from* $GF(q^m)$ that we denote as $\alpha_1, \alpha_2, \ldots, \alpha_L$ (we will discuss later how to select these elements), and we consider polynomials, of degree $n - 1$ or less, with *coefficients from* $GF(q)$. A polynomial is a code polynomial if each of the elements $\alpha_1, \alpha_2, \ldots, \alpha_L$ is a root of the polynomial. The code then consists of the set of all the code polynomials.

Using this method we see that $c(x) = c_0 + c_x + \cdots + c_{n-1}x^{n-1}$ is a code polynomial if and only if $c(\alpha_1) = c(\alpha_2) = \cdots = c(\alpha_L) = 0$; thus

$$\begin{bmatrix} (\alpha_1)^0 & (\alpha_1)^1 & (\alpha_1)^2 & \ldots & (\alpha_1)^{n-1} \\ (\alpha_2)^0 & (\alpha_2)^1 & (\alpha_2)^2 & \ldots & (\alpha_2)^{n-1} \\ \vdots & & & & \vdots \\ (\alpha_L)^0 & (\alpha_L)^1 & (\alpha_L)^2 & \ldots & (\alpha_L)^{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{11.162}$$

All vectors $\mathbf{c} = [c_0, c_1, \ldots, c_{n-1}]^T$ with elements from $GF(q)$ that are solutions of this set of equations, where operations are performed according to the rules of $GF(q^m)$, are



l	$p(l-1)$	$p(l-2)$	$p(l-3)$	$p(l-4)$
0	0	0	0	1
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	1	0	0	1
5	1	1	0	0
6	0	1	1	0
7	1	0	1	1
8	0	1	0	1
9	1	0	1	0
10	1	1	0	1
11	1	1	1	0
12	1	1	1	1
13	0	1	1	1
14	0	0	1	1
15	0	0	0	1

Figure 11.7. Generation of a PN sequence as a repetition of a code word of a simplex code with $L = n = 15$.

code words. The form of (11.162) resembles equation (11.20), where $\mathbf{H}$ is the generalized parity check matrix. One obvious difference is that in (11.20) $\mathbf{H}$ and $\mathbf{c}$ have elements from the same field, whereas this does not occur for the vector equation (11.162). However, this difference is not crucial as each element from $GF(q^m)$ can be written as a vector of length m with elements from $GF(q)$. Thus each element $(\alpha_i)^j$ in the matrix is replaced by a column vector with m components. The resultant matrix, with Lm rows and n columns, consists of elements from $GF(q)$ and is therefore just a generalized parity check matrix for the considered code.

From the above discussion it appears that, if L roots are specified, the resultant linear code has $r = Lm$ parity check symbols, as the parity check matrix has $r = Lm$ rows. However, not all rows of the matrix are necessarily independent; therefore the actual number of parity check symbols may be less than Lm.

We now show that if n is properly chosen, the resultant codes are cyclic codes. Let $m_j(x)$ be the minimum function of α_j, $j = 1, 2, \ldots, L$, where $\alpha_j \in GF(q^m)$ and $m_j(x)$ has coefficients from $GF(q)$. For Property 3 on page 858, every code polynomial $c(x)$ must be divisible by $m_1(x), m_2(x), \ldots, m_L(x)$, and is thus divisible by the least common multiple of such minimum functions, l.c.m.$(m_1(x), m_2(x), \ldots, m_L(x))$. If we define

$$g(x) = \text{l.c.m.}(m_1(x), m_2(x), \ldots, m_L(x)) \tag{11.163}$$

then all multiples of $g(x)$ are code words. In particular from Definition 11.12 the code is cyclic if

$$x^n - 1 = g(x)\, h(x) \tag{11.164}$$

Let ℓ_i be the order of α_i, $i = 1, 2, \ldots, L$, and furthermore let

$$n = \text{l.c.m.}(\ell_1, \ell_2, \ldots, \ell_L) \tag{11.165}$$

From the properties of the minimum function (see Property 2, page 861), we know that $g(x)$ divides $x^n - 1$; thus the code is cyclic if n is chosen as indicated by (11.165). We note that

$$r = \deg(g(x)) \leq mL \tag{11.166}$$

as $\deg(m_i(x)) \leq m$. We see that r is equal to mL if all minimum functions are distinct and are of degree m; conversely, $r < mL$ if any minimum function has degree less than m or if two or more minimum functions are identical.

Example 11.2.27
Choose $q = 2$ and let α be a primitive element of $GF(2^4)$; furthermore let the code polynomials have as roots the elements α, α^2, α^3, α^4. To derive the minimum functions of the chosen elements we look up for example the Appendix C of [3], where such functions are listed. Minimum functions and orders of elements chosen for this example are given in Table 11.15.

Then

$$\begin{aligned} g(x) &= (x^4 + x + 1)\,(x^4 + x^3 + x^2 + x + 1) \\ n &= \text{l.c.m.}(15, 15, 5, 15) = 15 \end{aligned} \tag{11.167}$$

The resultant code is therefore a (15,7) code; later we will show that $d_{min}^H = 5$.

Bose–Chaudhuri–Hocquenhem (BCH) codes

The BCH codes are error correcting codes with symbols from $GF(q)$ and roots of code polynomials from $GF(q^m)$.

Table 11.15 Minimum functions and orders of elements α, α^2, α^3, α^4, in $GF(2^4)$.

Roots	Minimum function	Order
α	$x^4 + x + 1$	15
α^2	$x^4 + x + 1$	15
α^3	$x^4 + x^3 + x^2 + x + 1$	5
α^4	$x^4 + x + 1$	15

The basic mathematical fact required to prove the error correcting capability of BCH codes is that if $\alpha_1, \alpha_2, \ldots, \alpha_r$ are elements from any field, the determinant of the Vandermonde matrix, given by

$$\det \begin{vmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_r \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_r^2 \\ \vdots & & & \vdots \\ \alpha_1^{r-1} & \alpha_2^{r-1} & \ldots & \alpha_r^{r-1} \end{vmatrix} \tag{11.168}$$

is non-zero if and only if $\alpha_i \neq \alpha_j$, for all indices $i \neq j$. In particular, we prove the following result.

Lemma. The determinant (11.168) is given by

$$D = \det \begin{vmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_r \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_r^2 \\ \vdots & & & \vdots \\ \alpha_1^{r-1} & \alpha_2^{r-1} & \ldots & \alpha_r^{r-1} \end{vmatrix} = (-1)^{\frac{r(r+1)}{2}} \prod_{\substack{i,\, j=1 \\ i<j}}^{r} (\alpha_i - \alpha_j) \tag{11.169}$$

Proof. Consider the polynomial $P(x)$ defined as

$$P(x) = \det \begin{vmatrix} 1 & 1 & \ldots & 1 \\ x & \alpha_2 & \ldots & \alpha_r \\ x^2 & \alpha_2^2 & \ldots & \alpha_r^2 \\ \vdots & & & \vdots \\ x^{r-1} & \alpha_2^{r-1} & \ldots & \alpha_r^{r-1} \end{vmatrix} \tag{11.170}$$

so that $D = P(\alpha_1)$. Now, $P(x)$ is a polynomial of degree at most $r-1$ whose zeros are $x = \alpha_2, x = \alpha_3, \ldots, x = \alpha_r$, because if $x = \alpha_i$, $i = 2, 3, \ldots, r$, the determinant D is equal to zero as two columns of the matrix are identical. Thus

$$P(x) = k_1(x - \alpha_2)(x - \alpha_3) \ldots (x - \alpha_r) \tag{11.171}$$

and

$$D = P(\alpha_1) = k_1(\alpha_1 - \alpha_2)(\alpha_1 - \alpha_3) \ldots (\alpha_1 - \alpha_r) \tag{11.172}$$

It remains to calculate k_1. The constant k_1 is the coefficient of x^{r-1}; therefore from (11.170) we get

$$(-1)^r k_1 = \det \begin{vmatrix} 1 & 1 & \ldots & 1 \\ \alpha_2 & \alpha_3 & \ldots & \alpha_r \\ \vdots & & & \vdots \\ \alpha_2^{r-2} & \alpha_3^{r-2} & \ldots & \alpha_r^{r-1} \end{vmatrix} = k_2(\alpha_2 - \alpha_3)(\alpha_2 - \alpha_4) \ldots (\alpha_2 - \alpha_r) \tag{11.173}$$

using a result similar to (11.172).

Proceeding we find

$$\begin{aligned}(-1)^{r-1}k_2 &= k_3(\alpha_3-\alpha_4)(\alpha_3-\alpha_5)\cdots(\alpha_3-\alpha_r)\\ (-1)^{r-2}k_3 &= k_4(\alpha_4-\alpha_5)(\alpha_4-\alpha_6)\cdots(\alpha_4-\alpha_r)\\ &\vdots\\ (-1)^2 k_{r-1} &= (-1)(\alpha_{r-1}-\alpha_r)\end{aligned} \tag{11.174}$$

and therefore

$$D=(-1)^{r+(r-1)+\cdots+2+1}\prod_{\substack{i,j=1\\ i<j}}^{r}(\alpha_i-\alpha_j)=(-1)^{\frac{r(r+1)}{2}}\prod_{\substack{i,j=1\\ i<j}}^{r}(\alpha_i-\alpha_j) \tag{11.175}$$

We now prove the important Bose–Chaudhuri–Hocquenhem theorem.

Theorem 11.2
Consider a code with symbols from $GF(q)$, whose code polynomials have as zeros the elements $\alpha^{m_0}, \alpha^{m_0+1}, \ldots, \alpha^{m_0+d-2}$, where α is any element from $GF(q^m)$ and m_0 is any integer. Then the resultant (n,k) cyclic code has the following properties:

a) it has minimum distance $d^H_{min} \geq d$ if the elements $\alpha^{m_0}, \alpha^{m_0+1}, \ldots, \alpha^{m_0+d-2}$, are distinct;

b) $n-k \leq (d-1)m$; if $q=2$ and $m_0=1$, then $n-k \leq \left\lceil \frac{d-1}{2} \right\rceil m$;

c) n is equal to the order of α, unless $d=2$, in which case n is equal to the order of α^{m_0};

d) $g(x)$ is equal to the least common multiple of the minimum functions of α^{m_0}, $\alpha^{m_0+1}, \ldots, \alpha^{m_0+d-2}$.

Proof. The proof of part d) has already been given (see (11.163)); the proof of part b) then follows by noting that each minimum function is at most of degree m, and there are at most $(d-1)$ distinct minimum functions. If $q=2$ and $m_0=1$, the minimum function of α raised to an even power, for example α^{2i}, is the same as the minimum function of α^i (see Property 2 on page 859), therefore there are at most $\left(\left\lceil \frac{d-1}{2} \right\rceil m\right)$ distinct minimum functions.

To prove part c) note that, if $d=2$, we have only the root α^{m_0}, so that n is equal to the order of α^{m_0}. If there is more than one root, then n must be the least common multiple of the order of the roots. If α^{m_0} and α^{m_0+1} are both roots, then $(\alpha^{m_0})^n=1$ and $(\alpha^{m_0+1})^n=1$, so that $\alpha^n=1$; thus n is a multiple of the order of α. On the other hand, if ℓ is the order of α, $(\alpha^{m_0+i})^\ell=(\alpha^\ell)^{m_0+i}=1^{m_0+i}=1$; therefore ℓ is a multiple of the order of every root. Then n is the least common multiple of numbers all of which divide ℓ, and therefore $n \leq \ell$; thus $n=\ell$.

Finally we prove part a). We note that the code words must satisfy the condition

$$\begin{bmatrix} 1 & \alpha^{m_0} & (\alpha^{m_0})^2 & \dots & (\alpha^{m_0})^{n-1} \\ 1 & \alpha^{m_0+1} & (\alpha^{m_0+1})^2 & \dots & (\alpha^{m_0+1})^{n-1} \\ \vdots & & & & \vdots \\ 1 & \alpha^{m_0+d-2} & (\alpha^{m_0+d-2})^2 & \dots & (\alpha^{m_0+d-2})^{n-1} \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{11.176}$$

We now show that no linear combination of $(d-1)$ or fewer columns is equal to 0. We do this by showing that the determinant of any set of $(d-1)$ columns is non-zero. Choose columns $j_1, j_2, \dots, j_{d-1}$; then

$$\det \begin{vmatrix} (\alpha^{m_0})^{j_1} & (\alpha^{m_0})^{j_2} & \dots & (\alpha^{m_0})^{j_{d-1}} \\ (\alpha^{m_0+1})^{j_1} & (\alpha^{m_0+1})^{j_2} & \dots & (\alpha^{m_0+1})^{j_{d-1}} \\ \vdots & & & \vdots \\ (\alpha^{m_0+d-2})^{j_1} & (\alpha^{m_0+d-2})^{j_2} & \dots & (\alpha^{m_0+d-2})^{j_{d-1}} \end{vmatrix} \tag{11.177}$$

$$= \alpha^{m_0(j_1+j_2+\cdots+j_{d-1})} \det \begin{vmatrix} 1 & 1 & \dots & 1 \\ \alpha^{j_1} & \alpha^{j_2} & \dots & \alpha^{j_{d-1}} \\ \vdots & & & \vdots \\ (\alpha^{j_1})^{d-2} & (\alpha^{j_2})^{d-2} & \dots & (\alpha^{j_{d-1}})^{d-2} \end{vmatrix} \tag{11.178}$$

$$= \alpha^{m_0(j_1+j_2+\cdots+j_{d-1})} (-1)^{\frac{(d-1)d}{2}} \prod_{\substack{i,k=1 \\ i<k}}^{d-1} (\alpha^{j_i} - \alpha^{j_k}) \neq 0 \tag{11.179}$$

Note that we have proven that $(d-1)$ columns of $\mathbf{H}$ are linearly independent even if they are multiplied by elements from $GF(q^m)$. All that would have been required was to show linear independence if the multipliers are from $GF(q)$.

Binary BCH codes

In this section we consider binary BCH codes. Choose $m_0 = 1$; then from Property c) of Theorem 11.2 we get

$$n = \begin{cases} 2^m - 1 & \text{if } \alpha \text{ is a primitive element of } GF(2^m) \\ \dfrac{2^m - 1}{c} & \text{if } \alpha = \beta^c, \text{ where } \beta \text{ is a primitive element of } GF(2^m) \end{cases} \tag{11.180}$$

and $r = n - k$ satisfies the relation (see Property b)

$$r \leq m \left\lceil \frac{d-1}{2} \right\rceil \tag{11.181}$$

Moreover for Property d)

$$g(x) = \text{l.c.m.(minimum functions of } \alpha, \alpha^3, \alpha^5, \dots, \alpha^{d-2}), \text{ with } d \text{ odd number} \tag{11.182}$$

Table 11.16 Minimum functions of the elements of $GF(2^6)$.

Roots	Minimum function
$\alpha^1\ \alpha^2\ \alpha^4\ \alpha^8\ \alpha^{16}\ \alpha^{32}$	x^6+x+1
$\alpha^3\ \alpha^6\ \alpha^{12}\ \alpha^{24}\ \alpha^{48}\ \alpha^{33}$	$x^6+x^4+x^2+x+1$
$\alpha^5\ \alpha^{10}\ \alpha^{20}\ \alpha^{40}\ \alpha^{17}\ \alpha^{34}$	$x^6+x^5+x^2+x+1$
$\alpha^7\ \alpha^{14}\ \alpha^{28}\ \alpha^{56}\ \alpha^{49}\ \alpha^{35}$	x^6+x^3+1
$\alpha^9\ \alpha^{18}\ \alpha^{36}$	x^3+x^2+1
$\alpha^{11}\ \alpha^{22}\ \alpha^{44}\ \alpha^{25}\ \alpha^{50}\ \alpha^{37}$	$x^6+x^5+x^3+x^2+1$
$\alpha^{13}\ \alpha^{26}\ \alpha^{52}\ \alpha^{41}\ \alpha^{19}\ \alpha^{38}$	$x^6+x^4+x^3+x+1$
$\alpha^{15}\ \alpha^{30}\ \alpha^{60}\ \alpha^{57}\ \alpha^{51}\ \alpha^{39}$	$x^6+x^5+x^4+x^2+1$
$\alpha^{21}\ \alpha^{42}$	x^2+x+1
$\alpha^{23}\ \alpha^{46}\ \alpha^{29}\ \alpha^{58}\ \alpha^{53}\ \alpha^{43}$	$x^6+x^5+x^4+x+1$
$\alpha^{27}\ \alpha^{54}\ \alpha^{45}$	x^3+x+1
$\alpha^{31}\ \alpha^{62}\ \alpha^{61}\ \alpha^{59}\ \alpha^{55}\ \alpha^{47}$	x^6+x^5+1

Example 11.2.28
Consider binary BCH codes of length 63, that is $q=2$ and $m=6$. To get a code with design distance d we choose as roots $\alpha, \alpha^2, \alpha^3, \ldots, \alpha^{d-1}$, where α is a primitive element from $GF(2^6)$. Using Table 11.11 on page 861, we get the minimum functions of the elements from $GF(2^6)$ given in Table 11.16.

Then the roots and generator polynomials for different values of d are given in Table 11.17; the parameters of the relative codes are given in Table 11.18.

Example 11.2.29
Let $q=2$, $m=6$, and choose as roots α, α^2, α^3, α^4, with $\alpha=\beta^3$, where β is a primitive element of $GF(2^6)$; then $n=\frac{2^m-1}{c}=\frac{63}{3}=21$, $d^H_{min}\geq d=5$, and

$$\begin{aligned} g(x) &= \text{l.c.m}(m_{\beta^3}(x), m_{\beta^6}(x), m_{\beta^9}(x), m_{\beta^{12}}(x)) \\ &= m_{\beta^3}(x)m_{\beta^9}(x) \\ &= (x^6+x^4+x^2+x+1)(x^3+x^2+1) \end{aligned} \tag{11.183}$$

As $r=\deg(g(x))=9$, then $k=n-r=12$; thus we obtain a (21,12) code.

Example 11.2.30
Let $q=2$, $m=4$, and choose as roots α, α^2, α^3, α^4, with α primitive element of $GF(2^4)$; then a (15,5) code is obtained having $d^H_{min}\geq 5$, and $g(x)=(x^4+x+1)(x^4+x^3+x^2+x+1)$.

Table 11.17 Roots and generator polynomials of BCH codes of length $n = 63 = 2^6 - 1$ for different values of d. α is a primitive element of $GF(2^6)$ (see (11.180)).

d	Roots	Generator polynomial
3	$\alpha\ \alpha^2$	$(x^6 + x + 1) = g_3(x)$
5	$\alpha\ \alpha^2\ \alpha^3\ \alpha^4$	$(x^6 + x + 1)(x^6 + x^4 + x^2 + x + 1) = g_5(x)$
7	$\alpha\ \alpha^2\ \ldots\ \alpha^6$	$(x^6 + x^5 + x^2 + x + 1)\, g_5(x) = g_7(x)$
9	$\alpha\ \alpha^2\ \ldots\ \alpha^8$	$(x^6 + x^3 + 1)\, g_7(x) = g_9(x)$
11	$\alpha\ \alpha^2\ \ldots\ \alpha^{10}$	$(x^3 + x^2 + 1)\, g_9(x) = g_{11}(x)$
13	$\alpha\ \alpha^2\ \ldots\ \alpha^{12}$	$(x^6 + x^5 + x^3 + x^2 + 1)\, g_{11}(x) = g_{13}(x)$
15	$\alpha\ \alpha^2\ \ldots\ \alpha^{14}$	$(x^6 + x^4 + x^3 + x + 1)\, g_{13}(x) = g_{15}(x)$
21	$\alpha\ \alpha^2\ \ldots\ \alpha^{20}$	$(x^6 + x^5 + x^4 + x^2 + 1)\, g_{15}(x) = g_{21}(x)$
23	$\alpha\ \alpha^2\ \ldots\ \alpha^{22}$	$(x^2 + x + 1)\, g_{21}(x) = g_{23}(x)$
27	$\alpha\ \alpha^2\ \ldots\ \alpha^{26}$	$(x^6 + x^5 + x^4 + x + 1)\, g_{23}(x) = g_{27}(x)$
31	$\alpha\ \alpha^2\ \ldots\ \alpha^{30}$	$(x^3 + x + 1)\, g_{27}(x) = g_{31}(x)$

Table 11.18 Parameters of BCH codes of length $n = 63$.

k	57	51	45	39	36	30	24	18	16	10	7
d	3	5	7	9	11	13	15	21	23	27	31
t	1	2	3	4	5	6	7	10	11	13	15

Example 11.2.31
Let $q = 2$, $m = 4$, and choose as roots α, α^2, α^3, α^4, α^5, α^6, with α primitive element of $GF(2^4)$; then a (15,5) code is obtained having $d_{min}^H \geq 7$, and $g(x) = (x^4 + x + 1)(x^4 + x^3 + x^2 + x + 1)(x^2 + x + 1)$.

Reed–Solomon codes

Reed–Solomon codes represent a particular case of BCH codes obtained by choosing $m = 1$; in other words, the field $GF(q)$ and the extension field $GF(q^m)$ coincide. Choosing α as a primitive element of (11.180) we get

$$n = q^m - 1 = q - 1 \tag{11.184}$$

Note that the minimum function with coefficients in $GF(q)$ of an element α^i from $GF(q)$ is

$$m_{\alpha^i}(x) = (x - \alpha^i) \tag{11.185}$$

For $m_0 = 1$, if we choose the roots $\alpha, \alpha^2, \alpha^3, \ldots, \alpha^{d-1}$, then

$$g(x) = (x - \alpha)(x - \alpha^2) \ldots (x - \alpha^{d-1}) \tag{11.186}$$

so that $r = (d - 1)$; the block length n is given by the order of α. In this case we show that $d_{min}^H = d$; in fact, for any code we have $d_{min}^H \leq r + 1$, as we can always choose a code word with every message symbol but one equal to zero and therefore its weight is at most equal to $r + 1$; from this it follows that $d_{min}^H \leq d$, but from the BCH theorem we know that $d_{min}^H \geq d$.

Example 11.2.32
Choose α as a primitive element of $GF(2^5)$, and choose the roots α, α^2, α^3, α^4, α^5, and α^6; then the resultant (31,25) code has $d_{min}^H = 7$, $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^3)(x - \alpha^4)(x - \alpha^5)(x - \alpha^6)$, and the symbols of the code words are from $GF(2^5)$.

Observation 11.2
The encoding of Reed–Solomon codes can be done by the devices of Figure 11.3 or Figure 11.4, where the operations are in $GF(q)$. In Table 11.19 and Table 11.20 we give, respectively, the tables of additions and multiplications between elements of $GF(q)$ for $q =$

Table 11.19 Addition table for the elements of $GF(2^4)$.

$+$	0	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}
0	0	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}
α^0	α^0	0	α^4	α^8	α^{14}	α^1	α^{10}	α^{13}	α^9	α^2	α^7	α^5	α^{12}	α^{11}	α^6	α^3
α^1	α^1	α^4	0	α^5	α^9	α^0	α^2	α^{11}	α^{14}	α^{10}	α^3	α^8	α^6	α^{13}	α^{12}	α^7
α^2	α^2	α^8	α^5	0	α^6	α^{10}	α^1	α^3	α^{12}	α^0	α^{11}	α^4	α^9	α^7	α^{14}	α^{13}
α^3	α^3	α^{14}	α^9	α^6	0	α^7	α^{11}	α^2	α^4	α^{13}	α^1	α^{12}	α^5	α^{10}	α^8	α^0
α^4	α^4	α^1	α^0	α^{10}	α^7	0	α^8	α^{12}	α^3	α^5	α^{14}	α^2	α^{13}	α^6	α^{11}	α^9
α^5	α^5	α^{10}	α^2	α^1	α^{11}	α^8	0	α^9	α^{13}	α^4	α^6	α^0	α^3	α^{14}	α^7	α^{12}
α^6	α^6	α^{13}	α^{11}	α^3	α^2	α^{12}	α^9	0	α^{10}	α^{14}	α^5	α^7	α^1	α^4	α^0	α^8
α^7	α^7	α^9	α^{14}	α^{12}	α^4	α^3	α^{13}	α^{10}	0	α^{11}	α^0	α^6	α^8	α^2	α^5	α^1
α^8	α^8	α^2	α^{10}	α^0	α^{13}	α^5	α^4	α^{14}	α^{11}	0	α^{12}	α^1	α^7	α^9	α^3	α^6
α^9	α^9	α^7	α^3	α^{11}	α^1	α^{14}	α^6	α^5	α^0	α^{12}	0	α^{13}	α^2	α^8	α^{10}	α^4
α^{10}	α^{10}	α^5	α^8	α^4	α^{12}	α^2	α^0	α^7	α^6	α^1	α^{13}	0	α^{14}	α^3	α^9	α^{11}
α^{11}	α^{11}	α^{12}	α^6	α^9	α^5	α^{13}	α^3	α^1	α^8	α^7	α^2	α^{14}	0	α^0	α^4	α^{10}
α^{12}	α^{12}	α^{11}	α^{13}	α^7	α^{10}	α^6	α^{14}	α^4	α^2	α^9	α^8	α^3	α^0	0	α^1	α^5
α^{13}	α^{13}	α^6	α^{12}	α^{14}	α^8	α^{11}	α^7	α^0	α^5	α^3	α^{10}	α^9	α^4	α^1	0	α^2
α^{14}	α^{14}	α^3	α^7	α^{13}	α^0	α^9	α^{12}	α^8	α^1	α^6	α^4	α^{11}	α^{10}	α^5	α^2	0

Table 11.20 Multiplication table for the elements of $GF(2^4)$.

$\cdot$	0	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
α^0	0	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}
α^1	0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0
α^2	0	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1
α^3	0	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2
α^4	0	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3
α^5	0	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4
α^6	0	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5
α^7	0	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6
α^8	0	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7
α^9	0	α^9	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8
α^{10}	0	α^{10}	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9
α^{11}	0	α^{11}	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}
α^{12}	0	α^{12}	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}
α^{13}	0	α^{13}	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}
α^{14}	0	α^{14}	α^0	α^1	α^2	α^3	α^4	α^5	α^6	α^7	α^8	α^9	α^{10}	α^{11}	α^{12}	α^{13}

2^4; the conversion of the symbol representation from binary to exponential is implemented using Table 11.21. We note that the encoding operations can be performed by interpreting the field elements as polynomials with coefficients from $GF(2)$, and applying the polynomial arithmetic $\bmod x^4 + x + 1$ (see Figure 11.1 and Figure 11.2).

Decoding of BCH codes

Suppose we transmit the code polynomial $c_0 + c_1 x + \cdots + c_{n-1}x^{n-1}$, and we receive $z_0 + z_1 x + \cdots + z_{n-1}x^{n-1}$. We define the polynomial $e(x) = (z_0 - c_0) + (z_1 - c_1)x + \cdots + (z_{n-1} - c_{n-1})x^{n-1} = e_0 + e_1 x + \cdots + e_{n-1}x^{n-1}$, where e_i are elements from $GF(q^m)$, the field in which parity control is performed. If we express polynomials as vectors we obtain $\mathbf{z} = \mathbf{c} + \mathbf{e}$; furthermore we recall that, defining the matrix

$$\mathbf{H} = \begin{bmatrix} 1 & \alpha^{m_0} & (\alpha^{m_0})^2 & \dots & (\alpha^{m_0})^{n-1} \\ 1 & \alpha^{m_0+1} & (\alpha^{m_0+1})^2 & \dots & (\alpha^{m_0+1})^{n-1} \\ \vdots & & & & \vdots \\ 1 & \alpha^{m_0+d-2} & (\alpha^{m_0+d-2})^2 & \dots & (\alpha^{m_0+d-2})^{n-1} \end{bmatrix} \tag{11.187}$$

Table 11.21 Three equivalent representations of the elements of $GF(2^4)$, obtained applying the polynomial arithmetic modulo $x^4 + x + 1$.

Exponential	Polynomial	Binary $(x^0 x^1 x^2 x^3)$
0	0	0 0 0 0
α^0	1	1 0 0 0
α^1	x	0 1 0 0
α^2	x^2	0 0 1 0
α^3	x^3	0 0 0 1
α^4	$1 + x$	1 1 0 0
α^5	$x + x^2$	0 1 1 0
α^6	$x^2 + x^3$	0 0 1 1
α^7	$1 + x + x^3$	1 1 0 1
α^8	$1 + x^2$	1 0 1 0
α^9	$x + x^3$	0 1 0 1
α^{10}	$1 + x + x^2$	1 1 1 0
α^{11}	$x + x^2 + x^3$	0 1 1 1
α^{12}	$1 + x + x^2 + x^3$	1 1 1 1
α^{13}	$1 + x^2 + x^3$	1 0 1 1
α^{14}	$1 + x^3$	1 0 0 1

we obtain

$$\mathbf{Hz} = \mathbf{He} = \mathbf{s} = \begin{bmatrix} s_{m_0} \\ s_{m_0+1} \\ s_{m_0+2} \\ \vdots \\ s_{m_0+d-2} \end{bmatrix} \tag{11.188}$$

where

$$s_j = \sum_{\ell=0}^{n-1} e_\ell (\alpha^j)^\ell \qquad j = m_0, m_0 + 1, \ldots, m_0 + d - 2 \tag{11.189}$$

Assuming there are ν errors, that is $w(\mathbf{e}) = \nu$, we can write

$$s_j = \sum_{i=1}^{\nu} \varepsilon_i (\xi_i)^j \qquad j = m_0, m_0 + 1, \ldots, m_0 + d - 2 \tag{11.190}$$

where the coefficients ε_i are elements from $GF(q^m)$ that represent the values of the errors, and the coefficients ξ_i are elements from $GF(q^m)$ that give the positions of the errors. In other words, if $\xi_i = \alpha^\ell$ then an error has occurred in position ℓ, where $\ell \in \{0, 1, 2, \ldots, n-1\}$. The idea of a decoding algorithm is to solve the set of non-linear equations for the unknowns ε_i and ξ_i; then there are 2ν unknowns and $d-1$ equations.

We show that it is possible to solve this set of equations if $2\nu \leq d-1$, assuming $m_0 = 1$; in the case $m_0 \neq 1$, the decoding procedure does not change.

Consider the polynomial in x, also called *error indicator polynomial*,

$$\lambda(x) = \lambda_\nu x^\nu + \lambda_{\nu-1} x^{\nu-1} + \cdots + \lambda_1 x + 1 \tag{11.191}$$

defined as the polynomial that has as zeros the inverse of the elements that locate the positions of the errors, that is ξ_i^{-1}, $i = 1, \ldots, \nu$. Then

$$\lambda(x) = (1 - x\xi_1)(1 - x\xi_2) \ldots (1 - x\xi_\nu) \tag{11.192}$$

If the coefficients of $\lambda(x)$ are known, it is possible to find the zeros of $\lambda(x)$, and thus determine the positions of the errors.

The first step of the decoding procedure consists in evaluating the coefficients $\lambda_1, \ldots, \lambda_\nu$ using the syndromes (11.190). We multiply both sides of (11.191) by $\varepsilon_i \xi_i^{j+\nu}$ and evaluate the expression found for $x = \xi_i^{-1}$, obtaining

$$0 = \varepsilon_i \xi_i^{j+\nu} (1 + \lambda_1 \xi_i^{-1} + \lambda_2 \xi_i^{-2} + \cdots + \lambda_\nu \xi_i^{-\nu}) \tag{11.193}$$

which can be written as

$$\varepsilon_i (\xi_i^{j+\nu} + \lambda_1 \xi_i^{j+\nu-1} + \lambda_2 \xi_i^{j+\nu-2} + \cdots + \lambda_\nu \xi_i^j) = 0 \tag{11.194}$$

(11.194) holds for $i = 1, \ldots, \nu$, and for every value of j. Adding these equations for $i = 1, \ldots, \nu$, we get

$$\sum_{i=1}^{\nu} \varepsilon_i (\xi_i^{j+\nu} + \lambda_1 \xi_i^{j+\nu-1} + \lambda_2 \xi_i^{j+\nu-2} + \cdots + \lambda_\nu \xi_i^j) = 0 \qquad \text{for every } j \tag{11.195}$$

or equivalently

$$\sum_{i=1}^{\nu} \varepsilon_i \xi_i^{j+\nu} + \lambda_1 \sum_{i=1}^{\nu} \varepsilon_i \xi_i^{j+\nu-1} + \lambda_2 \sum_{i=1}^{\nu} \varepsilon_i \xi_i^{j+\nu-2} + \cdots + \lambda_\nu \sum_{i=1}^{\nu} \varepsilon_i \xi_i^j = 0 \quad \text{for every } j \tag{11.196}$$

As $\nu \leq (d-1)/2$, if $1 \leq j \leq \nu$ the equations in (11.196) are equal to the syndromes (11.190); therefore we obtain

$$\lambda_1 s_{j+\nu-1} + \lambda_2 s_{j+\nu-2} + \cdots + \lambda_\nu s_j = -s_{j+\nu} \qquad j = 1, \ldots, \nu \tag{11.197}$$

(11.197) is a system of linear equations that can be written in the form

$$\begin{bmatrix} s_1 & s_2 & s_3 & \ldots & s_{\nu-1} & s_\nu \\ s_2 & s_3 & s_4 & \ldots & s_\nu & s_{\nu+1} \\ \vdots & & & & \vdots & \vdots \\ s_\nu & s_{\nu+1} & s_{\nu+2} & \ldots & s_{2\nu-2} & s_{2\nu-1} \end{bmatrix} \begin{bmatrix} \lambda_\nu \\ \lambda_{\nu-1} \\ \vdots \\ \lambda_1 \end{bmatrix} = \begin{bmatrix} -s_{\nu+1} \\ -s_{\nu+2} \\ \vdots \\ -s_{2\nu} \end{bmatrix} \tag{11.198}$$

Thus we have obtained ν equations in the unknowns $\lambda_1, \lambda_2, \ldots, \lambda_\nu$. To show that these equations are linearly independent we see that, from (11.190), the matrix of the coefficients can be factorized as

$$\begin{bmatrix} s_1 & s_2 & s_3 & \ldots & s_{\nu-1} & s_\nu \\ s_2 & s_3 & s_4 & \ldots & s_\nu & s_{\nu+1} \\ \vdots & & & & \vdots & \vdots \\ s_\nu & s_{\nu+1} & s_{\nu+2} & \ldots & s_{2\nu-2} & s_{2\nu-1} \end{bmatrix} =$$

$$\begin{bmatrix} 1 & 1 & 1 & \ldots & 1 \\ \xi_1 & \xi_2 & \xi_3 & \ldots & \xi_\nu \\ \vdots & & & & \vdots \\ \xi_1^{\nu-1} & \xi_2^{\nu-1} & \xi_3^{\nu-1} & \ldots & \xi_\nu^{\nu-1} \end{bmatrix} \begin{bmatrix} \varepsilon_1\xi_1 & & & \\ & \varepsilon_2\xi_2 & & 0 \\ & & \ddots & \\ & 0 & & \varepsilon_\nu\xi_\nu \end{bmatrix} \begin{bmatrix} 1 & \xi_1 & \ldots & \xi_1^{\nu-1} \\ 1 & \xi_2 & \ldots & \xi_2^{\nu-1} \\ \vdots & & & \vdots \\ 1 & \xi_\nu & \ldots & \xi_\nu^{\nu-1} \end{bmatrix} \tag{11.199}$$

The matrix of the coefficients has a non-zero determinant if each of the matrices on the right-hand side of (11.199) has a non-zero determinant. We note that the first and third matrix are Vandermonde matrices, hence they are non-singular if $\xi_i \neq \xi_m$, $i \neq m$; the second matrix is also non-singular, as it is a diagonal matrix with non-zero terms on the diagonal. This assumes we have at most ν errors, with $\nu \leq (d-1)/2$. As ν is arbitrary, we initially choose $\nu = (d-1)/2$ and compute the determinant of the matrix (11.199). If this is non-zero, then we have the correct value of ν; otherwise, if the determinant is zero, we reduce ν by one and repeat the computation. We proceed until we obtain a non-zero determinant, thus finding the number of errors that occurred.

After finding the solution for $\lambda_1, \ldots, \lambda_\nu$ (see (11.198)), we obtain the positions of the errors by finding the zeros of the polynomial $\lambda(x)$ (see (11.192)). Note that an exhaustive method to search for the ν zeros of the polynomial $\lambda(x)$ requires that at most n possible roots of the type α^ℓ, $\ell \in \{0, 1, \ldots, n-1\}$, are taken into consideration. We now compute the value of the non-zero elements ε_i, $i = 1, \ldots, \nu$, of the vector $\mathbf{e}$. If the code is binary, the components of the vector $\mathbf{e}$ are immediately known, otherwise we solve the system of linear equations (11.190) for the ν unknowns $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_\nu$. The determinant of the matrix of the system of linear equations is given by

$$\det \begin{bmatrix} \xi_1 & \xi_2 & \ldots & \xi_\nu \\ \xi_1^2 & \xi_2^2 & \ldots & \xi_\nu^2 \\ \vdots & \vdots & & \vdots \\ \xi_1^\nu & \xi_2^\nu & \ldots & \xi_\nu^\nu \end{bmatrix} = \xi_1\xi_2\ldots\xi_\nu \det \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \xi_1 & \xi_2 & \ldots & \xi_\nu \\ \vdots & & & \vdots \\ \xi_1^{\nu-1} & \xi_2^{\nu-1} & \ldots & \xi_\nu^{\nu-1} \end{bmatrix} \tag{11.200}$$

The determinant of the Vandermonde matrix in (11.200) is non-zero if ν errors occurred, as the elements $\xi_1, \ldots, \xi_\nu$ are non-zero and distinct.

In summary, the original system of non-linear equations (11.190) is solved in the following three steps.

Step 1: find the coefficients $\lambda_1, \lambda_2, \ldots, \lambda_\nu$ of the error indicator polynomial by solving a system of linear equations.

Step 2: find the ν roots $\xi_1, \xi_2, \ldots, \xi_\nu$ of a polynomial of degree ν.

Step 3: find the values of the errors $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_\nu$ by solving a system of linear equations.

For binary codes the last step is omitted.

Efficient decoding of BCH codes

The computational complexity for the decoding of BCH codes illustrated in the previous section lies mainly in the solution of the systems of linear equations (11.198) and (11.190). For small values of ν, the direct solution of these systems of equations by inverting matrices does not require a high computational complexity; we recall that the number of operations necessary to invert a $\nu \times \nu$ matrix is of the order of ν^3. However, in many applications it is necessary to resort to codes that are capable of correcting several errors, and it is thus desirable to find more efficient methods for the solution. The method developed by Berlekamp is based on the observation that the matrix of the coefficients and the known data vector in (11.198) have a particular structure. Assuming that the vector $\boldsymbol{\lambda} = [\lambda_\nu, \lambda_{\nu-1}, \ldots, \lambda_1]^T$ is known, then from (11.197) for the sequence of syndromes $s_1, s_2, \ldots, s_{2\nu}$, the recursive relation holds

$$s_j = -\sum_{i=1}^{\nu} \lambda_i \, s_{j-i} \qquad j = \nu + 1, \ldots, 2\nu \tag{11.201}$$

For a given $\boldsymbol{\lambda}$, (11.201) is the equation of a recursive filter, which can be implemented by a *shift register with feedback*, whose coefficients are given by $\boldsymbol{\lambda}$, as illustrated in Figure 11.8. The solution of (11.198) is thus equivalent to the problem of finding the shift register with feedback of minimum length that, if suitably initialized, yields the sequence of syndromes. This will identify the polynomial $\lambda(x)$ of minimum degree ν, that we recall exists and is unique, as the $\nu \times \nu$ matrix of the original problem admits the inverse.

The *Berlekamp–Massey algorithm* to find the recursive filter can be applied in any field and does not make use of the particular properties of the sequence of syndromes $s_1, s_2, \ldots, s_{d-1}$. To determine the recursive filter we must find two quantities, that we denote as $(L, \lambda(x))$, where L is the length of the shift register and $\lambda(x)$ is the polynomial whose degree ν must satisfy the condition $\nu \leq L$. The algorithm is inductive, that is for each r, starting from $r = 1$, we determine a shift register that generates the first r syndromes.

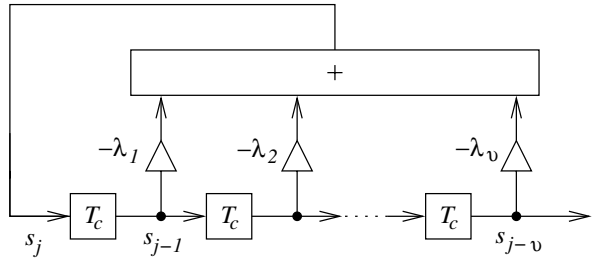


Figure 11.8. Recursive filter to compute syndromes (see (11.201)).

The shift register identified by $(L_r, \lambda^{(r)}(x))$ will then be a shift register of minimum length that generates the sequence $s_1, \ldots, s_r$.

Berlekamp–Massey algorithm. Let $s_1, \ldots, s_{d-1}$ be a sequence of elements from any field. Assuming the initial conditions $\lambda^{(0)}(x) = 1$, $\beta^{(0)}(x) = 1$, and $L_0 = 0$, we use the following set of recursive equations to determine $\lambda^{(d-1)}(x)$:

for $r = 1, \ldots, d-1$,

$$\Delta_r = \sum_{j=0}^{n-1} \lambda_j^{(r-1)} s_{r-j} \tag{11.202}$$

$$\delta_r = \begin{cases} 1 & \text{if } \Delta_r \neq 0 \text{ and } 2L_{r-1} \leq r-1 \\ 0 & \text{otherwise} \end{cases} \tag{11.203}$$

$$L_r = \delta_r (r - L_{r-1}) + (1 - \delta_r) L_{r-1} \tag{11.204}$$

$$\begin{bmatrix} \lambda^{(r)}(x) \\ \beta^{(r)}(x) \end{bmatrix} = \begin{bmatrix} 1 & -\Delta_r x \\ \Delta_r^{-1}\delta_r & (1-\delta_r)\, x \end{bmatrix} \begin{bmatrix} \lambda^{(r-1)}(x) \\ \beta^{(r-1)}(x) \end{bmatrix} \tag{11.205}$$

Then $\lambda^{(d-1)}(x)$ is the polynomial of minimum degree such that

$$\lambda_0^{(d-1)} = 1 \tag{11.206}$$

and

$$s_r = -\sum_{j=1}^{n-1} \lambda_j^{(d-1)} s_{r-j} \qquad r = L_{d-1} + 1, \ldots, d-1 \tag{11.207}$$

Note that Δ_r can be zero only if $\delta_r = 0$; in this case we assign to $\Delta_r^{-1} \cdot \delta_r$ the value zero. Moreover, we see that the algorithm requires a complexity of the order of d^2 operations, against a complexity of the order of d^3 operations needed by the matrix inversion in (11.198). To prove that the polynomial $\lambda^{(d-1)}(x)$ given by the algorithm is indeed the polynomial of minimum degree with $\lambda_0^{(d-1)} = 1$ that satisfies (11.207), we use the following two lemmas [2].

In Lemma 1 we find the relation between the lengths of the shift registers of minimum length obtained in two consecutive iterations, L_r and L_{r-1}. In Lemma 2 we use the algorithm to construct a shift register that generates $s_1, \ldots, s_r$ starting from the shift register of minimum length that generates $s_1, \ldots, s_{r-1}$. We will conclude that the construction yields the shift register of minimum length since it satisfies Lemma 1.

Lemma 1. We assume that $(L_{r-1}, \lambda^{(r-1)}(x))$ is the shift register of minimum length that generates $s_1, \ldots, s_{r-1}$, while $(L_r, \lambda^{(r)}(x))$ is the shift register of minimum length that generates $s_1, \ldots, s_{r-1}, s_r$, and $\lambda^{(r)}(x) \neq \lambda^{(r-1)}(x)$; then

$$L_r \geq \max(L_{r-1}, r - L_{r-1}) \tag{11.208}$$

Proof. The inequality (11.208) is the combination of the two inequalities $L_r \geq L_{r-1}$ and $L_r \geq r - L_{r-1}$. The first inequality is obvious, because if a shift register generates a certain sequence it must also generate any initial part of this sequence; the second inequality is obvious if $L_{r-1} \geq r$, because L_r is a non-negative quantity. Thus we assume $L_{r-1} < r$, and suppose that the second inequality is not satisfied; then $L_r \leq r - 1 - L_{r-1}$, or $r \geq L_{r-1} + L_r + 1$. By assumption we have

$$\begin{cases} s_j = -\sum_{i=1}^{L_{r-1}} \lambda_i^{(r-1)} s_{j-i} & j = L_{r-1}+1, \ldots, r-1 \\ s_r \neq -\sum_{i=1}^{L_{r-1}} \lambda_i^{(r-1)} s_{r-i} & \end{cases} \tag{11.209}$$

and

$$s_j = -\sum_{k=1}^{L_r} \lambda_k^{(r)} s_{j-k} \qquad j = L_r + 1, \ldots, r \tag{11.210}$$

We observe that

$$s_r = -\sum_{k=1}^{L_r} \lambda_k^{(r)} s_{r-k} = \sum_{k=1}^{L_r} \lambda_k^{(r)} \sum_{i=1}^{L_{r-1}} \lambda_i^{(r-1)} s_{r-k-i} \tag{11.211}$$

where the expression of s_{r-k} is valid, as $(r-k)$ goes from $r-1$ to $r-L_r$. Hence it belongs to the set $L_{r-1}+1, \ldots, r-1$, as it is assumed $r \geq L_{r-1} + L_r + 1$.

Furthermore

$$s_r \neq -\sum_{i=1}^{L_{r-1}} \lambda_i^{(r-1)} s_{r-i} = \sum_{i=1}^{L_{r-1}} \lambda_i^{(r-1)} \sum_{k=1}^{L_r} \lambda_k^{(r)} s_{r-i-k} \tag{11.212}$$

where the expression of s_{r-i} is valid as $(r-i)$ goes from $r-1$ to $r-L_{r-1}$. Hence it belongs to the set $L_r + 1, \ldots, r-1$, as it is assumed $r \geq L_{r-1} + L_r + 1$. The summations on the right-hand side of (11.212) can be exchanged, thus obtaining the right-hand side of (11.211). But this yields s_r of (11.211) different from s_r of (11.212), thus we get a contradiction.

Lemma 2. We assume that $(L_i, \lambda^{(i)}(x))$, $i = 1, \ldots, r$, identifies a sequence of shift registers of minimum length such that $\lambda^{(i)}(x)$ generates $s_1, \ldots, s_i$. If $\lambda^{(r)}(x) \neq \lambda^{(r-1)}(x)$, then

$$L_r = \max(L_{r-1}, r - L_{r-1}) \tag{11.213}$$

and every shift register that generates $s_1, \ldots, s_r$, and has a length that satisfies (11.213), is a shift register of minimum length. The Berlekamp–Massey algorithm yields this shift register.

Proof. From Lemma 1, L_r cannot be smaller than the right-hand side of (11.213); thus, if we construct a shift register that yields the given sequence and whose length satisfies

(11.213), then it must be a shift register of minimum length. The proof is obtained by induction.

We construct a shift register that satisfies the Lemma at the r-th iteration, assuming that shift registers were iteratively constructed for each value of the index k, with $k \leq r-1$. For each k, $k = 1, \ldots, r-1$, let $(L_k, \lambda^{(k)}(x))$ be the shift register of minimum length that generates $s_1, \ldots, s_k$. We assume that

$$L_k = \max(L_{k-1}, k - L_{k-1}) \qquad k = 1, \ldots, r-1 \tag{11.214}$$

Equation (11.214) is verified for $k = 0$, as $L_0 = 0$ and $L_1 = 1$. Let m be the index k at the most recent iteration that required a variation in the length of the shift register. In other words, at the end of the $(r-1)$-th iteration, m is the integer such that

$$L_{r-1} = L_m > L_{m-1} \tag{11.215}$$

From (11.209) and (11.202), we have that

$$s_j + \sum_{i=1}^{L_{r-1}} \lambda_i^{(r-1)} s_{j-i} = \sum_{i=0}^{L_{r-1}} \lambda_i^{(r-1)} s_{j-i} = \begin{cases} 0 & j = L_{r-1}, \ldots, r-1 \\ \Delta_r & j = r \end{cases} \tag{11.216}$$

If $\Delta_r = 0$, then the shift register $(L_{r-1}, \lambda^{(r-1)}(x))$ also generates the first r symbols of the sequence, hence

$$\begin{cases} L_r = L_{r-1} \\ \lambda^{(r)}(x) = \lambda^{(r-1)}(x) \end{cases} \tag{11.217}$$

If $\Delta_r \neq 0$, then it is necessary to find a new shift register. Recall from (11.215) that there was a variation in the length of the shift register for $k = m$; therefore

$$s_j + \sum_{i=1}^{L_{m-1}} \lambda_i^{(m-1)} s_{j-i} = \begin{cases} 0 & j = L_{m-1}, \ldots, m-1 \\ \Delta_m \neq 0 & j = m \end{cases} \tag{11.218}$$

and by induction,

$$L_{r-1} = L_m = \max(L_{m-1}, m - L_{m-1}) = m - L_{m-1} \tag{11.219}$$

as $L_m > L_{m-1}$. We now choose the new polynomial

$$\lambda^{(r)}(x) = \lambda^{(r-1)}(x) - \Delta_r \, \Delta_m^{-1} \, x^{r-m} \, \lambda^{(m-1)}(x) \tag{11.220}$$

and let $L_r = \deg(\lambda^{(r)}(x))$. Then, as $\deg(\lambda^{(r-1)}(x)) \leq L_{r-1}$, and $\deg[x^{r-m}\lambda^{(m-1)}(x)] \leq r - m + L_{m-1}$, we obtain

$$L_r \leq \max(L_{r-1}, r - m + L_{m-1}) \leq \max(L_{r-1}, r - L_{r-1}) \tag{11.221}$$

Thus, recalling Lemma 1, if $\lambda^{(r)}(x)$ generates $s_1, \ldots, s_r$, then

$$L_r = \max(L_{r-1}, r - L_{r-1}) \tag{11.222}$$

It remains to prove that the shift register $(L_r, \lambda^{(r)}(x))$ generates the given sequence. By direct computation we obtain

$$\begin{aligned} & s_j - \left(- \sum_{i=1}^{L_r} \lambda_i^{(r)} \, s_{j-i} \right) \\ & \quad = s_j + \sum_{i=1}^{L_{r-1}} \lambda_i^{(r-1)} \, s_{j-i} - \Delta_r \Delta_m^{-1} \left[s_{j-r+m} + \sum_{i=1}^{L_{m-1}} \lambda_i^{(L_{m-1})} \, s_{j-r+m-i} \right] \qquad (11.223) \\ & \quad = \begin{cases} 0 & j = L_r, L_r + 1, \ldots, r-1 \\ \Delta_r - \Delta_r \Delta_m^{-1} \Delta_m = 0 & j = r \end{cases} \end{aligned}$$

Therefore the shift register $(L_r, \lambda^{(r)}(x))$ generates $s_1, \ldots, s_r$. In particular, $(L_{d-1}, \lambda^{(d-1)}(x))$ generates $s_1, \ldots, s_{d-1}$. This completes the proof of Lemma 2.

We have seen that the computational complexity for the solution of the system of equations (11.198), that yields the error indicator polynomial, can be reduced by the Berlekamp–Massey algorithm. We now consider the system of equations (11.190) that yields the values of the errors. The computation of the inverse matrix can be avoided in the solution of (11.190) by applying the Forney algorithm [2]. We recall the expression (11.191) of the error indicator polynomial $\lambda(x)$, that has zeros for $x = \xi_i^{-1}$, $i = 1, \ldots, \nu$, given by

$$\lambda(x) = \prod_{\ell=1}^{\nu} (1 - x\xi_\ell) \qquad (11.224)$$

Define the syndrome polynomial as

$$s(x) = \sum_{j=1}^{d-1} s_j \, x^j = \sum_{j=1}^{d-1} \sum_{j=1}^{\nu} \varepsilon_i \, \xi_i^j \, x^j \qquad (11.225)$$

and furthermore define the *error evaluator polynomial* $\omega(x)$ as

$$\omega(x) = s(x) \, \lambda(x) \qquad \mod x^{d-1} \qquad (11.226)$$

Proposition. The error evaluator polynomial can be expressed as

$$\omega(x) = x \sum_{i=1}^{\nu} \varepsilon_i \, \xi_i \prod_{\substack{j=1 \\ j \neq i}}^{\nu} (1 - \xi_j x) \qquad (11.227)$$

Proof. From the definition (11.226) of $\omega(x)$, we obtain

$$\begin{aligned} \omega(x) &= \left[\sum_{j=1}^{d-1} \sum_{i=1}^{\nu} \varepsilon_i \, \xi_i^j \, x^j \right] \left[\prod_{\ell=1}^{\nu} (1 - \xi_\ell x) \right] \qquad \mod x^{d-1} \\ &= \sum_{i=1}^{\nu} \varepsilon_i \, \xi_i \, x \left[(1 - \xi_i x) \sum_{j=1}^{d-1} (\xi_i x)^{j-1} \right] \prod_{\substack{\ell=1 \\ \ell \neq i}}^{\nu} (1 - \xi_\ell x) \qquad \mod x^{d-1} \end{aligned} \qquad (11.228)$$

By inspection we see that the term within brackets is equal to $(1 - \xi_i^{d-1} x^{d-1})$; thus

$$\omega(x) = \sum_{i=1}^{\nu} \varepsilon_i \, \xi_i \, x(1 - \xi_i^{d-1} x^{d-1}) \prod_{\substack{\ell=1 \\ \ell \neq i}}^{\nu} (1 - \xi_\ell x) \qquad \bmod x^{d-1} \tag{11.229}$$

We now observe that (11.229), evaluated modulo x^{d-1}, is identical to (11.227).

Forney algorithm. We introduce the derivative of $\lambda(x)$ given by

$$\lambda'(x) = -\sum_{i=1}^{\nu} \xi_i \prod_{\substack{j=1 \\ j \neq i}}^{\nu} (1 - x\xi_j) \tag{11.230}$$

The values of the errors are given by

$$\varepsilon_\ell = \frac{\omega(\xi_\ell^{-1})}{\displaystyle\prod_{\substack{j=1 \\ j \neq \ell}}^{\nu} (1 - \xi_j \xi_\ell^{-1})} = -\frac{\omega(\xi_\ell^{-1})}{\xi_\ell^{-1} \lambda'(\xi_\ell^{-1})} \tag{11.231}$$

Proof. We evaluate (11.227) for $x = \xi_\ell^{-1}$, obtaining

$$\omega(\xi_\ell^{-1}) = \varepsilon_\ell \prod_{\substack{j=1 \\ j \neq \ell}}^{\nu} (1 - \xi_j \xi_\ell^{-1}) \tag{11.232}$$

which proves the first part of equality (11.231). Moreover, from (11.230), we have

$$\lambda'(\xi_\ell^{-1}) = -\xi_\ell \prod_{\substack{j=1 \\ j \neq \ell}}^{\nu} (1 - \xi_j \xi_\ell^{-1}) \tag{11.233}$$

which proves the second part of equality (11.231).

Example 11.2.33 (Reed–Solomon (15,9) code with $d = 7$ ($t = 3$), and elements from $GF(2^4)$) From (11.186), using Table 11.19 and Table 11.20, the generator polynomial is given by

$$\begin{aligned} g(x) &= (x - \alpha)(x - \alpha^2)(x - \alpha^3)(x - \alpha^4)(x - \alpha^5)(x - \alpha^6) \\ &= x^6 + \alpha^{10} x^5 + \alpha^{14} x^4 + \alpha^4 x^3 + \alpha^6 x^2 + \alpha^9 x + \alpha^6 \end{aligned} \tag{11.234}$$

Suppose that the code polynomial $c(x) = 0$ is transmitted, and that the received polynomial is

$$z(x) = \alpha x^7 + \alpha^5 x^5 + \alpha^{11} x^2 \tag{11.235}$$

In this case $e(x) = z(x)$. From (11.189), using Table 11.19 and Table 11.20, the syndromes are

$$\begin{aligned} s_1 &= \alpha\alpha^7 + \alpha^5\alpha^5 + \alpha^{11}\alpha^2 = \alpha^{12} \\ s_2 &= \alpha\alpha^{14} + \alpha^5\alpha^{10} + \alpha^{11}\alpha^4 = 1 \\ s_3 &= \alpha\alpha^{21} + \alpha^5\alpha^{15} + \alpha^{11}\alpha^6 = \alpha^{14} \\ s_4 &= \alpha\alpha^{28} + \alpha^5\alpha^{20} + \alpha^{11}\alpha^8 = \alpha^{13} \\ s_5 &= \alpha\alpha^{35} + \alpha^5\alpha^{25} + \alpha^{11}\alpha^{10} = 1 \\ s_6 &= \alpha\alpha^{42} + \alpha^5\alpha^{30} + \alpha^{11}\alpha^{12} = \alpha^{11} \end{aligned} \tag{11.236}$$

From (11.202)–(11.205), the algorithm develops according to the following $d - 1 = 6$ steps, starting from the initial conditions

$$\begin{aligned} \lambda^{(0)}(x) &= 1 \\ \beta^{(0)}(x) &= 1 \\ L_0 &= 0 \end{aligned} \tag{11.237}$$

Step 1 ($r = 1$): $\Delta_1 = \alpha^{12}$, and $2L_0 = 0 = r - 1$, $\delta_1 = 1$, $L_1 = 1$.

$$\begin{bmatrix} \lambda^{(1)}(x) \\ \beta^{(1)}(x) \end{bmatrix} = \begin{bmatrix} 1 & -\alpha^{12}x \\ \alpha^3 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{matrix} 1 + \alpha^{12}x \\ \alpha^3 \end{matrix}$$

Step 2 ($r = 2$): $\Delta_2 = 1 + \alpha^9 = \alpha^7$, and $2L_1 = 2 > r - 1$, $\delta_2 = 0$, $L_2 = L_1 = 1$.

$$\begin{aligned} \begin{bmatrix} \lambda^{(2)}(x) \\ \beta^{(2)}(x) \end{bmatrix} &= \begin{bmatrix} 1 & -\alpha^7 x \\ 0 & x \end{bmatrix} \begin{bmatrix} 1 + \alpha^{12}x \\ \alpha^3 \end{bmatrix} \\ &= \begin{bmatrix} 1 + \alpha^{12}x + \alpha^{10}x \\ \alpha^3 x \end{bmatrix} = \begin{bmatrix} 1 + \alpha^3 x \\ \alpha^3 x \end{bmatrix} \end{aligned}$$

Step 3 ($r = 3$): $\Delta_3 = \alpha^{14} + \alpha^3 = 1$, and $2L_2 = 2 = r - 1$, $\delta_3 = 1$, $L_3 = 3 - 1 = 2$.

$$\begin{bmatrix} \lambda^{(3)}(x) \\ \beta^{(3)}(x) \end{bmatrix} = \begin{bmatrix} 1 & -x \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 + \alpha^3 x \\ \alpha^3 x \end{bmatrix} = \begin{bmatrix} 1 + \alpha^3 x + \alpha^3 x^2 \\ 1 + \alpha^3 x \end{bmatrix}$$

Step 4 ($r = 4$): $\Delta_4 = \alpha^{13} + \alpha^3\alpha^{14} + \alpha^3 = 1$, and $2L_3 = 4 > r - 1$, $\delta_4 = 0$, $L_4 = L_3 = 2$.

$$\begin{aligned} \begin{bmatrix} \lambda^{(4)}(x) \\ \beta^{(4)}(x) \end{bmatrix} &= \begin{bmatrix} 1 & -x \\ 0 & x \end{bmatrix} \begin{bmatrix} 1 + \alpha^3 x + \alpha^3 x^2 \\ 1 + \alpha^3 x \end{bmatrix} \\ &= \begin{bmatrix} 1 + \alpha^3 x + \alpha^3 x^2 + x + \alpha^3 x^2 \\ x + \alpha^3 x^2 \end{bmatrix} = \begin{bmatrix} 1 + \alpha^{14} x \\ x + \alpha^3 x^2 \end{bmatrix} \end{aligned}$$

Step 5 ($r = 5$): $\Delta_5 = 1 + \alpha^{14}\alpha^{13} = \alpha^{11}$, and $2L_4 = 4 = r - 1$, $\delta_5 = 1$, $L_5 = 5 - 2 = 3$.

$$\begin{bmatrix} \lambda^{(5)}(x) \\ \beta^{(5)}(x) \end{bmatrix} = \begin{bmatrix} 1 & -\alpha^{11}x \\ \alpha^4 & 0 \end{bmatrix} \begin{bmatrix} 1 + \alpha^{14}x \\ x + \alpha^3 x^2 \end{bmatrix} = \begin{bmatrix} 1 + \alpha^{14}x + \alpha^{11}x^2 + \alpha^{14}x^3 \\ \alpha^4 + \alpha^3 x \end{bmatrix}$$

Step 6 ($r = 6$): $\Delta_6 = \alpha^{11} + \alpha^{14} + \alpha^{11}\alpha^{13} + \alpha^{14}\alpha^{14} = 0$, $\delta_6 = 0$, $L_6 = L_5 = 3$.

$$\begin{bmatrix} \lambda^{(6)}(x) \\ \beta^{(6)}(x) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & x \end{bmatrix} \begin{bmatrix} 1 + \alpha^{14}x + \alpha^{11}x^2 + \alpha^{14}x^3 \\ \alpha^4 + \alpha^3 x \end{bmatrix}$$

$$= \begin{bmatrix} 1 + \alpha^{14}x + \alpha^{11}x^2 + \alpha^{14}x^3 \\ \alpha^4 x + \alpha^3 x^2 \end{bmatrix}$$

The error indicator polynomial is $\lambda(x) = \lambda^{(6)}(x)$. By using the exhaustive method to find the three roots, we obtain

$$\lambda(x) = 1 + \alpha^{14}x + \alpha^{11}x^2 + \alpha^{14}x^3 = (1 - \alpha^7 x)(1 - \alpha^5 x)(1 - \alpha^2 x) \tag{11.238}$$

Consequently the three errors are at positions $\ell = 2, 5,$ and 7.

To determine the values of the errors we use the Forney algorithm. The derivative of $\lambda(x)$ is given by

$$\lambda'(x) = \alpha^{14} + \alpha^{14}x^2 \tag{11.239}$$

The error evaluator polynomial is given by

$$\begin{aligned} \omega(x) &= (\alpha^{12}x + x^2 + \alpha^{14}x^3 + \alpha^{13}x^4 + x^5 + \alpha^{11}x^6) \\ &\quad (1 + \alpha^{14}x + \alpha^{11}x^2 + \alpha^{14}x^3) \bmod x^6 \\ &= \alpha^{12}x + \alpha^{12}x^2 + \alpha^8 x^3 \end{aligned} \tag{11.240}$$

Thus the values of the errors are

$$\begin{aligned} \varepsilon_2 &= -\frac{\omega(\alpha^{-2})}{\alpha^{-2}\lambda'(\alpha^{-2})} = \alpha^{11} \\ \varepsilon_5 &= -\frac{\omega(\alpha^{-5})}{\alpha^{-5}\lambda'(\alpha^{-5})} = \alpha^5 \\ \varepsilon_7 &= -\frac{\omega(\alpha^{-7})}{\alpha^{-7}\lambda'(\alpha^{-7})} = \alpha \end{aligned} \tag{11.241}$$

An alternative approach for the encoding and decoding of Reed–Solomon codes utilizes the concept of Fourier transform on a Galois field [2, 5]. Let α be a primitive element of the field $GF(q)$. The Fourier transform on the field $GF(q)$ (GFFT) of a vector $\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$ of n bits is defined as $(C_0, C_1, \ldots, C_{n-1})$, where

$$C_j = \sum_{i=0}^{n-1} c_i \, \alpha^{ij} \qquad j = 0, \ldots, n-1 \tag{11.242}$$

Let us consider a code word $\mathbf{c}$ of n bits in the "time domain" from a Reed–Solomon cyclic code that corrects up to t errors; then $\mathbf{c}$ corresponds to a code polynomial that has as roots $2t = d-1$ consecutive powers of α. If we take the GFFT of this word, we find that in the "frequency domain" the transform has $2t$ consecutive components equal to zero. Indeed from (11.176), specialized to Reed–Solomon codes, and from (11.242), we can show that the two conditions are equivalent, that is a polynomial has $2t$ consecutive powers of α as roots if and only if the transform has $2t$ consecutive components equal to zero. The approach that resorts to the GFFT is therefore the mirror of the approach that uses the generator polynomial. This observation leads to the development of efficient methods for encoding and decoding.

11.2.6 Performance of block codes

In this section we consider the probability of error in the decoding of block codes, in the case of decoding with *hard* or *soft* input (see Section 6.8). For an in-depth study of the subject we refer the reader, for example, to [6].

With reference to Figure 6.20, let P_{bit} be the bit error probability for the detection of the bits of the binary sequence $\{\tilde{c}_m\}$, or bit error probability of the channel, P_w the error probability for a code word, and $P_{bit}^{(dec)}$ the error probability for a bit of the binary sequence $\{\hat{b}_l\}$ obtained after decoding. For a (n, k) block code with $t = (d_{min}^H - 1)/2$ and hard input decoding the following inequality holds:

$$P_w \leq \sum_{i=t+1}^{n} \binom{n}{i} P_{bit}^i (1 - P_{bit})^{n-i} \tag{11.243}$$

which, under the condition $n P_{bit} \ll 1$, can be approximated as

$$P_w \simeq \binom{n}{t+1} P_{bit}^{t+1} (1 - P_{bit})^{n-t-1} \tag{11.244}$$

The inequality (11.243) follows from the channel model (11.12) assuming errors that are i.i.d., and from the consideration that the code may not be perfect (see page 839), and therefore it could correct also some received words with more than t errors.

If a word error occurs, the most probable event is that the decoder decides for a code word with distance $d_{min}^H = 2t + 1$ from the transmitted code word, thus making d_{min}^H bit errors in the sequence $\{\hat{c}_m\}$. As $\mathbf{c}$ is formed of n bits, we have that at the decoder output the bit error probability is

$$P_{bit}^{(dec)} \simeq \frac{2t+1}{n} P_w \tag{11.245}$$

Example 11.2.34
For a (5,1) repetition code with $d_{min}^H = 5$ (see page 839), decoding with hard input yields

$$P_w = \binom{5}{3} P_{bit}^3 (1 - P_{bit})^2 + \binom{5}{4} P_{bit}^4 (1 - P_{bit}) + P_{bit}^5 \tag{11.246}$$

Example 11.2.35
For an (n, k) Hamming code with $d_{min}^H = 3$ (see page 839), (11.243) yields

$$\begin{aligned} P_w &\leq \sum_{i=2}^{n} \binom{n}{i} P_{bit}^i (1 - P_{bit})^{n-i} \\ &= 1 - [(1 - P_{bit})^n + n P_{bit}(1 - P_{bit})^{n-1}] \end{aligned} \tag{11.247}$$

For example, for a (15,11) code, if $P_{bit} = 10^{-3}$ then $P_w \simeq 10^{-4}$, and from (11.245) we get $P_{bit}^{(dec)} \simeq 2\ 10^{-5}$.

The decoders that have been considered so far are classified as hard input decoders, as the demodulator output is quantized to the values of the coded symbols before decoding. In general, other decoding algorithms with soft input may be considered, that directly process the demodulated signal, and consequently the decoder input is real valued (see Section 11.3.2).

In the case of antipodal binary signals and soft input decoding we obtain (see also Section 6.8 on page 496)

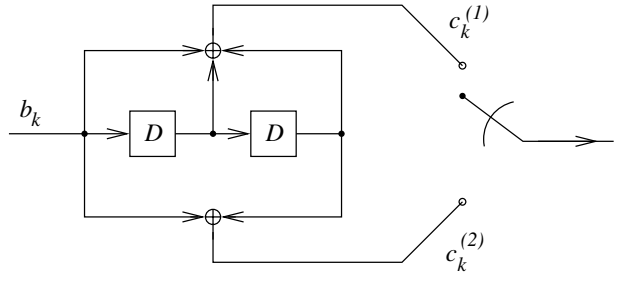
$$P_w \simeq (2^k - 1)\, Q\left(\sqrt{\frac{R_c\, d_{min}^H\, 2E_b}{N_0}}\right) \tag{11.248}$$

11.3 Convolutional codes

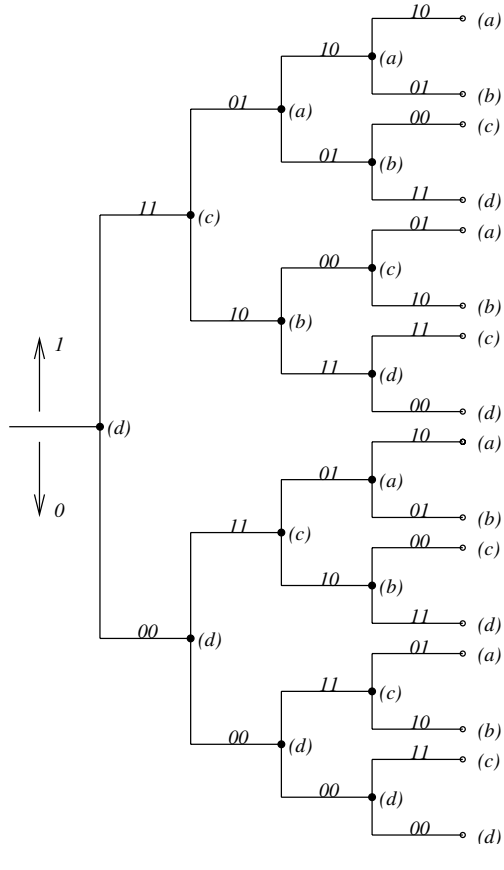
Convolutional codes are a subclass of the class of *tree codes*, so called because their code words are conveniently represented as sequences of nodes in a tree. Tree codes are of great interest because decoding algorithms have been found that are easy to implement, and can be applied to the entire class of tree codes, in contrast to decoding algorithms for block codes, each designed for a specific class of codes, as for example BCH codes.

Several approaches have been used in the literature for describing convolutional codes; here we will illustrate these approaches by first considering a specific example.

Example 11.3.1
Consider a rate $1/2$ binary convolutional code, obtained by the encoder illustrated in Figure 11.9a. For each bit b_k that enters the encoder, two output bits, $c_k^{(1)}$ and $c_k^{(2)}$, are transmitted. The first output $c_k^{(1)}$ is obtained if the switch at the output is in the upper position, and the second output $c_k^{(2)}$ is obtained if the switch is in the lower position; the two previous input bits, b_{k-1} and b_{k-2}, are stored in the memory of the encoder. As the information bit is not presented directly to one of the outputs, we say that the code is nonsystematic. The two coded bits are generated as linear combinations of the bits of the message; denoting the input sequence as $\ldots, b_0, b_1, b_2, b_3, \ldots$, and the output sequence as $\ldots, c_0^{(1)}, c_0^{(2)}, c_1^{(1)}, c_1^{(2)}, c_2^{(1)}, c_2^{(2)}, c_3^{(1)}, c_3^{(2)}, \ldots$, then the following



(a)



(b)

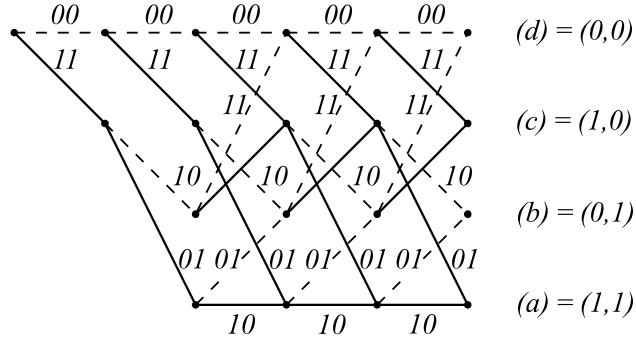
Figure 11.9. (a) Encoder and (b) tree diagram for the convolutional code of Example 11.3.1.

relations hold:

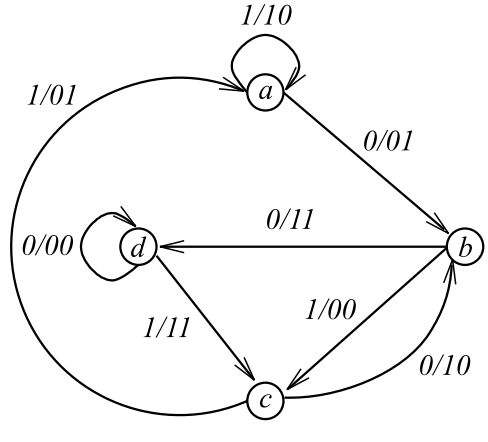
$$\begin{aligned} c_k^{(1)} &= b_k \oplus b_{k-1} \oplus b_{k-2} \\ c_k^{(2)} &= b_k \oplus b_{k-2} \end{aligned} \tag{11.249}$$

A convolutional code may be described in terms of a tree, trellis, or state diagram; for the code defined by (11.249) these descriptions are illustrated in Figures 11.9b, 11.10a, and 11.10b, respectively.

With reference to the tree diagram of Figure 11.9b, we begin at the left (root) node and proceed to the right by choosing an upper path if the input bit is equal to 1 and a lower path if the input bit is 0. We output the two bits represented by the label on the branch that takes us to the next node, and then repeat this process at the next node. The nodes or



(a)



(b)

Figure 11.10. (a) Trellis diagram and (b) state diagram for the convolutional code of Example 11.3.1.

states of the encoder are labeled with the letters a, b, c, and d, which indicate the relation with the four possible values assumed by the two bits stored in the encoder, according to the table:

b_{k-1}	b_{k-2}	*label*
0	0	d
1	0	c
0	1	b
1	1	a

(11.250)

If for example we input the sequence $b_0, b_1, b_2, b_3, \cdots = 1\ 1\ 0\ 1\ \ldots$, we would then output the sequence $c_0^{(1)}, c_0^{(2)}, c_1^{(1)}, c_1^{(2)}, c_2^{(1)}, c_2^{(2)}, c_3^{(1)}, c_3^{(2)}, \cdots = 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ \ldots$. As, at any depth in the tree, nodes with the same label will have the same tree growing from them, we can superimpose these nodes on a single node. This results in the trellis diagram represented in Figure 11.10a, where solid and dashed lines correspond to transitions determined by input bits equal to 1 and 0, respectively.

The state diagram for the encoder is illustrated in Figure 11.10b. The four states (a, b, c, d) correspond to the four possible combinations of bits stored in the encoder. If the encoder is in a certain state, a transition to one of two possible states occurs, depending on the value of the input bit. Possible transitions between states are represented as arcs, on which an arrow indicates the direction of the transition; with each arc is associated a label that indicates the value assumed by the input bit, and also the value of the resulting output bits. The description of the encoder by the state diagram is convenient for analyzing the properties of the code, as we will see later.

It is also convenient to represent code sequences in terms of the D transform, as

$$\begin{aligned} b(D) &= b_0 + b_1 D + b_2 D^2 + b_3 D^3 + \cdots \\ c^{(1)}(D) &= c_0^{(1)} + c_1^{(1)} D + c_2^{(1)} D^2 + c_3^{(1)} D^3 + \cdots = g^{(1,1)}(D)\, b(D) \\ c^{(2)}(D) &= c_0^{(2)} + c_1^{(2)} D + c_2^{(2)} D^2 + c_3^{(2)} D^3 + \cdots = g^{(2,1)}(D)\, b(D) \end{aligned} \tag{11.251}$$

where $g^{(1,1)}(D) = 1 + D + D^2$, and $g^{(2,1)}(D) = 1 + D^2$.

11.3.1 General description of convolutional codes

In general we consider convolutional codes with symbols from $GF(q)$; assuming the encoder produces n_0 output code symbols for every k_0 input message symbols, the code rate is equal to k_0/n_0. It is convenient to think of the message sequence as being the interlaced version of k_0 different message sequences, and to think of the code sequence as the interlaced version of n_0 different code sequences. In other words, given the information sequence $\{b_\ell\}$ we form the k_0 subsequences

$$b_k^{(i)} = b_{kk_0+i-1} \qquad i = 1, \ldots, k_0 \tag{11.252}$$

that have D transform defined as

$$\begin{aligned} b_0^{(1)} b_1^{(1)} b_2^{(1)} \cdots &\Longleftrightarrow b^{(1)}(D) = b_0^{(1)} + b_1^{(1)} D + b_2^{(1)} D^2 + \cdots \\ b_0^{(2)} b_1^{(2)} b_2^{(2)} \cdots &\Longleftrightarrow b^{(2)}(D) = b_0^{(2)} + b_1^{(2)} D + b_2^{(2)} D^2 + \cdots \\ &\vdots \qquad\qquad\qquad\qquad \vdots \\ b_0^{(k_0)} b_1^{(k_0)} b_2^{(k_0)} \cdots &\Longleftrightarrow b^{(k_0)}(D) = b_0^{(k_0)} + b_1^{(k_0)} D + b_2^{(k_0)} D^2 + \cdots \end{aligned} \tag{11.253}$$

Let $c^{(1)}(D), c^{(2)}(D), \ldots, c^{(n_0)}(D)$ be the D transforms of the n_0 output sequences; then

$$c^{(j)}(D) = \sum_{i=1}^{k_0} g^{(j,i)}(D)\, b^{(i)}(D) \qquad j = 1, 2, \ldots, n_0 \tag{11.254}$$

An (n_0, k_0) convolutional code is then specified by giving the coefficients of all the polynomials $g^{(j,i)}(D)$, $i = 1, 2, \ldots, k_0$, $j = 1, 2, \ldots, n_0$.

If for all $j = 1, 2, \ldots, k_0$, we have

$$g^{(j,i)}(D) = \begin{cases} 1 & j = i \\ 0 & j \neq i \end{cases} \tag{11.255}$$

then the code is systematic and k_0 of the n_0 output sequences are just the message sequences.

An encoder for a convolutional code needs storage elements. Let ν be the *constraint length* of the code,[5]

$$\nu = \max_{j,i} (\deg g^{(j,i)}(D)) \tag{11.256}$$

Therefore the encoder of a convolutional code must store ν previous blocks of k_0 message symbols to form a block of n_0 output symbols.

The general structure of an encoder for a code with $k_0 = 1$ and $n_0 = 2$ is illustrated in Figure 11.11; for such an encoder, νk_0 storage elements are necessary. If the code is systematic, then the encoder can be implemented with $\nu(n_0 - k_0)$ storage elements, as illustrated in Figure 11.12 for $k_0 = 2$ and $n_0 = 3$.

If we interpret the sequence $\{c_k\}$ as the output of a sequential finite-state machine (see Appendix 8.D), at instant k the trellis of a nonsystematic code is defined by the three signals:

1. Input $$[b_k^{(1)}, b_k^{(2)}, \ldots, b_k^{(k_0)}] \tag{11.257}$$
2. State $$[b_k^{(1)}, \ldots, b_k^{(k_0)}, \ldots, b_{k-(\nu-1)}^{(1)}, \ldots, b_{k-(\nu-1)}^{(k_0)}] \tag{11.258}$$
3. Output $$[c_k^{(1)}, c_k^{(2)}, \ldots, c_k^{(n_0)}] \tag{11.259}$$

where $c_k^{(j)}$, $j = 1, \ldots, n_0$, is given by (11.254). Then there are $q^{k_0 \nu}$ states in the trellis. There are q^{k_0} branches departing from each state and q^{k_0} branches merging into a state. The output vector consists of n_0 q–ary symbols.

[5] Many authors define the constraint length as $\nu + 1$, where ν is given by (11.256).

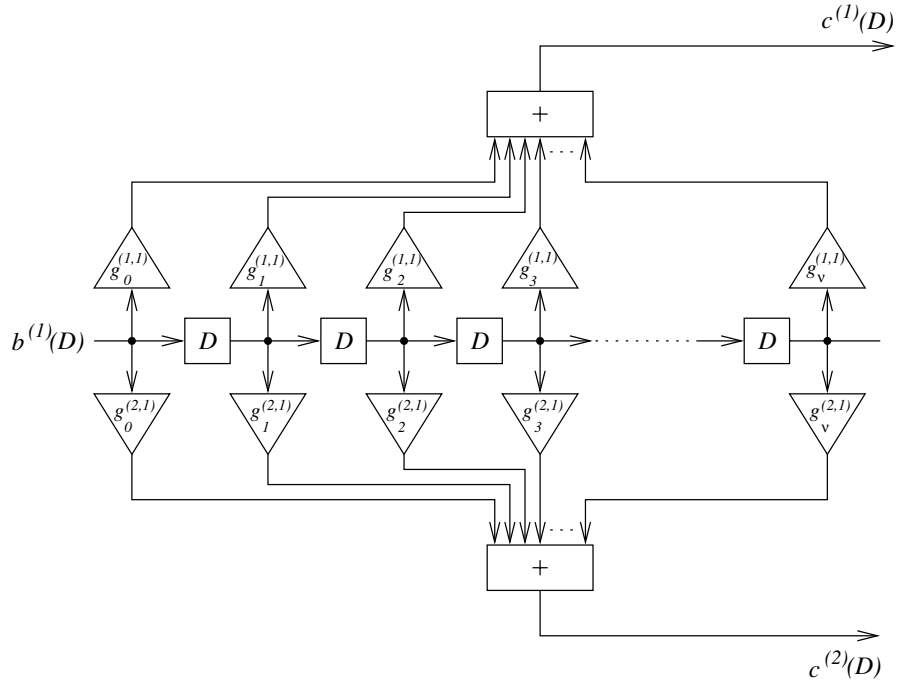


Figure 11.11. Block diagram of an encoder for a convolutional code with $k_0 = 1$, $n_0 = 2$, and constraint length ν.

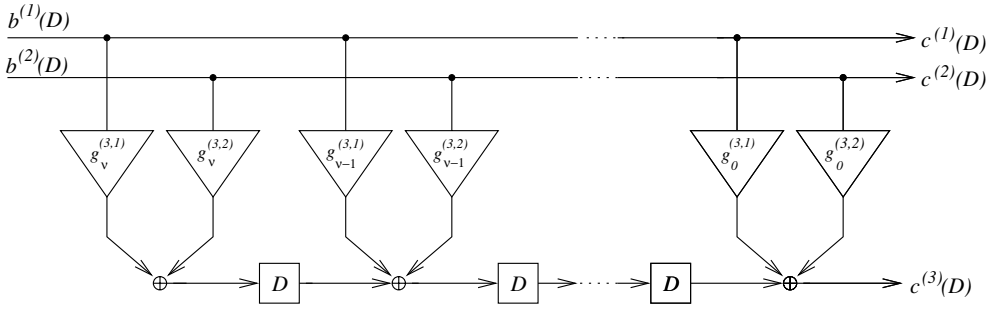


Figure 11.12. Block diagram of an encoder for a systematic convolutional code with $k_0 = 2$, $n_0 = 3$, and constraint length ν.

Parity check matrix

A semi-infinite parity check matrix can be defined in general for convolutional codes; however, we note that it is only in the case of systematic codes that we can easily express the elements of this matrix in terms of the coefficients of the generator polynomials $g^{(j,i)}(D)$.

We write the coefficients of the generator polynomials in the form

$$\begin{matrix} g_0^{(1,1)} g_0^{(2,1)} \cdots g_0^{(n_0,1)} & g_1^{(1,1)} \cdots g_1^{(n_0,1)} & \cdots & g_\nu^{(1,1)} g_\nu^{(2,1)} \cdots g_\nu^{(n_0,1)} \\ g_0^{(1,2)} g_0^{(2,2)} \cdots g_0^{(n_0,2)} & g_1^{(1,2)} \cdots g_1^{(n_0,2)} & \cdots & g_\nu^{(1,2)} g_\nu^{(2,2)} \cdots g_\nu^{(n_0,2)} \\ \vdots & & & \vdots \\ g_0^{(1,k_0)} g_0^{(2,k_0)} \cdots g_0^{(n_0,k_0)} & g_1^{(1,k_0)} \cdots g_1^{(n_0,k_0)} & \cdots & g_\nu^{(1,k_0)} g_\nu^{(2,k_0)} \cdots g_\nu^{(n_0,k_0)} \end{matrix} \tag{11.260}$$

If the code is systematic, the parity matrix of the generator polynomials can be written as

$$\mathbf{I}\ \mathbf{P}_0\ \mathbf{0}\ \mathbf{P}_1\ \ldots\ \mathbf{0}\ \mathbf{P}_\nu \tag{11.261}$$

where $\mathbf{I}$ and $\mathbf{0}$ are $k_0 \times k_0$ matrices and $\mathbf{P}_i$, $i = 0, \ldots, \nu$, are $k_0 \times (n_0 - k_0)$ matrices. The semi-infinite parity check matrix is then

$$\mathbf{H}_\infty = \begin{bmatrix} -\mathbf{P}_0^T & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \ldots \\ -\mathbf{P}_1^T & \mathbf{0} & -\mathbf{P}_0^T & \mathbf{I} & \mathbf{0} & \mathbf{0} & \ldots \\ \ldots & & \ldots & & \ldots & & \ldots \\ -\mathbf{P}_\nu^T & \mathbf{0} & -\mathbf{P}_{\nu-1}^T & \mathbf{0} & -\mathbf{P}_{\nu-2}^T & \mathbf{0} & \ldots \\ \mathbf{0} & \mathbf{0} & -\mathbf{P}_\nu^T & \mathbf{0} & -\mathbf{P}_{\nu-1}^T & \mathbf{0} & \ldots \\ \ldots & & \ldots & & \ldots & & \ldots \end{bmatrix} \tag{11.262}$$

Thus for any code word $\mathbf{c}$ of infinite length, $\mathbf{H}_\infty \mathbf{c} = \mathbf{0}$. Often, rather than considering the semi-infinite matrix $\mathbf{H}_\infty$, we consider the finite matrix $\mathbf{H}$ defined as

$$\mathbf{H} = \begin{bmatrix} -\mathbf{P}_0^T & \mathbf{I} & \mathbf{0} & \mathbf{0} & \ldots & \mathbf{0} & \mathbf{0} \\ -\mathbf{P}_1^T & \mathbf{0} & -\mathbf{P}_0^T & \mathbf{I} & \ldots & \mathbf{0} & \mathbf{0} \\ \ldots & & \ldots & & \ldots & \ldots & \\ -\mathbf{P}_\nu^T & \mathbf{0} & -\mathbf{P}_{\nu-1}^T & \mathbf{0} & \ldots & -\mathbf{P}_0^T & \mathbf{I} \end{bmatrix} \tag{11.263}$$

The bottom row of matrices of the matrix $\mathbf{H}$ is called the *basic parity check matrix*. From it we can see that the parity symbols in a block are given by the linear combination of information bits in that block, corresponding to non-zero terms in $-\mathbf{P}_0^T$, in the immediately preceding block, corresponding to non-zero terms in $-\mathbf{P}_1^T$, and so on until the ν-th preceding block, corresponding to non-zero terms in $-\mathbf{P}_\nu^T$.

Generator matrix

From (11.260), we introduce the matrices

$$\mathbf{g}_i = \begin{bmatrix} g_i^{(1,1)} & g_i^{(2,1)} & \cdots & g_i^{(n_0,1)} \\ \vdots & & & \vdots \\ g_i^{(1,k_0)} & g_i^{(2,k_0)} & \cdots & g_i^{(n_0,k_0)} \end{bmatrix} \qquad i = 0, \ldots, \nu \tag{11.264}$$

Hence the generator matrix is of the form

$$\mathbf{G}_\infty = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \cdots & \mathbf{g}_\nu & \mathbf{0} & \ldots \\ \mathbf{0} & \mathbf{g}_0 & \ldots & \mathbf{g}_{\nu-1} & \mathbf{g}_\nu & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \end{bmatrix} \tag{11.265}$$

Some examples of convolutional codes with the corresponding encoders and generator matrices are illustrated in Figure 11.13.

Transfer function

An important parameter of a convolutional code is d^H_{free}, that determines the performance of the code (see Section 11.3.3).

Definition 11.14
Let $\mathbf{e}(D) = [e^{(n_0)}(D), \ldots, e^{(1)}(D)]$, be any error sequence between two code words $\mathbf{c}_1(D) = [c_1^{(n_0)}(D), \ldots, c_1^{(1)}(D)]$ and $\mathbf{c}_2(D) = [c_2^{(n_0)}(D), \ldots, c_2^{(1)}(D)]$, that is $\mathbf{c}_1(D) = \mathbf{c}_2(D) + \mathbf{e}(D)$, and $\mathbf{e}_k = [e_k^{(n_0)}, \ldots, e_k^{(1)}]$ denotes the k-th element of the sequence. We define the free Hamming distance of the code as

$$d^H_{free} = \min_{\mathbf{e}(D)} \sum_{k=0}^{\infty} w(\mathbf{e}_k) \tag{11.266}$$

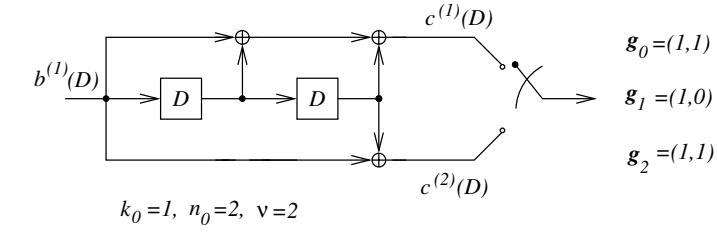
where w is introduced in Definition 11.4 on page 832. As the code is linear, d^H_{free} corresponds to the minimum number of symbols different from zero in a non-zero code word.

Next we consider a method to compute the weights of all code words in a convolutional code; to illustrate the method we examine the simple binary encoder of Figure 11.9a.

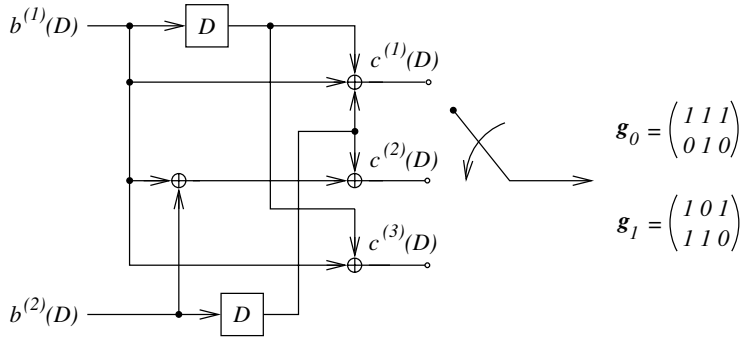
We begin by reproducing the trellis diagram of the code in Figure 11.14, where each path is now labeled with the weight of the output bits corresponding to that path. We consider all paths that diverge from state (d) and return to state (d) for the first time after a number of steps j. By inspection, we find one such path of weight 5 returns to state (d) after 3 steps; moreover, we find two distinct paths of weight 6, one that returns to state (d) after 4 steps and another after 5 steps. Hence we find that this code has $d^H_{free} = 5$.

We now look for a method that enables us to find the weights of all code words as well as the lengths of the paths that give origin to the code words with these weights. Consider the state diagram for this code, redrawn in Figure 11.15 with branches labeled as D^2, D, or $D^0 = 1$, where the exponent corresponds to the weight of the output bits corresponding to that branch. Next we split node (0,0) to obtain the state diagram of Figure 11.16, and we compute a *generating function* for the weights. The generating function is the transfer function of a signal flow graph with unit input. From Figure 11.16, we obtain this transfer function by solving the system of equations

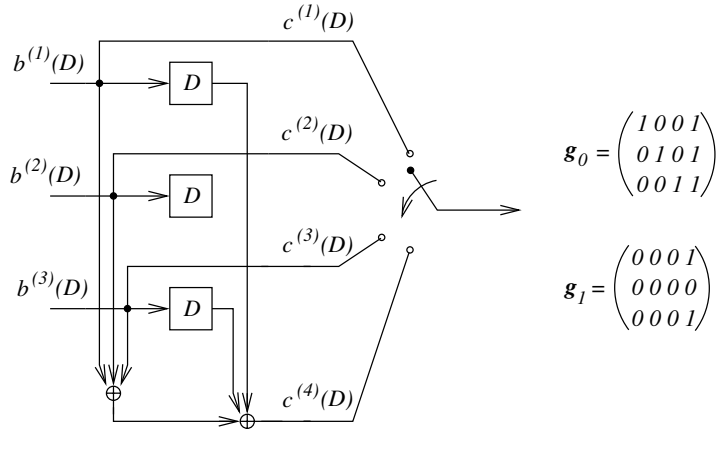
$$\begin{aligned} \beta &= D^2\alpha + 1\gamma \\ \gamma &= D\beta + D\delta \\ \delta &= D\beta + D\delta \\ \eta &= D^2\gamma \end{aligned} \tag{11.267}$$



(a)



(b)



(c)

Figure 11.13. Examples of encoders for three convolutional codes.

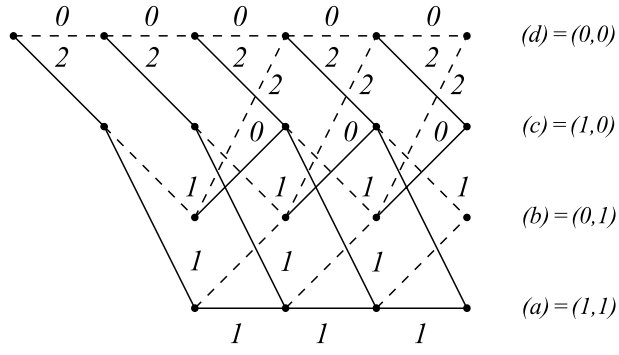


Figure 11.14. Trellis diagram of the code of Example 11.3.1; the labels represent the Hamming weight of the output bits.

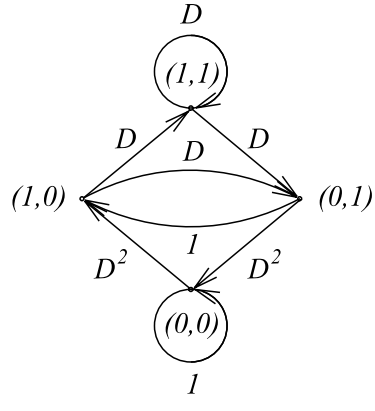


Figure 11.15. State diagram of the code of Example 11.3.1; the labels represent the Hamming weight of the generated bits.

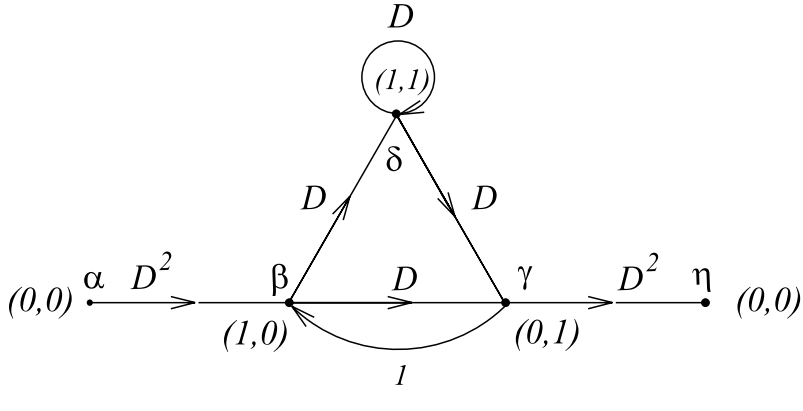


Figure 11.16. State diagram of the code of Example 11.3.1; node (0,0) is split to compute the transfer function of the code.

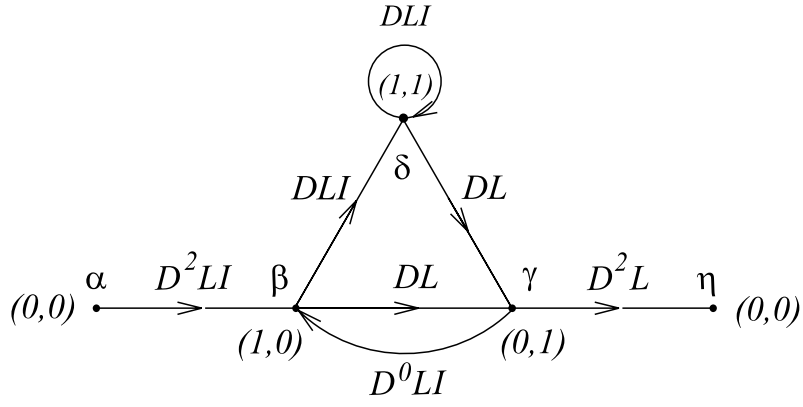


Figure 11.17. State diagram of the code of Example 11.3.1; node (0,0) is split to compute the augmented transfer function.

Then we get

$$t(D) = \frac{\eta}{\alpha} = \frac{D^5}{1-2D} = D^5 + 2D^6 + 4D^7 + \cdots + 2^i D^{i+5} + \cdots \tag{11.268}$$

From inspection of $t(D)$, we find there is one code word of weight 5, two of weight 6, four of weight 7, ... Equation (11.268) holds for code words of infinite length.

If we want to find code words that return to state (d) after j steps we refer to the state diagram of Figure 11.17. The term L introduced in the label on each branch allows to keep track of the length of the sequence, as the power of L is augmented by 1 every time a transition occurs. Furthermore, we introduce the term I in the label on a branch if the corresponding transition is due to an information bit equal to 1; this allows computation for each path on the trellis diagram of the corresponding number of information bits equal to 1. The augmented transfer function is given by

$$\begin{aligned} t(D, L, I) &= \frac{D^5 L^3 I}{1 - DL(1+L)I} \\ &= D^5 L^3 I + D^6 L^4 (1+L) I^2 + D^7 L^5 (1+L)^2 I^3 + \cdots \\ &\quad + D^{5+i} L^{3+i} (1+L)^i I^{1+i} + \cdots \end{aligned} \tag{11.269}$$

Thus we see that the code word of weight 5 is of length 3 and is originated by a sequence of information bits that contains one bit equal to 1, there are two code words of weight 6, one of length 4 and the other of length 5, both of which are originated by a sequence of information bits that contain two bits equal to 1, ...

Catastrophic error propagation

For certain codes a finite number of channel errors may lead to an infinite number of errors in the sequence of decoded bits. For example, consider the code with encoder and state diagram illustrated in Figure 11.18a and b, respectively. Note that in the state diagram the self-loop at

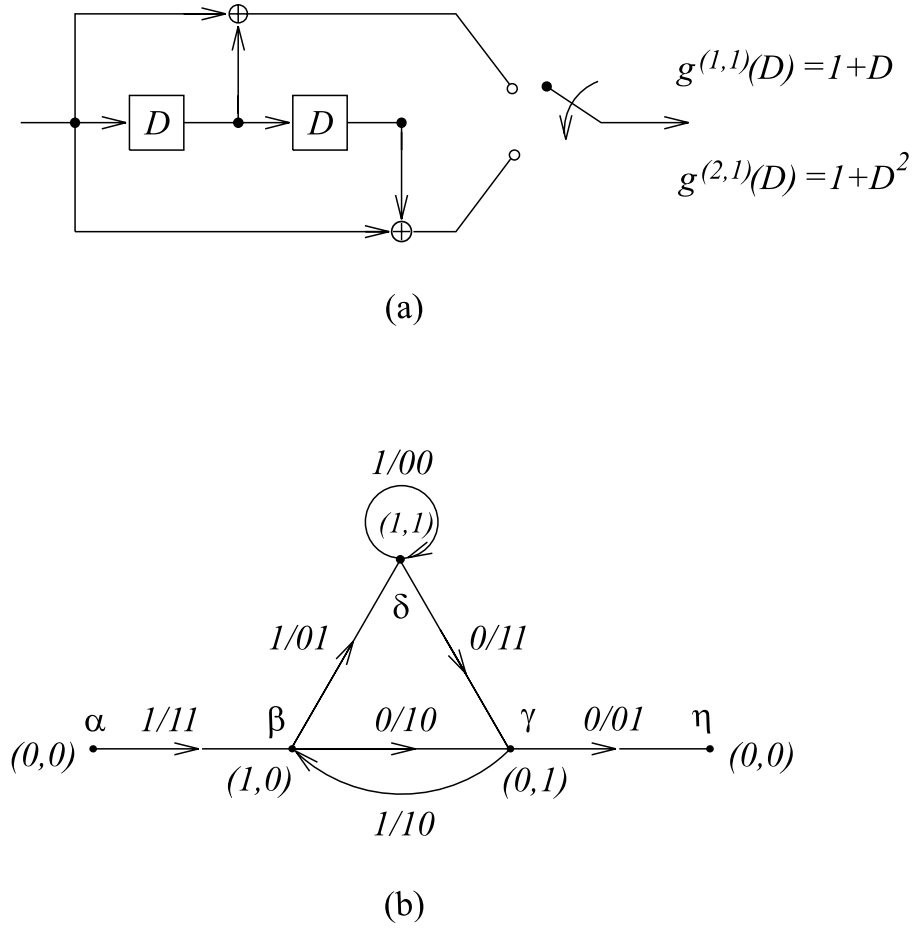


Figure 11.18. (a) Encoder and (b) state diagram for a catastrophic convolutional code.

state (1, 1) does not increase the weight of the code word, so that a code word corresponding to a path passing through the states (0, 0), (1, 0), (1, 1), (1, 1), . . . , (1, 1), (0, 1), (0, 0) is of weight 6, independently of the number of times it passes through the self loop at state (1,1). In other words, long sequences of coded bits equal to zero may be obtained by remaining in the state (0, 0) with a sequence of information bits equal to zero, or by remaining in the state (1, 1) with a sequence of information bits equal to one. Therefore a limited number of channel errors, in this case 6, can cause a large number of errors in the sequence of decoded bits.

Definition 11.15
A convolutional code is *catastrophic* if there exists a closed loop in the state diagram that has all branches with zero weight.

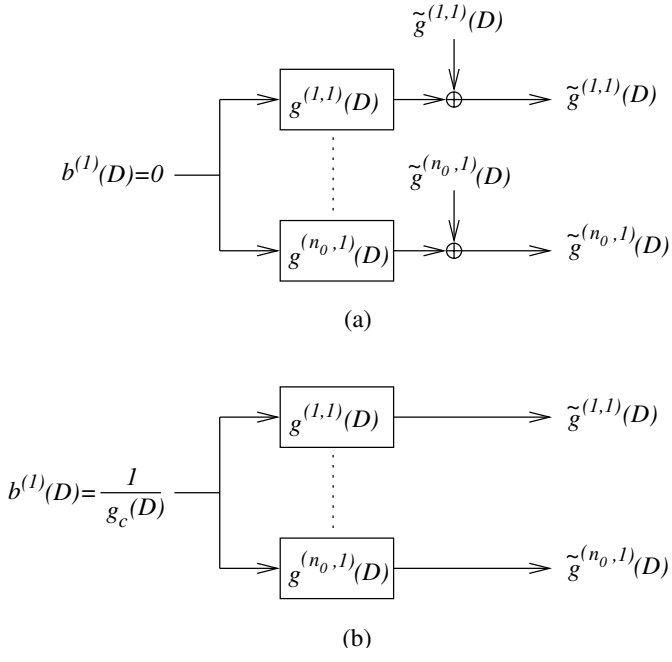


Figure 11.19. Two distinct infinite sequences of information bits that produce the same output sequence with a finite number of errors.

For codes with rate $1/n_0$, it has been shown that a code is catastrophic if and only if all generator polynomials have a common polynomial factor. In the above example, the common factor is $1 + D$. This can be proved using the following argument: suppose that $g^{(1,1)}(D), g^{(2,1)}(D), \ldots, g^{(k_0,1)}(D)$ all have the common factor $g_c(D)$, so that

$$g^{(i,1)}(D) = g_c(D)\, \tilde{g}^{(i,1)}(D) \tag{11.270}$$

Suppose the all zero sequence is sent, $b^{(1)}(D) = 0$, and that the finite error sequence $\tilde{g}^{(i,1)}(D)$, equal to that defined in (11.270), occurs in the i-th subsequence output, for $i = 1, 2, \ldots, n_0$, as illustrated in Figure 11.19a. The same output sequence is obtained if the sequence of information bits with infinite length $b^{(1)}(D) = 1/g_c(D)$ is sent, and no channel errors occur, as illustrated in Figure 11.19b. Thus a finite number of errors yields a decoded sequence of information bits that differ from the transmitted sequence in an infinite number of positions.

11.3.2 Decoding of convolutional codes

Various algorithms have been developed for the decoding of convolutional codes. One of the first decoding methods was algebraic decoding, which is similar to the methods developed for the decoding of block codes. However, this method has the disadvantages that it is applicable only to a limited number of codes having particular characteristics, and exhibits

performance that is lower as compared to decoding methods based on the observation of the whole received sequence. The latter methods, also called *probabilistic decoding methods*, include the Viterbi algorithm (VA), the sequential decoding algorithm by Fano [6], and the forward-backward algorithm by Bahl–Cocke–Jelinek–Raviv (BCJR).

Before illustrating the various decoding methods, we consider an important function.

Interleaving

The majority of block codes as well as convolutional codes are designed by assuming that the errors introduced by the noisy channel are statistically independent. This assumption is not always true in practice. To make the channel errors, at least approximately, statistically independent it is customary to resort to an interleaver, which performs a permutation of the bits of a sequence. For example, a block interleaver orders the coded bits in a matrix with M_1 rows and M_2 columns. The coded bits are usually written in the matrix by row and then read by column before being forwarded to the bit mapper. At the receiver, a deinterleaver stores the detected bits in a matrix of the same $M_1 \times M_2$ dimensions, where the writing is done by column and the reading by row. As a result, possible error bursts of length $M_1 B$ are broken up into bursts of shorter length B.

Two decoding models

We consider a binary convolutional code with $k_0 = 1$, $n_0 = 2$, and constraint length ν. In general, from (11.254) we write the code sequence as a function of the message sequence as

$$\begin{aligned} c_k^{(1)} &= g^{(1,1)}(b_k, \ldots, b_{k-\nu}) \\ c_k^{(2)} &= g^{(2,1)}(b_k, \ldots, b_{k-\nu}) \end{aligned} \tag{11.271}$$

At the receiver, two models may be adopted.

Model with hard input. With reference to the transmission system of Figure 6.20, we consider the sequence at the output of the binary channel. In this case the demodulator has already detected the transmitted symbols, for example, by a threshold detector, and the inverse bit mapper provides the binary sequence $\{z_m = \tilde{c}_m\}$ to the decoder, from which we obtain the interlaced binary sequences

$$\begin{aligned} z_k^{(1)} &= z_{2k} = c_k^{(1)} \oplus e_k^{(1)} \\ z_k^{(2)} &= z_{2k+1} = c_k^{(2)} \oplus e_k^{(2)} \end{aligned} \tag{11.272}$$

where the errors $e_k^{(i)} \in \{0, 1\}$, for a memoryless binary symmetric channel, are i.i.d. (see (6.91)).

From the description on page 904, introducing the state of the encoder at instant k as the vector with ν elements

$$\mathbf{s}_k = [b_k, \ldots, b_{k-(\nu-1)}] \tag{11.273}$$

the desired sequence in (11.272), that coincides with the encoder output, can be written as (see (11.271) and (11.273))

$$\begin{aligned} c_k^{(1)} &= f^{(1)}(\mathbf{s}_k, \mathbf{s}_{k-1}) \\ c_k^{(2)} &= f^{(2)}(\mathbf{s}_k, \mathbf{s}_{k-1}) \end{aligned} \tag{11.274}$$

Model with soft input. Again with reference to Figure 6.20, at the decision point of the receiver the signal can be written as (see (8.173))

$$z_k = u_k + w_k \tag{11.275}$$

where we assume w_k is white Gaussian noise with variance $\sigma_w^2 = 2\sigma_I^2$, and u_k is given by (8.174)

$$u_k = \sum_{n=-L_1}^{L_2} \eta_n \, a_{k-n} \tag{11.276}$$

where $\{a_k\}$ is the sequence of symbols at the output of the bit mapper. Note that in (11.276) the symbols $\{a_k\}$ are in general not independent, as the input of the bit mapper is a code sequence according to the law (11.274).

The relation between u_k and the bits $\{b_\ell\}$ depends on the intersymbol interference in (11.276), the type of bit mapper and the encoder (11.271). We consider the case of absence of ISI, that is

$$u_k = a_k \tag{11.277}$$

and a 16-PAM system where, without interleaving, four consecutive code bits $c_{2k}^{(1)}$, $c_{2k}^{(2)}$, $c_{2k-1}^{(1)}$, $c_{2k-1}^{(2)}$ are mapped into a symbol of the constellation. For an encoder with constraint length ν we have

$$\begin{aligned} u_k = \tilde{f}(a_k) &= \tilde{f}[BMAP\{[c_{2k}^{(1)}, c_{2k}^{(2)}, c_{2k-1}^{(1)}, c_{2k-1}^{(2)}]\}] \\ &= \tilde{f}[BMAP\{g^{(1,1)}(b_{2k}, \ldots, b_{2k-\nu}), g^{(2,1)}(b_{2k}, \ldots, b_{2k-\nu}) \\ &\quad \times g^{(1,1)}(b_{2k-1}, \ldots, b_{2k-1-\nu}), g^{(2,1)}(b_{2k-1}, \ldots, b_{2k-1-\nu})\}] \end{aligned} \tag{11.278}$$

In other words, let

$$\mathbf{s}_k = [b_{2k}, \ldots, b_{2k-\nu+1}] \tag{11.279}$$

we can write

$$u_k = f(\mathbf{s}_k, \mathbf{s}_{k-1}) \tag{11.280}$$

We observe that in this example each state of the trellis admits four possible transitions. As we will see in Chapter 12, better performance is obtained by jointly optimizing the encoder and the bit mapper.

Viterbi algorithm

The Viterbi algorithm, described in Section 8.10.1, is a probabilistic decoding method that implements the maximum likelihood criterion, which minimizes the probability of detecting a sequence that is different from the transmitted sequence.

VA with hard input. The trellis diagram is obtained by using the definition (11.273), and the branch metric is the Hamming distance between $\mathbf{z}_k = [z_k^{(1)}, z_k^{(2)}]^T$ and $\mathbf{c}_k = [c_k^{(1)}, c_k^{(2)}]^T$, (see Definition 11.1),

$$d^H(\mathbf{z}_k, \mathbf{c}_k) = \text{number of positions where } \mathbf{z}_k \text{ differs from } \mathbf{c}_k \tag{11.281}$$

where $\mathbf{c}_k$ is generated according to the rule (11.274).

VA with soft input. The trellis diagram is now obtained by using the definition (11.279), and the branch metric is the Euclidean distance between z_k and u_k,

$$|z_k - u_k|^2 \tag{11.282}$$

where u_k, in the case of the previous example of absence of ISI and 16-PAM transmission, is given by (11.280).

As an alternative to the VA we can use the FBA of Section 8.10.2.

Forward-backward algorithm

The previous approach, which considers joint detection in the presence of ISI and convolutional decoding, requires a computational complexity that in many applications may turn out to be exceedingly large. In fact, the state (11.279), that takes into account both encoding and the presence of ISI, usually is difficult to define and is composed of several bits of the sequence $\{b_\ell\}$. An approximate solution is obtained by considering the detection and the decoding problems separately, however, assuming that the detector passes the soft information on the detected bits to the decoder.

Soft output detection by FBA. By using a trellis diagram that takes into account the ISI introduced by the channel, the code bits $\{c_n\}$ are detected assuming that they are i.i.d., and the reliability of the detection is computed (*soft detection*). For this purpose we use the FBA of page 670, that determines for each state a metric $V_k(i)$, $i = 1, \ldots, N_s$.

Now, with reference to the example of the channel given by (11.277) and 16-PAM transmission, the state is identified by $\mathbf{s}_k = (a_k) = [c_{4k}, c_{4k-1}, c_{4k-2}, c_{4k-3}]$, where[6] $\{c_n\}$, $c_n \in \{-1, 1\}$, is assumed to be a sequence of i.i.d. symbols. By considering the binary state representation, and by suitably adding the values $V_k(i)$, we get the MAP metric, or

[6] It is sometimes convenient to view the encoder output c_n and/or the encoder input b_n as symbols from the alphabet $\{-1, +1\}$, rather than $\{0, 1\}$. It will be clear from the context to which alphabet we refer.

likelihood, associated with the bits $\{c_m\}$,

$$\mathtt{L}_{4k-t}^{(in)}(\alpha) = \sum_{\substack{i=1 \\ \boldsymbol{\sigma}_i \text{ with } t\text{-th binary} \\ \text{component equal to } \alpha}}^{N_s} V_k(i) \qquad \alpha \in \{-1, 1\} \qquad t = 0, 1, 2, 3 \tag{11.283}$$

or equivalently the Log-MAP metric, or log-likelihood,

$$\ell_n^{(in)}(\alpha) = \ln \mathtt{L}_n^{(in)}(\alpha) \qquad \alpha \in \{-1, 1\} \tag{11.284}$$

By the above formulation, the soft decision associated with the bit c_n is given by

$$\ell_n^{(in)} = \ell_n^{(in)}(1) - \ell_n^{(in)}(-1) \tag{11.285}$$

also called log-likelihood ratio (LLR).

Observation 11.3
For binary transmission in the absence of ISI, from (8.269) on page 675, we have, apart from a non-essential additive constant,

$$\ell_n^{(in)}(\alpha) = -\frac{(z_n - \alpha)^2}{2\sigma_I^2} \qquad \alpha \in \{-1, 1\} \tag{11.286}$$

where σ_I^2 is the variance of real-valued noise samples. Then we get

$$\ell_n^{(in)} = \frac{2}{\sigma_I^2} z_n \tag{11.287}$$

In other words, apart from a constant factor, the LLR associated with the bit c_n coincides with the demodulator output z_n.

Rather than (11.284) and (11.283), we can use the Max-Log-MAP criterion (8.267) that yields an approximate log-likelihood,

$$\tilde{\ell}_{4k-t}^{(in)}(\alpha) = \max_{\substack{i \in \{1, \ldots, N_s\} \\ \boldsymbol{\sigma}_i \text{ with } t\text{-th binary} \\ \text{component} \\ \text{equal to } \alpha}} v_k(i) \tag{11.288}$$

An alternative to the FBA is obtained by modifying the VA to yield a soft output (SOVA), as discussed in the next section.

Convolutional decoding with soft input (SI). The decoder for the convolutional code typically uses the VA with branch metric (associated with a cost function to be minimized) given by

$$\| \boldsymbol{\ell}_k^{(in)} - \mathbf{c}_k \|^2 \tag{11.289}$$

where $\mathbf{c}_k$ is given by (11.274) for a code with $n_0 = 2$, and $\boldsymbol{\ell}_k^{(in)} = [\ell_k^{(in,1)}, \ell_k^{(in,2)}]$ are the LLR associated, respectively, with $c_k^{(1)}$ and $c_k^{(2)}$. As $|c_k^{(j)}|^2 = 1$, (11.289) can be rewritten as:

$$\begin{aligned}(\ell_k^{(in,1)} - c_k^{(1)})^2 + (\ell_k^{(in,2)} - c_k^{(2)})^2 \\ = (\ell_k^{(in,1)})^2 + (\ell_k^{(in,2)})^2 + 2 - 2c_k^{(1)}\ell_k^{(in,1)} - 2c_k^{(2)}\ell_k^{(in,2)}\end{aligned} \tag{11.290}$$

Leaving out the terms that do not depend on $\mathbf{c}_k$, and extending the formulation to a convolutional code with rate k_0/n_0, the branch metric (associated with a cost function to be maximized) is expressed as (see also [7])

$$2\sum_{j=1}^{n_0} c_k^{(j)}\ell_k^{(in,j)} \tag{11.291}$$

where the factor 2 can be omitted.

Observation 11.4
As we have previously stated, best system performance is obtained by jointly designing the encoder and the bit mapper. However in some systems, typically radio, an interleaver is used between the encoder and the bit mapper. In this case joint detection and decoding are impossible to implement in practice. Detection with soft output followed by decoding with soft input remains a valid approach, obviously after re-ordering the LLR as determined by the deinterleaver.

In applications that require a soft output (see Section 11.6), the decoder, that is called in this case *soft-input soft-output* (SISO), can use one of the versions of the FBA or the SOVA.[7]

Sequential decoding

Sequential decoding of convolutional codes represented the first practical algorithm for ML decoding. It has been employed, for example, for the decoding of signals transmitted by deep-space probes, such as the Pioneer, 1968 [10]. There exist several variants of sequential decoding algorithms, that are characterized by the search of the optimum path in a tree diagram (see Figure 11.9b), instead of along a trellis diagram, as considered, e.g., by the VA.

Sequential decoding is an attractive technique for the decoding of convolutional codes and trellis codes if the number of states of the encoder is large [11]. In fact, as the implementation complexity of ML decoders such as the Viterbi decoder grows exponentially with the constraint length of the code, ν, the complexity of sequential decoding algorithms is essentially independent of ν. On the other hand, sequential decoding presents the drawback that the number of computations N_{op} required for the decoding process to advance by one branch in the decoder tree is a random variable with a Pareto distribution, i.e.

$$P[N_{op} > N] = AN^{-\rho} \tag{11.292}$$

[7] An extension of SISO decoders for the decoding of block codes is found in [8, 9].

where A and ρ are constants that depend on the channel characteristics and on the specific code and the specific version of sequential decoding used. Real-time applications of sequential decoders require buffering of the received samples. As practical sequential decoders can perform only a finite number of operations in a given time interval, resynchronization of the decoder must take place if the maximum number of operations that is allowed for decoding without incurring buffer saturation is exceeded.

To determine whether it is practical for a receiver to adopt sequential decoding, we recall the definition of *cut-off rate* of a transmission channel, and the associated minimum signal-to-noise ratio $(E_b/N_0)_0$ (see page 509). Sequential decoders exhibit very good performance, with a reduced complexity as compared to the VA, if the constraint length of the code is sufficiently large and the signal-to-noise ratio is larger than $(E_b/N_0)_0$. If the latter condition is not verified, the average number of operations required to produce one symbol at the decoder output is very large.

The Fano Algorithm In this section we consider sequential decoding of trellis codes, a class of codes that will be studied in detail in Chapter 12. However, the algorithm can be readily extended to convolutional codes. At instant k, the k_0 information bits $\mathbf{b}_k = [b_k^{(1)}, \ldots, b_k^{(k_0)}]$ are input to a rate $k_0/(k_0+1)$ convolutional encoder with constraint length ν that outputs the coded bits $\mathbf{c}_k = [c_k^{(0)}, \ldots, c_k^{(k_0)}]$. The k_0+1 coded bits select from a constellation with $M = 2^{k_0+1}$ elements a symbol a_k, which is transmitted over an additive white Gaussian noise channel. Note that the encoder tree diagram has 2^{k_0} branches that correspond to the values of $\mathbf{b}_k$ stemming from each node. Each branch is labeled by the symbol a_k selected by the vector $\mathbf{c}_k$. The received signal is given by (see (11.275))

$$z_k = a_k + w_k \tag{11.293}$$

The received signal sequence is input to a sequential decoder. Using the notation of Section 8.10.1, in the absence of ISI and assuming $\mathbf{a} = \boldsymbol{\alpha}$, the ML metric to be maximized can be written as [6]

$$\Gamma(\boldsymbol{\alpha}) = \sum_{k=0}^{K-1} \left[\log_2 \frac{P_{z_k|a_k}(\rho_k|\alpha_k)}{\sum_{\alpha_i \in \mathcal{A}} P_{z_k|a_k}(\rho_k|\alpha_i) P_{a_k}(\alpha_i)} - B \right] \tag{11.294}$$

where B is a suitable constant that determines a trade–off between computational complexity and performance and is related to the denominator in (11.294). Choosing $B = k_0$ and $P_{a_k}(\alpha_i) = \frac{1}{M} = 2^{-(k_0+1)}$, $\alpha_i \in \mathcal{A}$, we obtain

$$\Gamma(\boldsymbol{\alpha}) = \sum_{k=0}^{K-1} \left[\log_2 \frac{e^{-\frac{|\rho_k - \alpha_k|^2}{2\sigma_I^2}}}{\sum_{\alpha_i \in \mathcal{A}} e^{-\frac{|\rho_k - \alpha_i|^2}{2\sigma_I^2}}} + 1 \right] \tag{11.295}$$

Various algorithms have been proposed for sequential decoding [12, 13, 14]. We will restrict our attention here to the Fano algorithm [6, 11].

The Fano algorithm examines only one path of the decoder tree at any time using the metric in (11.294). The considered path extends to a certain node in the tree and corresponds to a segment of the entire code sequence $\boldsymbol{\alpha}$, up to symbol α_k.

Three types of moves between consecutive nodes on the decoder tree are allowed: forward, lateral, and backward. On a *forward move*, the decoder goes one branch to the right in the decoder tree from the previously hypothesized node. This corresponds to the insertion of a new symbol α_{k+1} in (11.294). On a *lateral move*, the decoder goes from a path on the tree to another path differing only in the last branch. This corresponds to the selection of a different symbol α_k in (11.294). The ordering among the nodes is arbitrary, and a lateral move takes place to the next node in order after the current one. A *backward move* is a move one branch to the left on the tree. This corresponds to the removal of the symbol α_k from (11.294).

To determine which move needs to be made after reaching a certain node, it is necessary to compute the metric Γ_k of the current node being hypothesized, and consider the value of the metric Γ_{k-1} of the node one branch to the left of the current node, as well as the current value of a threshold Th, which can assume values that are multiples of a given constant Δ. The transition diagram describing the Fano algorithm is illustrated in Figure 11.20. Typically, Δ assumes values that are of the order of the minimum distance between symbols.

As already mentioned, real-time applications of sequential decoding require buffering of the input samples with a buffer of size S. Furthermore, the depth of backward search is also finite and is usually chosen to be at least five times the constraint length of the code. To avoid erasures of output symbols in case of buffer saturation, in [15] a *buffer looking algorithm* (BLA) is proposed. The buffer is divided into L sections, each with size S_j, $j = 1, \ldots, L$. A conventional sequential decoder (primary decoder) and $L - 1$ secondary decoders are used. The secondary decoders employ fast algorithms, such as the M-algorithm [16], or variations of the Fano algorithm that are obtained by changing the value of the bias B in the metric (11.294).

Example 11.3.2 (Sequential decoding of a 512-state 16-PAM trellis code)
We illustrate sequential decoding with reference to a 512-state 16-PAM trellis code specified for SHDSL transmission (see Chapter 17). The encoder for this trellis code comprises a rate $1/2$ nonsystematic non-recursive convolutional encoder with constraint length $\nu = 9$ and a bit mapper as specified in Figure 11.21. The symbol error probabilities versus signal-to-noise ratio Γ obtained by sequential decoding with infinite buffer size and depth of backward search of 64 and 128 symbols, and by a 512−state VA decoding with length of the path memory of 64 and 128 symbols are shown in Figure 11.22. Also shown for comparison are the error probabilities obtained for uncoded 8-PAM and 16-PAM transmission.

11.3.3 Performance of convolutional codes

For binary convolutional codes with free distance d_{free}^H, and bit error probability of the channel equal to P_{bit}, decoding with *hard input* yields

$$P_{bit}^{(dec)} \simeq A\, 2^{d_{free}^H}\, P_{bit} \tag{11.296}$$

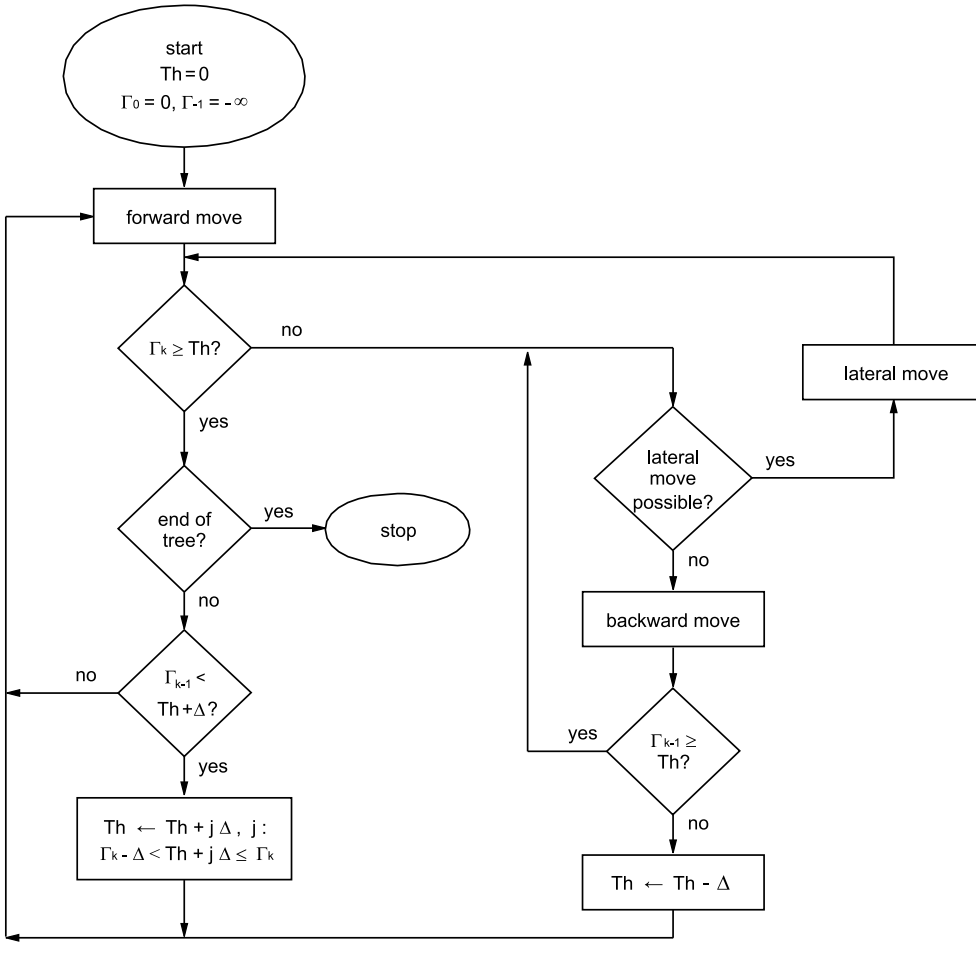


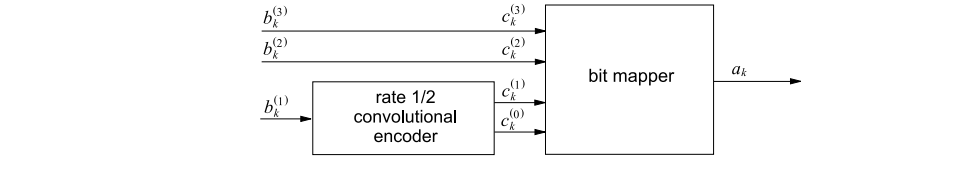
Figure 11.20. Transition diagram of the Fano algorithm.

and decoding with *soft input*, for a system with antipodal signals, yields

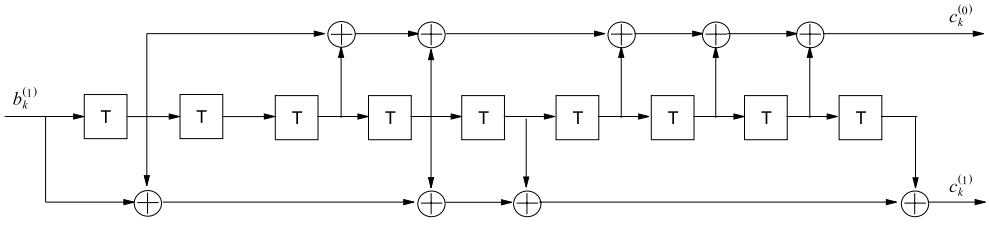
$$P_{bit}^{(dec)} \simeq A\, Q\left(\sqrt{\frac{R_c\, d_{free}^{H}\, 2E_b}{N_0}}\right) \tag{11.297}$$

where A is a constant [17].

In particular, we consider BPSK transmission over an ideal AWGN channel. Assuming an encoder with rate $R_c = 1/2$ and constraint length $\nu = 6$, the coding gain for a soft Viterbi decoder is about 3.5 dB for $P_{bit} = 10^{-3}$; it becomes about 4.6 dB for $P_{bit} = 10^{-5}$. Note that a soft decoder allows gain of about 2.4 dB with respect to a hard decoder, for $P_{bit} < 10^{-3}$.



(a)



(b)

-15	-13	-11	-9	-7	-5	-3	-1	1	3	5	7	9	11	13	15	a_k
0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	$c_k^{(0)}$
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	$c_k^{(1)}$
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	$c_k^{(2)}$
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	$c_k^{(3)}$

(c)

Figure 11.21. (a) Block diagram of the encoder and bit mapper for a trellis code for 16-PAM transmission, (b) structure of the rate-1/2 convolutional encoder, and (c) bit mapping for the 16-PAM format.

11.4 Concatenated codes

Concatenated coding is usually introduced to achieve an improved error correction capability [18]. Interleaving is also commonly used in concatenated coding schemes, as illustrated in Figure 11.23, so that the decoding processes of the two codes (inner and outer) can be considered approximately independent. The first decoding stage is generally utilized to produce soft decisions on the information bits, that are passed to the second decoding stage. While the forward-backward algorithm directly provides a soft output (see (11.285)), the Viterbi algorithm must be slightly modified.

Soft-output Viterbi algorithm (SOVA)

We have seen that the FBA in the original MAP version directly yields a soft output, at the expense of a large computational complexity (see page 916). The Max-Log-MAP criterion has a reduced complexity, but there remains the problem of having to perform the two procedures, forward and backward. We now illustrate how to modify the VA to obtain a soft output equivalent to that produced by the Max-Log-MAP.

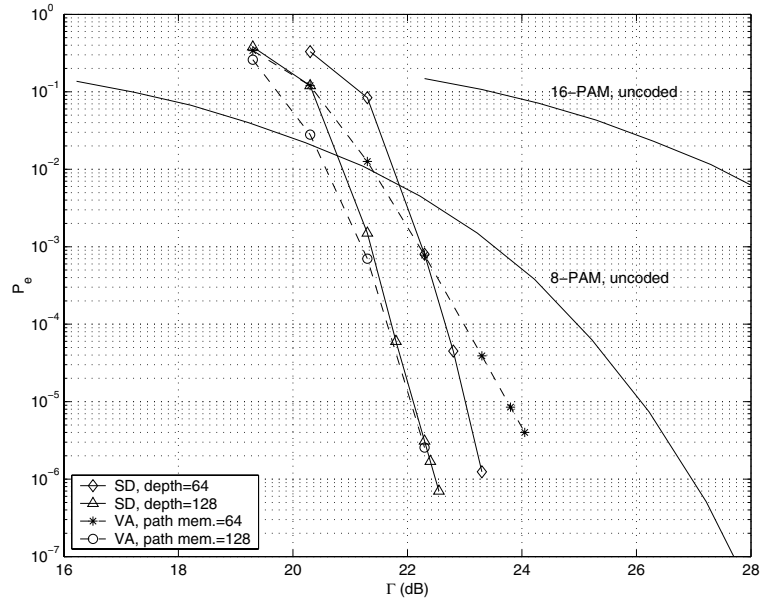


Figure 11.22. Symbol error probabilities for the 512-state 16-PAM trellis code with sequential decoding (depth of search limited to 64 or 128) and 512-state Viterbi decoding (length of path memory limited to 64 or 128). Symbol error probabilities for uncoded 8-PAM and 16-PAM transmission are also shown.

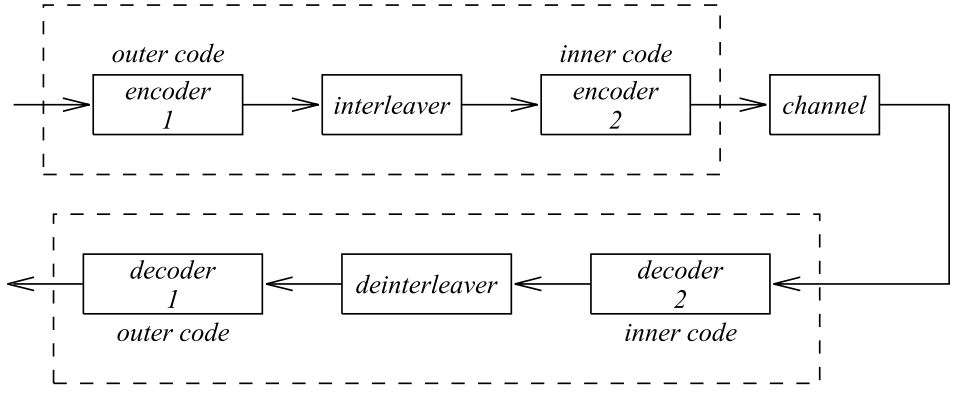


Figure 11.23. Transmission scheme with concatenated codes and interleaver.

In this section we consider different methods to generate the soft output.

The difference metric algorithm (DMA). Figure 11.24 shows a section of a trellis diagram with four states, where we assume $\mathbf{s}_k = (b_k, b_{k-1})$. We consider two states at instant $k-1$ that differ in the least significant bit b_{k-2} of the binary representation, that is $\mathbf{s}_{k-1}^{(0)} = (00)$

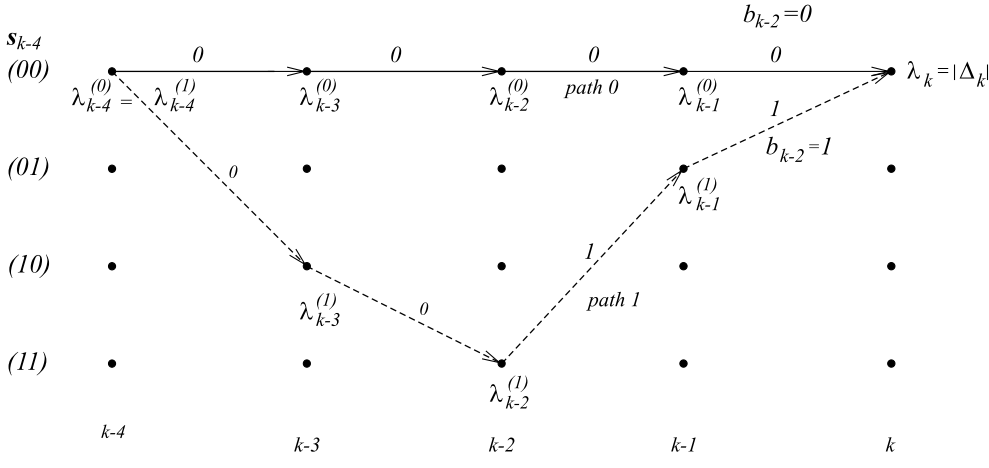


Figure 11.24. Soft-output Viterbi algorithm.

and $\mathbf{s}_{k-1}^{(1)} = (01)$. A transition from each of these two states to state $\mathbf{s}_k = (00)$ at instant k is allowed. According to the VA, we choose as survivor sequence the sequence that minimizes the metric

$$\min_{i\in\{0,1\}} \left\{ \Gamma_{k-1}(\mathbf{s}_{k-1}^{(i)}) + \gamma_k^{\mathbf{s}_{k-1}^{(i)} \to \mathbf{s}_k} \right\} \tag{11.298}$$

where $\Gamma_{k-1}(\mathbf{s}_{k-1}^{(i)})$ is the path metric associated with the survivor sequence up to state $\mathbf{s}_{k-1}^{(i)}$ at instant $k-1$, and $\gamma_k^{\mathbf{s}_{k-1}^{(i)} \to \mathbf{s}_k}$ denotes the branch metric associated with the transition from state $\mathbf{s}_{k-1}^{(i)}$ to state $\mathbf{s}_k$. Let $A_k = \Gamma_{k-1}(00) + \gamma_k^{00\to 00}$ and $B_k = \Gamma_{k-1}(01) + \gamma_k^{01\to 00}$, then we choose the upper or the lower transition according to whether $\Delta_k = A_k - B_k$ is smaller or larger than zero, respectively. Note that $|\Delta_k|$ is a reliability measure of the selection of a certain sequence as survivor sequence.

In other words, if $|\Delta_k|$ is small, there is a non-negligible probability that the bit b'_{k-2} associated with the transition from state $\mathbf{s}_{k-1}^{(i)}$ to $\mathbf{s}_k$ on the survivor sequence is in error. The difference $|\Delta_k| = \lambda_k$ yields the value of the soft decision for b_{k-2}, in case the final sequence chosen by the Viterbi algorithm includes the state $\mathbf{s}_k$; conversely, this information is disregarded. Thus the DMA can be formulated as follows.

For each state $\mathbf{s}_k$ of the trellis diagram at instant k, the metric $\Gamma_k(\mathbf{s}_k)$ and the most recent $(K_d + 1)$ bits of the survivor sequence $\mathbf{b}'(\mathbf{s}_k) = \{b'_k, \ldots, b'_{k-K_d}\}$ are memorized, where K_d denotes the path memory depth of the VA. Furthermore, the reliability measures $\boldsymbol{\lambda}(\mathbf{s}_k) = \{\lambda_k, \ldots, \lambda_{k-K_d}\}$ associated with the bits $\mathbf{b}'(\mathbf{s}_k)$ are also memorized. Interpreting b_k and $\hat{b}_k$ as binary symbols in the alphabet $\{-1, 1\}$ (see note 6 on page 915), the soft output associated with b_k is given by

$$\tilde{\ell}_k = \hat{b}_k \, \lambda_{k+2} \tag{11.299}$$

where $\{\hat{b}_k\}$ is the sequence of information bits associated with the ML sequence.

The soft-output VA (SOVA). As for the DMA, the SOVA determines the difference between the metrics of the survivor sequences on the paths that converge to each state of the trellis, and updates at every instant k the reliability information $\boldsymbol{\lambda}(\mathbf{s}_k)$ for each state of the trellis. To perform this update, the sequences on the paths that converge to a certain state are compared to identify the positions at which the information bits of the two sequences differ. With reference to Figure 11.24, we denote the two paths that converge to the state (00) at instant k as *path 0* and *path 1*. Without loss of generality we assume that the sequence associated with path 0 is the survivor sequence, and thus the sequence with the smaller cost; furthermore we define $\boldsymbol{\lambda}(\mathbf{s}_k^{(0)}) = \{\lambda_k^{(0)}, \ldots, \lambda_{k-K_d}^{(0)}\}$ and $\boldsymbol{\lambda}(\mathbf{s}_k^{(1)}) = \{\lambda_k^{(1)}, \ldots, \lambda_{k-K_d}^{(1)}\}$ as the two reliability vectors associated with the information bits of two sequences. If one information bit *along path 0* differs from the corresponding information bit *along path 1*, then its reliability is updated according to the rule

$$\begin{aligned} &\text{for } i = k - K_d, \ldots, k-1 \\ &\quad \lambda_i = \min(|\Delta_k|, \lambda_i^{(0)}) \qquad \text{if } b_{i-2}^{(0)} \neq b_{i-2}^{(1)} \end{aligned} \tag{11.300}$$

With reference to Figure 11.24, the two sequences on path 0 and on path 1 diverge from state $\mathbf{s}_k = (00)$ at instant $k-4$. The two sequences differ in the associated information bits at the instants k and $k-1$; therefore, only λ_{k-1} will be updated.

Modified SOVA. In the modified version of the SOVA, the reliability of an information bit along the survivor path is also updated if the information bit is the same, according to the rule

$$\text{for } i = k - K_d, \ldots, k-1$$

$$\lambda_i = \begin{cases} \min\left(|\Delta_k|, \lambda_i^{(0)}\right) & \text{if } b_{i-2}^{(0)} \neq b_{i-2}^{(1)} \\ \min\left(|\Delta_k| + \lambda_i^{(1)}, \lambda_i^{(0)}\right) & \text{if } b_{i-2}^{(0)} = b_{i-2}^{(1)} \end{cases} \tag{11.301}$$

Note that (11.300) is still used to update the reliability if the information bits differ; this version of the SOVA gives a better estimate of λ_i. As proved in [19], if the VA is used as decoder, the modified SOVA is equivalent to Max-Log-MAP decoding. An example of how the modified SOVA works is illustrated in Figure 11.25.

11.5 Turbo codes

Turbo codes, proposed in 1993 by Berrou and Glavieux [20, 21], constitute an evolution of concatenated codes, in the form of parallel concatenated convolutional codes (PCCC), and allow reliable transmission of information at rates near the Shannon limit [20, 21, 22]. As will be discussed in this section, the term turbo, even though it is used to qualify these codes, is rather tied to the decoder, whose principle is reminiscent of that of turbo engines.

Encoding

For the description of turbo codes, we refer to the first code of this class that appeared in the scientific literature [20, 21]. A sequence of information bits is encoded by a simple

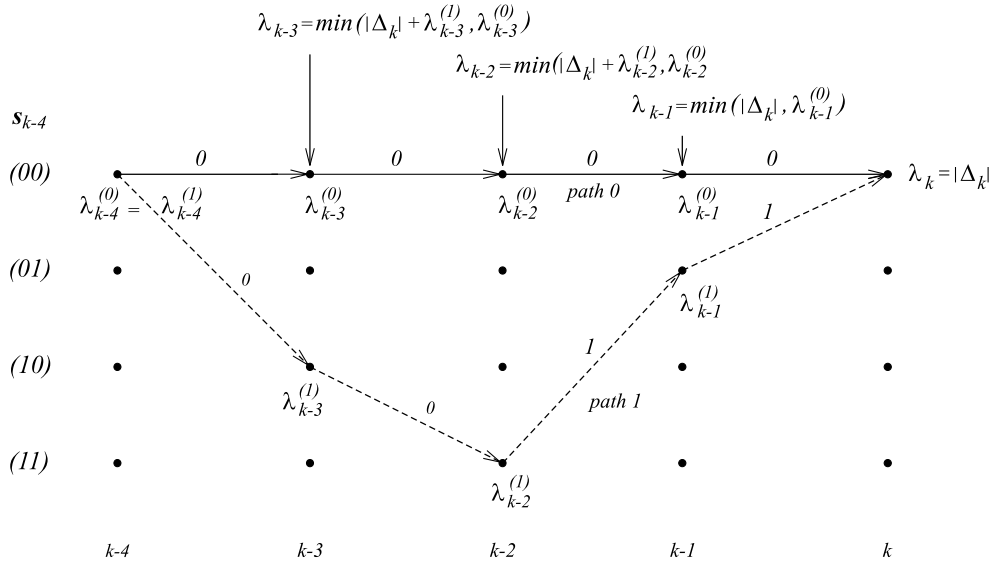


Figure 11.25. Modified soft-output Viterbi algorithm.

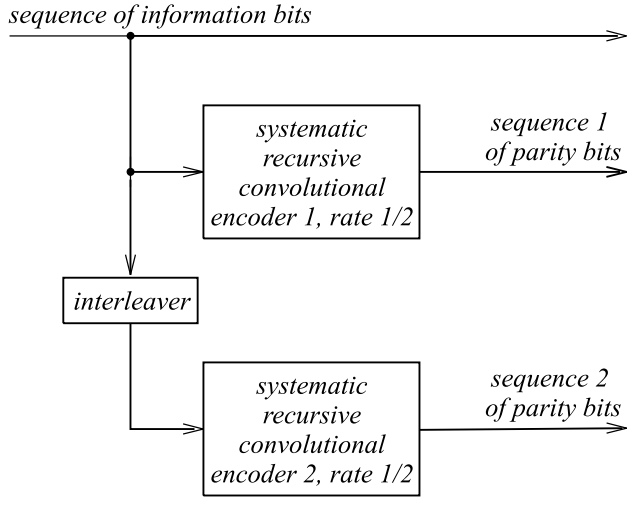


Figure 11.26. Encoder of a turbo code with code rate $R_c = \frac{1}{3}$.

recursive systematic convolutional (RSC) binary encoder with code rate $1/2$, to produce a first sequence of parity bits, as illustrated in Figure 11.26. The same sequence of information bits is permuted by a long interleaver and then encoded by a second recursive systematic convolutional encoder with code rate $1/2$ to produce a second sequence of parity bits. Then the sequence of information bits and the two sequences of parity bits are transmitted. Note

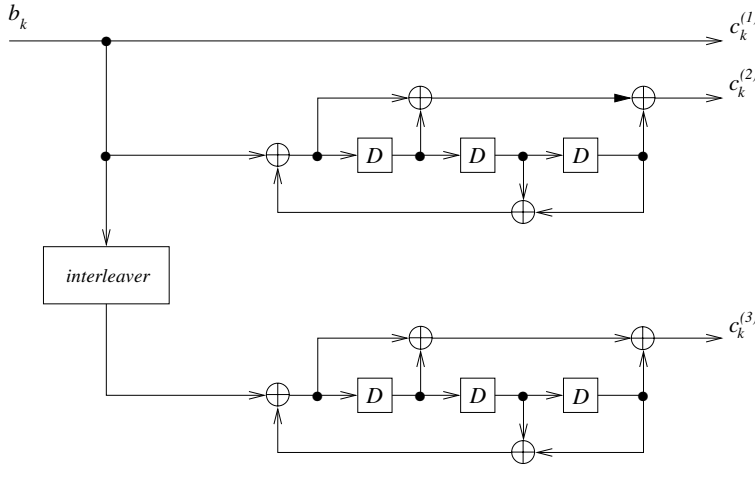


Figure 11.27. Turbo encoder adopted by the UMTS standard.

that the resulting code has rate $R_c = 1/3$. Higher code rates R_c are obtained by transmitting only some of the parity bits (puncturing). For example, for the turbo code in [20, 21], a code rate equal to 1/2 is obtained by transmitting only the bits of the parity sequence 1 with odd indices, and the bits of the parity sequence 2 with even indices. A specific example of turbo encoder is reported in Figure 11.27.

The exceptional performance of turbo codes is due to one particular characteristic. We can think of a turbo code as being a block code for which an input word has a length equal to the interleaver length, and a code word is generated by initializing to zero the memory elements of the convolutional encoders before the arrival of each input word. This block code has a group structure. As for the usual block codes, the asymptotic performance, for large values of the signal-to-noise ratio, is determined by the code words of minimum weight and by their number. For low values of the signal-to-noise ratio, also the code words of non-minimum weight and their multiplicity need to be taken into account. Before the introduction of turbo codes, the focus on designing codes was mainly on asymptotic performance, and thus on maximizing the minimum distance. With turbo codes, the approach is different. Because of the large ensemble of code words, the performance curve, in terms of bit error probability as a function of the signal-to-noise ratio, rapidly decreases for low values of the signal-to-noise ratio. For P_{bit} lower than 10^{-5}, where performance is determined by the minimum distance between code words, the bit error probability curve usually exhibits a reduction in the value of slope.

The two encoders that compose the scheme of Figure 11.26 are called *component encoders* and they are usually identical. As mentioned above, Berrou and Glavieux proposed two recursive systematic convolutional encoders as component encoders. Later it was shown that it is not necessary to use systematic encoders [23, 17].

Recursive convolutional codes are characterized by the property that the code bits at a given instant do not depend only on the information bits at the present instant and the

previous ν instants, where ν is the constraint length of the code, but on all the previous information bits, as the encoder exhibits a structure with feedback.

Starting from a non-recursive nonsystematic convolutional encoder for a code with rate $1/n_0$, it is possible to obtain in a very simple way a recursive systematic encoder for a code with the same rate and the same code words, and hence with the same free distance d_{free}^H. Obviously, for a given input word, the output code words will be different in the two cases. Consider for example a non-recursive nonsystematic convolutional encoder for a code with code rate $1/2$. The code bits can be expressed in terms of the information bits as (see (11.254))

$$\begin{aligned} c'^{(1)}(D) &= g^{(1,1)}(D)\, b(D) \\ c'^{(2)}(D) &= g^{(2,1)}(D)\, b(D) \end{aligned} \tag{11.302}$$

The corresponding recursive systematic encoder is obtained by dividing the polynomials in (11.302) by $g^{(1,1)}(D)$, and implementing the functions

$$c^{(1)}(D) = b(D) \tag{11.303}$$

$$c^{(2)}(D) = \frac{g^{(2,1)}(D)}{g^{(1,1)}(D)}\, b(D) \tag{11.304}$$

Let us define

$$d(D) = \frac{c^{(2)}(D)}{g^{(2,1)}(D)} = \frac{b(D)}{g^{(1,1)}(D)} \tag{11.305}$$

then the code bits can be expressed as a function of the information bits and the bits of the sequence $\{d_k\}$ as

$$c_k^{(1)} = b_k \tag{11.306}$$

$$c_k^{(2)} = \sum_{i=0}^{\nu} g_i^{(2,1)} d_{k-i} \tag{11.307}$$

where, using the fact that $g_0^{(1,1)}(D) = 1$, from (11.305) we get

$$d_k = b_k + \sum_{i=1}^{\nu} g_i^{(1,1)} d_{k-i} \tag{11.308}$$

We recall that the operations in the above equations are in GF(2). Another recursive systematic encoder that generates a code with the same free distance is obtained by exchanging the role of the polynomials $g^{(1,1)}(D)$ and $g^{(2,1)}(D)$ in the above equations.

One recursive systematic encoder corresponding to the non-recursive nonsystematic encoder of Figure 11.9(a) is illustrated in Figure 11.28.

The 16-state component encoder for a code with code rate $1/2$ used in the turbo code of Berrou and Glavieux [20, 21], is shown in Figure 11.29. The encoder in Figure 11.27, with an 8-state component encoder for a code with code rate 1/2, is adopted in the standard for third generation *universal mobile telecommunications systems* (UMTS) [24]. Turbo codes

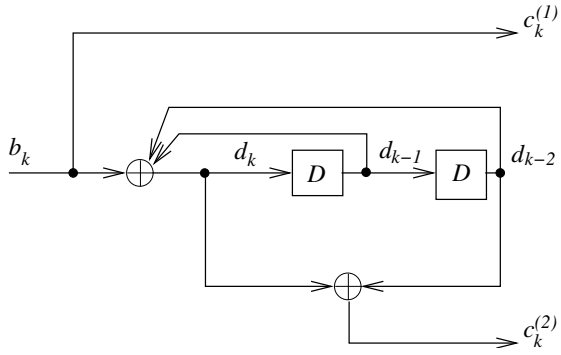


Figure 11.28. Recursive systematic encoder that generates a code with the same free distance as the non-recursive nonsystematic encoder of Figure 11.9(a).

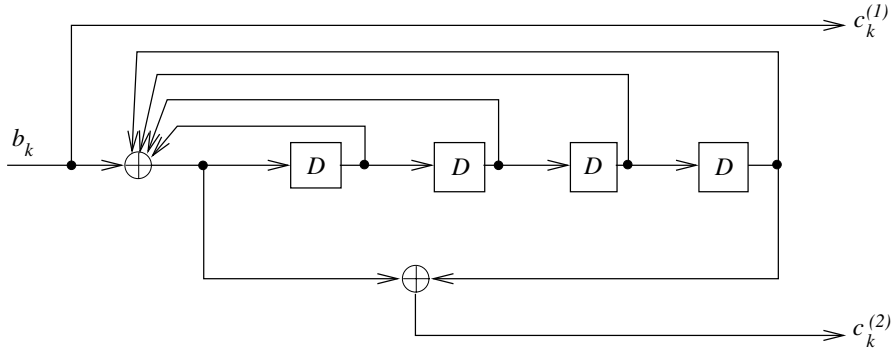


Figure 11.29. A 16-state component encoder for the turbo code of Berrou and Glavieux.

are also used in *digital video broadcasting* (DVB) [25] standards and in space telemetry applications as defined by the *Consultative Committee for Space Data Systems* (CCSDS) [26]. In [27] are listed generator polynomials of recursive systematic convolutional encoders for codes with rates 1/2, 1/3, 1/4, 2/3, 3/4, 4/5, and 2/4, that can be used for the construction of turbo codes.

Another fundamental component in the structure of turbo codes is represented by a *non-uniform* interleaver. We recall that a uniform[8] interleaver, as that described in Section 11.3.2, operates by writing input bits in a matrix by rows and reading them by columns. In practice, a non-uniform interleaver determines the permutation of the sequence of input bits so that adjacent bits in the input sequence are separated, after the permutation, by a number of bits that varies with the position of the bits in the input sequence. The interleaver determines directly the minimum distance of the code and therefore performance for high values of the signal-to-noise ratio. Nevertheless, the choice of the interleaver is not critical for low values of the signal-to-noise ratio. Beginning with the interleaver originally proposed

[8] The adjective "uniform", referred to an interleaver, is used with a different meaning in [23].

in [20, 21], various interleavers have since been proposed (see [28] and references contained therein).

One of the interleavers that yields better performance is the so-called spread interleaver [29]. Consider a block of M_1 input bits. The integer numbers that indicate the position of these bits after the permutation are randomly generated with the following constraint: each integer randomly generated is compared with the S_1 integers previously generated; if the distance from them is shorter than a prefixed threshold S_2, the generated integer is discarded and another one is generated until the condition is satisfied. The two parameters S_1 and S_2 must be larger than the memory of the two-component encoders. If the two-component encoders are equal, it is convenient to choose $S_1 = S_2$. The computation time needed to generate the interleaver increases with S_1 and S_2, and there is no guarantee that the procedure terminates successfully. Empirically, it has been verified that, choosing both S_1 and S_2 equal to the closest integer to $\sqrt{M_1/2}$, it is possible to generate the interleaver in a finite number of steps.

Many variations of the basic idea of turbo codes have been proposed. For example, codes generated by serial concatenation of two convolutional encoders, connected by means of a non-uniform interleaver [30]. Parallel and serial concatenation schemes were then extended to the case of multilevel constellations to obtain coded modulation schemes with high spectral efficiency (see [31] and references contained therein).

The basic principle of iterative decoding

The presence of the interleaver in the scheme of Figure 11.26 makes an encoder for a turbo code have a very large memory even if very simple component encoders are used. Therefore the optimum MLSD decoder would require a Viterbi decoder with an exceedingly large number of states and it would not be realizable in practice. For this reason we resort to a suboptimum iterative decoding scheme with a much lower complexity than the optimum scheme. However, as it was verified empirically, this scheme exhibits near optimum performance [23]. The decoder for the turbo encoder of Figure 11.26 is illustrated in Figure 11.30. The received sequences corresponding to the sequence of information bits and the first sequence of parity bits are decoded using a soft input soft output decoder, corresponding to the first convolutional encoder. Thus this decoder provides a soft decision for each information bit; these soft decisions are then used by a second decoder corresponding to the second convolutional encoder, together with the received sequence corresponding to the second sequence of parity bits. Soft decisions thus obtained are taken back to the input of the first decoder for a new iteration, where the additional information obtained by the second decoder is used to produce more reliable soft decisions. The procedure continues iteratively for about 10–20 cycles until final decisions are made on the information bits.

The two component decoders of the scheme in Figure 11.30 are soft input soft output decoders that produce estimates on the reliability of the decisions. Therefore they implement the SOVA or the FBA (or one of its simplified realizations). The basic principle of iterative decoding is the following: each component decoder uses the "hints" of the other to produce more reliable decisions. In the next sections we will see in detail how this is achieved, and in particular how the reliability of the decisions is determined.

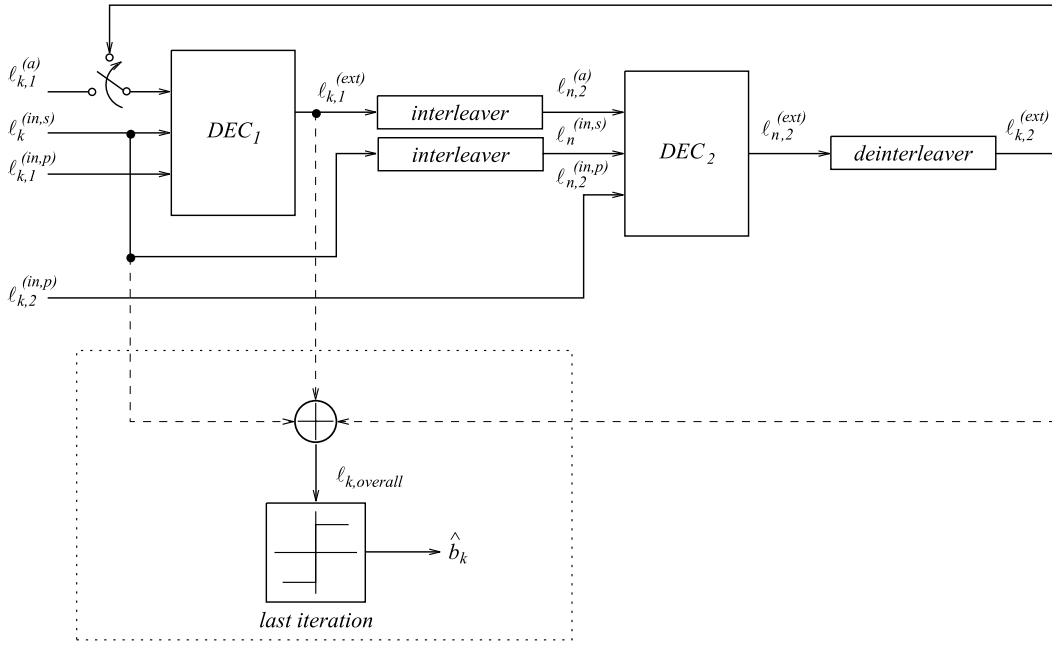


Figure 11.30. Principle of the decoder for a turbo code with rate 1/3.

The algorithms for iterative decoding introduced with the turbo codes were also immediately applied in wider contexts. In fact, this iterative procedure may be used every time the transmission system includes multiple processing elements with memory interconnected by an interleaver. Iterative decoding procedures may be used, for example, for detection in the presence of intersymbol interference, also called turbo equalization or turbo detection [32] (see Section 11.6), for non-coherent decoding [33, 34], and for joint detection and decoding in the case of transmission over channels with fading [35].

Before discussing in detail iterative decoding, it is useful to revisit the FBA.

The forward-backward algorithm revisited

The formulation of the FBA presented here is useful for the decoding of recursive systematic convolutional codes [36].

We consider a binary recursive systematic convolutional encoder for a code with rate k_0/n_0, and constraint length ν. Let the encoder input be given by a sequence of K vectors, each composed of k_0 binary components. As described on page 903, each information vector to be encoded is denoted by (see (11.252))

$$\mathbf{b}_k = [b_k^{(1)}, b_k^{(2)}, \ldots, b_k^{(k_0)}] \qquad b_k^{(i)} \in \{-1, 1\} \qquad k = 0, 1, \ldots K-1 \tag{11.309}$$

where k_0 can be seen either as the number of encoder inputs or as the length of an information vector. As the convolutional encoder is systematic, at instant k the state of the

convolutional encoder is given by the vector (see extension of (11.308))

$$\mathbf{s}_k = [\, d_k^{(k_0+1)}, \ldots, d_{k-\nu+1}^{(k_0+1)}, d_k^{(k_0+2)}, \ldots, d_{k-\nu+1}^{(k_0+2)}, \ldots, d_k^{(n_0)}, \ldots, d_{k-\nu+1}^{(n_0)} \,] \tag{11.310}$$

which has a number of components $N_2 = \nu \cdot (n_0 - k_0)$, equal to the number of the encoder memory elements. The set of states $\mathcal{S}$, that is the possible values assumed by $\mathbf{s}_k$, is given by

$$\mathbf{s}_k \in \mathcal{S} = \{\boldsymbol{\sigma}_1, \boldsymbol{\sigma}_2, \ldots, \boldsymbol{\sigma}_{N_s}\} \tag{11.311}$$

where $N_s = 2^{N_2}$ is the number of encoder states. It is important to observe that the convolutional encoder can be seen as a sequential finite-state machine with i.i.d. input $\mathbf{b}_k$, and state transition function $\mathbf{s}_k = f_s(\mathbf{b}_k, \mathbf{s}_{k-1})$. Hence, for a given information vector $\mathbf{b}_k$, the transition from state $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i$ to state $\mathbf{s}_k = \boldsymbol{\sigma}_j$ is unique, in correspondence of which a code vector is generated, that is expressed as

$$\mathbf{c}_k = [c_k^{(1)}, c_k^{(2)}, \ldots, c_k^{(k_0)}, c_k^{(k_0+1)}, \ldots, c_k^{(n_0)}] = [\, \mathbf{c}_k^{(s)}, \mathbf{c}_k^{(p)} \,] \tag{11.312}$$

where the superscript $^{(s)}$ denotes systematic bits, and $^{(p)}$ denotes parity check bits. Then $\mathbf{c}_k^{(s)} = \mathbf{b}_k$, and from (11.307) we can express $\mathbf{c}_k^{(p)}$ as a function of $\mathbf{s}_k$ and $\mathbf{s}_{k-1}$ as

$$\mathbf{c}_k^{(p)} = f^{(p)}(\mathbf{s}_k, \mathbf{s}_{k-1}) \tag{11.313}$$

The values assumed by the code vectors are indicated by

$$\boldsymbol{\beta} = [\, \beta^{(1)}, \beta^{(2)}, \ldots, \beta^{(k_0)}, \beta^{(k_0+1)}, \ldots, \beta^{(n_0)} \,] = [\, \boldsymbol{\beta}^{(s)}, \boldsymbol{\beta}^{(p)} \,] \tag{11.314}$$

For simplicity, we assume that the code binary symbols so determined are transmitted by a binary modulation scheme over an AWGN channel. In this case, at the decision point of the receiver, we get the signal (see (11.275)),

$$z_k = c_k + w_k \tag{11.315}$$

where $c_k \in \{-1, 1\}$ denotes a code bit, and $\{w_k\}$ is a sequence of real-valued i.i.d. Gaussian noise samples with variance σ_I^2. It is useful to organize the samples $\{z_k\}$ into subsequences that follow the structure of the code vectors (11.312). Then we introduce the vectors

$$\mathbf{z}_k = [\, z_k^{(1)}, z_k^{(2)}, \ldots, z_k^{(k_0)}, z_k^{(k_0+1)}, \ldots, z_k^{(n_0)} \,] = [\, \mathbf{z}_k^{(s)}, \mathbf{z}_k^{(p)} \,] \tag{11.316}$$

As usual we denote as $\boldsymbol{\rho}_k$ an observation of $\mathbf{z}_k$,

$$\boldsymbol{\rho}_k = [\, \rho_k^{(1)}, \rho_k^{(2)}, \ldots, \rho_k^{(k_0)}, \rho_k^{(k_0+1)}, \ldots, \rho_k^{(n_0)} \,] = [\, \boldsymbol{\rho}_k^{(s)}, \boldsymbol{\rho}_k^{(p)} \,] \tag{11.317}$$

We recall from Section 8.10 that the FBA yields the *detection of the single information vector* $\mathbf{b}_k$, $k = 0, 1, \ldots, K-1$, expressed as

$$\hat{\mathbf{b}}_k = [\, \hat{b}_k^{(1)}, \hat{b}_k^{(2)}, \ldots, \hat{b}_k^{(k_0)} \,] \tag{11.318}$$

through the computation of the *a posteriori* probability. We also recall that in general for a sequence $\mathbf{a} = [\mathbf{a}_0, \ldots, \mathbf{a}_k, \ldots, \mathbf{a}_{K-1}]$, with the notation $\mathbf{a}_l^m$ we indicate the subsequence formed by the components $[\mathbf{a}_l, \mathbf{a}_{l+1}, \ldots, \mathbf{a}_m]$.

Defining the likelihood of the generic information vector (see (8.220)),

$$\mathtt{L}_k(\boldsymbol{\beta}^{(s)}) = P[\mathbf{b}_k = \boldsymbol{\beta}^{(s)} \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \tag{11.319}$$

detection by the MAP criterion is expressed as

$$\hat{\mathbf{b}}_k = \arg\max_{\boldsymbol{\beta}^{(s)}} \mathtt{L}_k(\boldsymbol{\beta}^{(s)}) \qquad k = 0, 1, \ldots, K-1 \tag{11.320}$$

We note that the likelihood associated with the individual bits of the information vector $\mathbf{b}_k$ are obtained by suitably adding (11.319), as

$$\begin{aligned} \mathtt{L}_{k,i}(\alpha) &= P[b_k^{(i)} = \alpha \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \\ &= \sum_{\substack{\boldsymbol{\beta}^{(s)} \in \{-1,1\}^{k_0} \\ \beta^{(i)} = \alpha}} \mathtt{L}_k(\boldsymbol{\beta}^{(s)}) \qquad \alpha \in \{-1, 1\} \end{aligned} \tag{11.321}$$

In a manner similar to the analysis of page 668, we introduce the following quantities:

1. The state transition probability $\Pi(j \mid i) = P[\mathbf{s}_k = \boldsymbol{\sigma}_j \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i]$, that assumes non-zero values only if there is a transition from the state $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i$ to the state $\mathbf{s}_k = \boldsymbol{\sigma}_j$ for a certain input $\boldsymbol{\beta}^{(s)}$, and we write

$$\Pi(j \mid i) = P[\mathbf{b}_k = \boldsymbol{\beta}^{(s)}] = \mathtt{L}_k^{(a)}(\boldsymbol{\beta}^{(s)}) \tag{11.322}$$

   $\mathtt{L}_k^{(a)}(\boldsymbol{\beta}^{(s)})$ is called the *a priori information* on the information vector $\mathbf{b}_k = \boldsymbol{\beta}^{(s)}$, and is one of the soft inputs.

2. For an AWGN channel the channel transition probability $p_{\mathbf{z}_k}(\boldsymbol{\rho}_k \mid j, i)$ can be separated into two contributions, one due to the systematic bits and the other to the parity check bits,

$$\begin{aligned} p_{\mathbf{z}_k}(\boldsymbol{\rho}_k \mid j, i) &= P[\mathbf{z}_k = \boldsymbol{\rho}_k \mid \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] \\ &= P[\mathbf{z}_k^{(s)} = \boldsymbol{\rho}_k^{(s)} \mid \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] \\ &\quad P[\mathbf{z}_k^{(p)} = \boldsymbol{\rho}_k^{(p)} \mid \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] \\ &= P[\mathbf{z}_k^{(s)} = \boldsymbol{\rho}_k^{(s)} \mid \mathbf{c}_k^{(s)} = \boldsymbol{\beta}^{(s)}] \; P[\mathbf{z}_k^{(p)} = \boldsymbol{\rho}_k^{(p)} \mid \mathbf{c}_k^{(p)} = \boldsymbol{\beta}^{(p)}] \\ &= \left(\left(\frac{1}{\sqrt{2\pi\sigma_I^2}} \right)^{k_0} e^{-\frac{1}{2\sigma_I^2} \|\boldsymbol{\rho}_k^{(s)} - \boldsymbol{\beta}^{(s)}\|^2} \right) \\ &\quad \left(\left(\frac{1}{\sqrt{2\pi\sigma_I^2}} \right)^{n_0 - k_0} e^{-\frac{1}{2\sigma_I^2} \|\boldsymbol{\rho}_k^{(p)} - \boldsymbol{\beta}^{(p)}\|^2} \right) \end{aligned} \tag{11.323}$$

3. We merge (11.322) and (11.323) into one variable (see (8.229)),

$$C_k(j \mid i) = P[\mathbf{z}_k = \boldsymbol{\rho}_k, \mathbf{s}_k = \boldsymbol{\sigma}_j \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i] = p_{\mathbf{z}_k}(\boldsymbol{\rho}_k \mid j, i)\, \Pi(j \mid i)$$
$$= \left(\frac{1}{\sqrt{2\pi\sigma_I^2}} \right)^{n_0} C_k^{(s)}(j \mid i)\, C_k^{(p)}(j \mid i) \tag{11.324}$$

where

$$C_k^{(s)}(j \mid i) = e^{-\frac{1}{2\sigma_I^2}\|\boldsymbol{\rho}_k^{(s)} - \boldsymbol{\beta}^{(s)}\|^2}\, \mathtt{L}_k^{(a)}(\boldsymbol{\beta}^{(s)}) \tag{11.325}$$

$$C_k^{(p)}(j \mid i) = e^{-\frac{1}{2\sigma_I^2}\|\boldsymbol{\rho}_k^{(p)} - \boldsymbol{\beta}^{(p)}\|^2} \tag{11.326}$$

The two previous quantities are related, respectively, to the systematic bits and the parity check bits of a code vector. Observe that the exponential term in (11.325) represents the reliability of a certain *a priori* information $\mathtt{L}_k^{(a)}(\boldsymbol{\beta}^{(s)})$ associated with $\boldsymbol{\beta}^{(s)}$.

4. The computation of the forward and backward metrics is carried out as in the general case.

   - *Forward metric*, for $k = 0, 1, \ldots, K-1$:

$$F_k(j) = \sum_{\ell=1}^{N_s} C_k(j \mid \ell)\, F_{k-1}(\ell) \qquad j = 1, \ldots, N_s \tag{11.327}$$

   - *Backward metric*, for $k = K-1, K-2, \ldots, 0$:

$$B_k(i) = \sum_{m=1}^{N_s} B_{k+1}(m)\, C_{k+1}(m \mid i) \qquad i = 1, \ldots, N_s \tag{11.328}$$

   Suitable initializations are obtained, respectively, through (8.237) and (8.244).

5. By using the total probability theorem, the likelihood (11.319) can be written as

$$\mathtt{L}_k(\boldsymbol{\beta}^{(s)}) = \mathrm{A} \sum_{\substack{i=1 \\ \boldsymbol{\sigma}_j = f_s(\boldsymbol{\beta}^{(s)}, \boldsymbol{\sigma}_i)}}^{N_s} P[\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i, \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \tag{11.329}$$

where f_s is the state transition function, and the multiplicative constant $\mathrm{A} = 1/P[\mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}]$ is irrelevant for vector detection as can be seen from (11.320). We note that the summation in (11.329) is over all transitions from the general state $\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i$ to the state $\mathbf{s}_k = \boldsymbol{\sigma}_j = f_s(\boldsymbol{\beta}^{(s)}, \boldsymbol{\sigma}_i)$ generated by the information vector $\mathbf{b}_k = \boldsymbol{\beta}^{(s)}$. On the other hand, the probability in (11.329) can be

written as

$$\begin{aligned}
&P[\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i, \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \\
&\quad = P[\mathbf{z}_{k+1}^{K-1} = \boldsymbol{\rho}_{k+1}^{K-1} \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i, \mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{z}_0^{k} = \boldsymbol{\rho}_0^{k}] \\
&\qquad P[\mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{z}_k = \boldsymbol{\rho}_k \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i, \mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}] \\
&\qquad P[\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i, \mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}] \\
&\quad = P[\mathbf{z}_{k+1}^{K-1} = \boldsymbol{\rho}_{k+1}^{K-1} \mid \mathbf{s}_k = \boldsymbol{\sigma}_j] \\
&\qquad P[\mathbf{s}_k = \boldsymbol{\sigma}_j, \mathbf{z}_k = \boldsymbol{\rho}_k \mid \mathbf{s}_{k-1} = \boldsymbol{\sigma}_i]\ P[\mathbf{s}_{k-1} = \boldsymbol{\sigma}_i, \mathbf{z}_0^{k-1} = \boldsymbol{\rho}_0^{k-1}] \\
&\quad = B_k(j)\ C_k(j \mid i)\ F_{k-1}(i)
\end{aligned} \tag{11.330}$$

Substituting for $C_k(j \mid i)$ the expression in (11.324), the likelihood becomes

$$\mathtt{L}_k(\boldsymbol{\beta}^{(s)}) = \mathrm{B}\ \mathtt{L}_k^{(a)}(\boldsymbol{\beta}^{(s)})\ \mathtt{L}_k^{(int)}(\boldsymbol{\beta}^{(s)})\ \mathtt{L}_k^{(ext)}(\boldsymbol{\beta}^{(s)}) \tag{11.331}$$

where $\mathrm{B} = \mathrm{A}/(2\pi\sigma_I^2)$ is an irrelevant constant,

$$\mathtt{L}_k^{(int)}(\boldsymbol{\beta}^{(s)}) = e^{-\frac{1}{2\sigma_I^2}\|\boldsymbol{\rho}_k^{(s)} - \boldsymbol{\beta}^{(s)}\|^2} \tag{11.332}$$

and

$$\mathtt{L}_k^{(ext)}(\boldsymbol{\beta}^{(s)}) = \sum_{\substack{i=1 \\ \boldsymbol{\sigma}_j = f_s(\boldsymbol{\beta}^{(s)}, \boldsymbol{\sigma}_i)}}^{N_s} B_k(j)\ C_k^{(p)}(j \mid i)\ F_{k-1}(i) \tag{11.333}$$

Observing each term in (11.331), we make the following considerations.

i. $\mathtt{L}_k^{(a)}(\boldsymbol{\beta}^{(s)})$ represents the *a priori information* on the information vector $\mathbf{b}_k = \boldsymbol{\beta}^{(s)}$.

ii. $\mathtt{L}_k^{(int)}(\boldsymbol{\beta}^{(s)})$ depends on the received samples associated with the information vector and on the channel characteristics.

iii. $\mathtt{L}_k^{(ext)}(\boldsymbol{\beta}^{(s)})$ represents the *extrinsic information* extracted from the received samples associated with the parity check bits. This is the incremental information on the information vector obtained by the decoding process.

6. Typically it is easier to work with the logarithm of the various likelihoods.

We associate with each bit of the code vector $\mathbf{c}_k$ a log-likelihood ratio (LLR) that depends on the channel (see (11.285)), that is

$$\boldsymbol{\ell}_k^{(in)} = [\,\ell_k^{(in,1)}, \ldots, \ell_k^{(in,n_0)}\,] = [\,\boldsymbol{\ell}_k^{(in,s)}, \boldsymbol{\ell}_k^{(in,p)}\,] \tag{11.334}$$

For binary modulation, from (11.315), we get (see (11.287))

$$\boldsymbol{\ell}_k^{(in)} = \frac{2}{\sigma_I^2}\boldsymbol{\rho}_k \tag{11.335}$$

where $\boldsymbol{\rho}_k$ is the observation at the instant k.

We define now two quantities that are related, respectively, to the systematic bits and the parity check bits of the code vector, as

$$\ell_k^{(s)}(\boldsymbol{\beta}^{(s)}) = \frac{1}{2}\sum_{m=1}^{k_0} \ell_k^{(in,m)}\beta^{(m)} \tag{11.336}$$

and

$$\ell_k^{(p)}(j,i) = \frac{1}{2}\sum_{m=k_0+1}^{n_0} \ell_k^{(in,m)}\beta^{(m)} \tag{11.337}$$

where by (11.313) and (11.314) we have

$$\boldsymbol{\beta}^{(p)} = [\,\beta^{(k_0+1)}, \ldots, \beta^{(n_0)}\,] = f^{(p)}(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i) \tag{11.338}$$

Expressing (11.325) and (11.326) as a function of the likelihoods (11.336) and (11.337), apart from factors that do not depend on $\{\beta^{(m)}\}$, $m = 1, \ldots, n_0$, we get

$$C_k^{'(s)}(j \mid i) = e^{\ell_k^{(s)}(\boldsymbol{\beta}^{(s)})} e^{\ell_k^{(a)}(\boldsymbol{\beta}^{(s)})} \tag{11.339}$$

and

$$C_k^{'(p)}(j \mid i) = e^{\ell_k^{(p)}(j,i)} \tag{11.340}$$

To compute the forward and backward metrics, we use, respectively, (11.327) and (11.328), where the variable $C_k(j \mid i)$ is replaced by $C_k^{'}(j \mid i) = C_k^{'(s)}(j \mid i)\, C_k^{'(p)}(j \mid i)$. Similarly in (11.333) $C_k^{(p)}(j \mid i)$ is replaced by $C_k^{'(p)}(j \mid i)$. Taking the logarithm of (11.333) we obtain the extrinsic component $\ell_k^{(ext)}(\boldsymbol{\beta}^{(s)})$.

Finally, from (11.331), by ignoring non-essential terms, the log-likelihood associated with the information vector $\mathbf{b}_k = \boldsymbol{\beta}^{(s)}$ is given by

$$\ell_k(\boldsymbol{\beta}^{(s)}) = \ell_k^{(a)}(\boldsymbol{\beta}^{(s)}) + \ell_k^{(int)}(\boldsymbol{\beta}^{(s)}) + \ell_k^{(ext)}(\boldsymbol{\beta}^{(s)}) \tag{11.341}$$

where $\ell_k^{(int)}(\boldsymbol{\beta}^{(s)}) = \ell_k^{(s)}(\boldsymbol{\beta}^{(s)})$ is usually called the intrinsic component.

Expression (11.341) suggests an alternative method to (11.333) to obtain $\ell_k^{(ext)}(\boldsymbol{\beta}^{(s)})$, which uses the direct computation of $\ell_k(\boldsymbol{\beta}^{(s)})$ by (11.329) and (11.330), where $C_k(j \mid i)$ is replaced by $C_k^{'}(j \mid i)$, whose factors are given in (11.339) and

(11.340). From the known *a priori* information $\ell_k^{(a)}(\boldsymbol{\beta}^{(s)})$ and the intrinsic information (11.336), from (11.341) we get

$$\ell_k^{(ext)}(\boldsymbol{\beta}^{(s)}) = \ell_k(\boldsymbol{\beta}^{(s)}) - \ell_k^{(a)}(\boldsymbol{\beta}^{(s)}) - \ell_k^{(int)}(\boldsymbol{\beta}^{(s)}) \tag{11.342}$$

Going back to the expression (11.341), detection of the vector $\mathbf{b}_k$ is performed according to the rule

$$\hat{\mathbf{b}}_k = \arg\max_{\boldsymbol{\beta}^{(s)}} \ell_k(\boldsymbol{\beta}^{(s)}) \tag{11.343}$$

Note that to compute $\ell_k^{(ext)}(\boldsymbol{\beta}^{(s)})$ ($\ell_k(\boldsymbol{\beta}^{(s)})$) by the logarithm of (11.333) (or (11.329) and (11.330)), we can use the Max-Log-MAP method discussed in Section 8.10.2.

Example 11.5.1 (Systematic convolutional code with rate $1/2$*)*
For a convolutional code with rate $R_c = 1/2$ the information vector $\mathbf{b}_k = [b_k]$ is composed of only one bit ($k_0 = 1$), like the systematic part and the parity check part of $\mathbf{c}_k = [c_k^{(s)}, c_k^{(p)}]$.

In this case it is sufficient to determine the log-likelihoods $\ell_k(-1)$ and $\ell_k(1)$, or better the LLR

$$\ell_k = \ln \frac{\mathtt{L}_k(1)}{\mathtt{L}_k(-1)} = \ell_k(1) - \ell_k(-1) \tag{11.344}$$

Detection of the information bit is performed according to the rule

$$\hat{b}_k = \operatorname{sgn}(\ell_k) \tag{11.345}$$

The *a priori* information at the decoder input is given by

$$\ell_k^{(a)} = \ln \frac{P[b_k = 1]}{P[b_k = -1]} \tag{11.346}$$

from which we derive the *a priori* probabilities

$$P[b_k = \beta^{(s)}] = e^{\ell_k^{(a)}(\boldsymbol{\beta}^{(s)})} = \frac{e^{\beta^{(s)}\ell_k^{(a)}}}{1 + e^{\beta^{(s)}\ell_k^{(a)}}} = \frac{e^{-\frac{1}{2}\ell_k^{(a)}}}{1 + e^{-\ell_k^{(a)}}}\, e^{\frac{1}{2}\beta^{(s)}\ell_k^{(a)}} \qquad \beta^{(s)} = \{-1, 1\} \tag{11.347}$$

By using LLRs, (11.336) yields

$$\ell_k^{(int)} = \ell_k^{(int)}(1) - \ell_k^{(int)}(-1) = \ell_k^{(in,1)} = \ell_k^{(in,s)} \tag{11.348}$$

In turn (11.339) and (11.340) for $k_0 = 1$ and $n_0 = 2$ simplify into

$$C_k^{'(s)}(j \mid i) = e^{\frac{1}{2}\,\beta^{(s)}\,(\ell_k^{(in,s)} + \ell_k^{(a)})} \tag{11.349}$$

$$C_k^{'(p)}(j \mid i) = e^{\frac{1}{2}\,\beta^{(p)}\,\ell_k^{(in,p)}} \tag{11.350}$$

The extrinsic component is obtained starting from (11.333) and using the above variables

$$\ell_k^{(ext)} = \ln \frac{\mathtt{L}_k^{(ext)}(1)}{\mathtt{L}_k^{(ext)}(-1)} = \ell_k^{(ext)}(1) - \ell_k^{(ext)}(-1) \tag{11.351}$$

From (11.341), apart from irrelevant terms, the LLR associated with the information bit b_k can be written as

$$\ell_k = \ell_k^{(a)} + \ell_k^{(int)} + \ell_k^{(ext)} \tag{11.352}$$

where the meaning of each of the three contributions is as follows.

- *A priori information* $\ell_k^{(a)}$. It is an *a priori* reliability measure on the bit b_k. This value can be extracted either from the known statistic of the information sequence or, in the case of iterative decoding of turbo codes, from the previous analysis.
- *Channel information* $\ell_k^{(int)} = \ell_k^{(in,s)}$. As it is evident from the case of binary modulation, where $\ell_k^{(in,s)} = \frac{2}{\sigma_I^2}\rho_k^{(s)}$, if the noise variance is low, the contribution of $\ell_k^{(int)}$ usually dominates with respect to the other two terms; in this case bit detection is simply obtained by the sign of $\rho_k^{(s)}$.
- *Extrinsic information* $\ell_k^{(ext)}$. It is a reliability measure that is determined by the redundancy in the transmitted sequence. This contribution improves the reliability of transmission over a noisy channel using the parity check bits.

The decomposition (11.352) forms the basis for the iterative decoding of turbo codes.

Observation 11.5
In the case of multilevel modulation and/or for transmission over channels with ISI, the previous formulation of the decoding scheme remains unchanged, provided the expression (11.285) for $\{\ell_k^{(in,m)}\}$, $m = 1, \ldots, n_0$, is used in place of (11.335).

Example 11.5.2 (Nonsystematic code and LLR associated with the code bits)
Consider the case of a nonsystematic code. If the code is also non-recursive, for example as illustrated on page 915 for $k_0 = 1$, we need to use in place of (11.310) the state definition (11.273).

Now all bits are parity check bits and (11.312) and (11.316) become, respectively,

$$\mathbf{c}_k = \mathbf{c}_k^{(p)} = [\, c_k^{(1)}, \ldots, c_k^{(n_0)} \,] \tag{11.353}$$

$$\mathbf{z}_k = \mathbf{z}_k^{(p)} = [\, z_k^{(1)}, \ldots, z_k^{(n_0)} \,] \tag{11.354}$$

However, the information vector is still given by $\mathbf{b}_k = [\, b_k^{(1)}, \ldots, b_k^{(k_0)} \,]$ with values $\boldsymbol{\alpha} = [\, \alpha^{(1)}, \ldots, \alpha^{(k_0)} \,]$, $\alpha^{(i)} \in \{-1, 1\}$. The likelihood (11.319) is given by

$$\mathtt{L}_k(\boldsymbol{\alpha}) = P[\mathbf{b}_k = \boldsymbol{\alpha} \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}] \tag{11.355}$$

The various terms with superscript $^{(s)}$ of the previous analysis vanish by setting $k_0 = 0$. Therefore (11.336) and (11.337) become

$$\ell_k^{(int)}(\boldsymbol{\beta}^{(s)}) = \ell_k^{(s)}(\boldsymbol{\beta}^{(s)}) = 0 \tag{11.356}$$

and

$$\ell_k^{(p)}(j,i) = \frac{1}{2}\sum_{m=1}^{n_0} \ell_k^{(in,m)}\beta^{(m)} \tag{11.357}$$

where $\boldsymbol{\beta} = \boldsymbol{\beta}^{(p)} = [\,\beta^{(1)}, \ldots, \beta^{(n_0)}\,] = f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i)$ is the code vector associated with the transition from state $\boldsymbol{\sigma}_i$ to state $\boldsymbol{\sigma}_j$. Note that, apart from irrelevant factors, (11.357) coincides with (11.291).

For $k_0 = 1$, it is convenient to use LLRs; in particular (11.352) yields a LLR associated with the information bit b_k that is given by

$$\ell_k = \ell_k^{(a)} + \ell_k^{(ext)} \tag{11.358}$$

where $\ell_k^{(ext)}$ can be obtained directly using (11.351), (11.340), and (11.333).

In some applications it is useful to associate a LLR with the encoded bit $c_k^{(q)}, q = 1, \ldots, n_0$, rather than to the information bit b_k. We define

$$\bar{\ell}_{k,q} = \ln \frac{P[c_k^{(q)} = 1 \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}]}{P[c_k^{(q)} = -1 \mid \mathbf{z}_0^{K-1} = \boldsymbol{\rho}_0^{K-1}]} \tag{11.359}$$

Let $\bar{\ell}_{k,q}^{(a)}$ be the *a priori* information on the code bits. The analysis is similar to the previous case but now, with respect to the encoder output, $c_k^{(q)}$ is regarded as an information bit, while the remaining bits $c_k^{(m)}, m = 1, \ldots, n_0$, $m \neq q$, are regarded as parity check bits. Equations (11.336), (11.337), (11.349), and (11.350), are modified as follows:

$$\bar{\ell}_{k,q}^{(s)}(\boldsymbol{\beta}^{(q)}) = \frac{1}{2}\ell_k^{(in,q)}\beta^{(q)} \tag{11.360}$$

$$\bar{\ell}_{k,q}^{(p)}(j,i) = \frac{1}{2}\sum_{\substack{m=1 \\ m\neq q}}^{n_0} \ell_k^{(in,m)}\beta^{(m)} \tag{11.361}$$

and

$$C_{k,q}^{\prime(s)}(j \mid i) = e^{\frac{1}{2}\beta^{(q)}\,(\ell_k^{(in,q)} + \bar{\ell}_{k,q}^{(a)})} \tag{11.362}$$

$$C_{k,q}^{\prime(p)}(j \mid i) = e^{\bar{\ell}_{k,q}^{(p)}(j,i)} \tag{11.363}$$

Associated with (11.363) we obtain $\bar{\ell}_{k,q}^{(ext)}$ by using (11.351) and (11.333). The overall result is given by

$$\bar{\ell}_{k,q} = \bar{\ell}_{k,q}^{(a)} + \ell_k^{(in,q)} + \bar{\ell}_{k,q}^{(ext)} \qquad q = 1, \ldots, n_0 \tag{11.364}$$

Example 11.5.3 (Systematic code and LLR associated with the code bits)
With reference to the previous example, if the code is systematic, whereas (11.352) holds for the systematic bit $c_k^{(1)}$, for the parity check bits $c_k^{(q)}$ the following relations hold [37]. For $k_0 = 1$ let α be the value of the information bit b_k, $b_k = \alpha$, with $\alpha \in \{-1, 1\}$, associated with the code vector

$$\mathbf{c}_k = \boldsymbol{\beta} = [\alpha, \beta^{(2)}, \ldots, \beta^{(n_0)}] \tag{11.365}$$

where we assume $\beta^{(1)} = \alpha$. For $q = 2, \ldots, n_0$, we get

$$\bar{\ell}_{k,q}^{(s)}(\beta^{(q)}) = \frac{1}{2}\ell_k^{(in,q)}\beta^{(q)} \tag{11.366}$$

$$\bar{\ell}_{k,q}^{(p)}(j,i) = \frac{1}{2}\sum_{\substack{m=1 \\ m \neq q}}^{n_0} \ell_k^{(in,m)}\beta^{(m)} + \frac{1}{2}\ell_k^{(a)}\alpha \tag{11.367}$$

where $\ell_k^{(a)}$ is the *a priori* information of b_k. Furthermore

$$C_{k,q}^{'(s)}(j \mid i) = e^{\frac{1}{2}\beta^{(q)}\ell_k^{(in,q)}} \tag{11.368}$$

$$C_{k,q}^{'(p)}(j \mid i) = e^{\bar{\ell}_{k,q}^{(p)}(j,i)} \tag{11.369}$$

From (11.369) we get $\bar{\ell}_{k,q}^{(ext)}$ using (11.351) and (11.333). The overall result is given by

$$\bar{\ell}_{k,q} = \ell_k^{(in,q)} + \bar{\ell}_{k,q}^{(ext)} \tag{11.370}$$

Iterative decoding

In this section we consider the iterative decoding of turbo codes with $k_0 = 1$. In this case, as seen in Example 11.5.1, using the LLRs simplifies the procedure. In general, for $k_0 > 1$ we should refer to the formulation (11.341).

We now give a step-by-step description of the decoding procedure of a turbo code with rate 1/3, of the type shown in Figure 11.27, where each of the two component decoders DEC_1 and DEC_2 implements the FBA for recursive systematic convolutional codes with rate 1/2. The decoder scheme is shown in Figure 11.30 where the subscript in LLR corresponds to the component decoder. In correspondence to the information bit b_k, the turbo code generates the vector

$$\mathbf{c}_k = [c_k^{(1)}, c_k^{(2)}, c_k^{(3)}] \tag{11.371}$$

where $c_k^{(1)} = b_k$. We now introduce the following notation for the observation vector $\boldsymbol{\ell}_k^{(in)}$ that relates to the considered decoder:

$$\boldsymbol{\ell}_k^{(in)} = [\ell_k^{(in,s)}, \ell_{k,1}^{(in,p)}, \ell_{k,2}^{(in,p)}] \tag{11.372}$$

where $\ell_k^{(in,s)}$ corresponds to the systematic part, and $\ell_{k,1}^{(in,p)}$ and $\ell_{k,2}^{(in,p)}$ correspond to the parity check parts generated by the first and second convolutional encoder, respectively.

If some parity check bits are punctured to increase the rate of the code, at the receiver the corresponding LLRs $\ell_k^{(in,m)}$ are set to zero.

1. *First iteration*

   1.1 *Decoder DEC_1*
   If the statistic of the information bits is unknown, then the bits of the information sequence are considered i.i.d. and the *a priori* information is zero,

$$\ell_{k,1}^{(a)} = \ln \frac{P[b_k = 1]}{P[b_k = -1]} = 0 \tag{11.373}$$

   For $k = 0, 1, 2, \ldots, K-1$, observed $\ell_k^{(in,s)}$ and $\ell_{k,1}^{(in,p)}$, we compute according to (11.349) and (11.350) the variables $C_k^{'(s)}$ and $C_k^{'(p)}$, and from these the corresponding forward metric $F_k(j)$ (11.327). After the entire sequence has been received, we compute the backward metric $B_k(i)$ (11.328) and, using (11.333), we find $\mathtt{L}_{k,1}^{(ext)}(1)$ and $\mathtt{L}_{k,1}^{(ext)}(-1)$. The decoder soft output is the extrinsic information obtained by the LLR

$$\ell_{k,1}^{(ext)} = \ln \frac{\mathtt{L}_{k,1}^{(ext)}(1)}{\mathtt{L}_{k,1}^{(ext)}(-1)} \tag{11.374}$$

   1.2 *Interleaver*
   Because of the presence of the interleaver, the parity check bit $c_n^{(3)}$ is obtained in correspondence to a transition of the convolutional encoder state determined by the information bit b_n, where n depends on the interleaver pattern. In decoding, the extrinsic information $\ell_{k,1}^{(ext)}$, extracted from DEC_1, and the systematic observation $\ell_k^{(in,s)}$ are scrambled by the turbo code interleaver and associated with the corresponding observation $\ell_{n,2}^{(in,p)}$ to form the input of the second component decoder.

   1.3 *Decoder DEC_2*
   The extrinsic information generated by DEC_1 is set as the *a priori* information $\ell_{n,2}^{(a)}$ to the component decoder DEC_2,

$$\ell_{n,2}^{(a)} = \ln \frac{P[b_n = 1]}{P[b_n = -1]} = \ell_{n,1}^{(ext)} \tag{11.375}$$

   The basic idea consists in supplying DEC_2 only with the extrinsic part $\ell_{n,1}^{(ext)}$ of $\ell_{n,1}$, in order to minimize the correlation between the *a priori* information and the observations used by DEC_2. Ideally, the *a priori* information should be an independent estimate.

   As done for DEC_1, we extract the extrinsic information $\ell_{n,2}^{(ext)}$.

1.4 *Deinterleaver* The deinterleaver realizes the inverse function of the interleaver, so that the extrinsic information extracted from DEC_2, $\ell_{n,2}^{(ext)}$, is synchronized with the systematic part $\ell_k^{(in,s)}$ and the parity check part $\ell_{k,1}^{(in,p)}$ of the observation of DEC_1. By a feedback loop the *a posteriori* information $\ell_{k,2}^{(ext)}$ is placed at the input of DEC_1 as *a priori* information $\ell_{k,1}^{(a)}$, giving origin to an iterative structure.

2. *Successive iterations*
Starting from the second iteration each component decoder has at its input an *a priori* information. The information on the bits become more reliable as the *a priori* information stabilizes in sign and increases in amplitude.

3. *Last iteration*
When the decoder achieves convergence, the iterative process can stop and form the overall LLR (11.352),

$$\ell_{k,overall} = \ell_k^{(in,s)} + \ell_{k,1}^{(ext)} + \ell_{k,2}^{(ext)} \qquad k = 0, 1, \ldots, K-1 \tag{11.376}$$

and detection of the information bits b_k is obtained by

$$\hat{b}_k = \text{sgn}(\ell_{k,overall}) \tag{11.377}$$

To make decoding more reliable, the final state of each component decoder is set to zero, thus enabling an initialization of the backward metric as in (8.244). As illustrated in Figure 11.31, at the instant following the input of the last information bit, that is for $k = K$, the commutator is switched to the lower position, and therefore we have $d_k = 0$; after ν clock intervals the zero state is reached. The bits $c_k^{(1)}$ and $c_k^{(2)}$, for $k = K, K+1, \ldots, K+\nu-1$, are appended at the end of the code sequence to be transmitted.

Performance evaluation

Performance of the turbo code with the encoder of Figure 11.27 is evaluated in terms of error probability and convergence of the iterative decoder implemented by the FBA. For the memoryless AWGN channel, error probability curves versus E_b/N_0 are plotted in Figure 11.32 for a sequence of information bits of length $K = 640$, and various numbers of

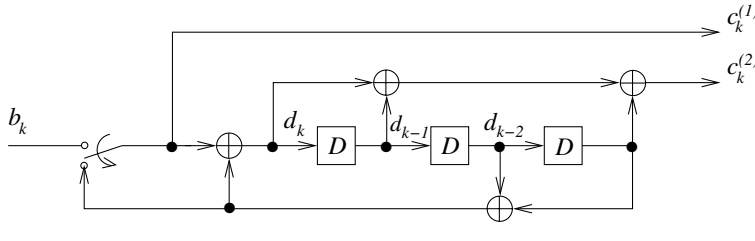


Figure 11.31. Termination of trellis.

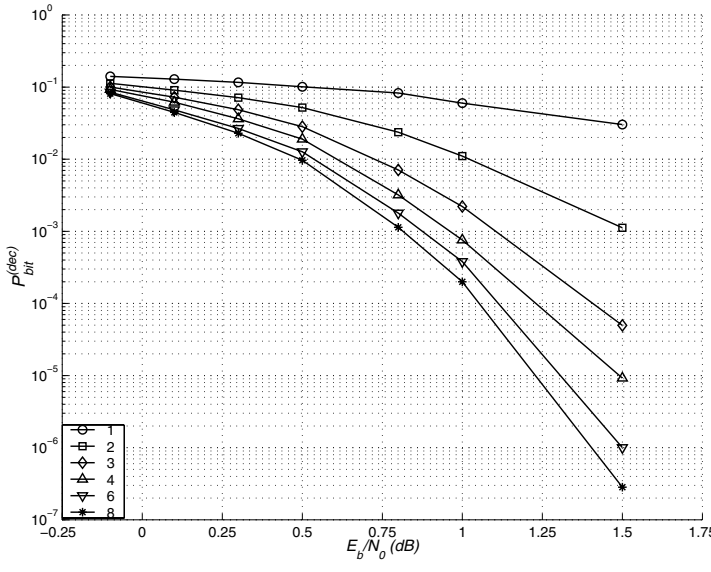


Figure 11.32. Performance of the turbo code defined by the UMTS standard, with length of the information sequence $K = 640$, and various numbers of iterations of the iterative decoding process.

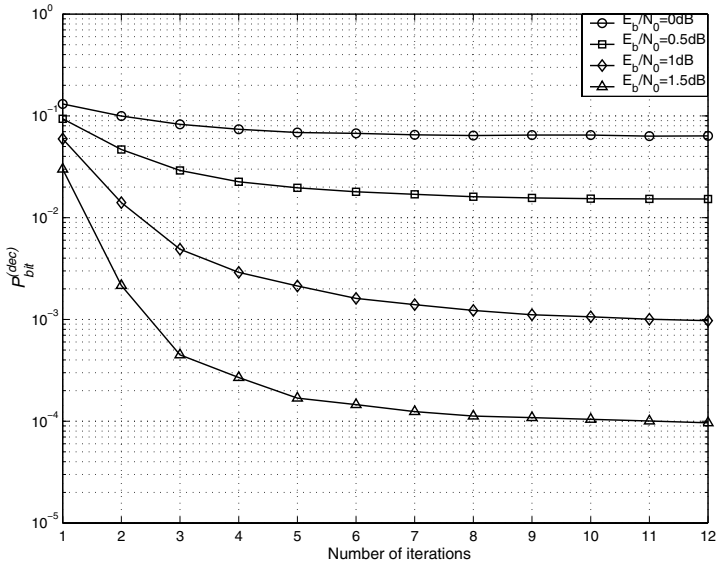


Figure 11.33. Curves of convergence of the decoder for the turbo code defined by the UMTS standard, for $K = 320$ and various values of E_b/N_0.

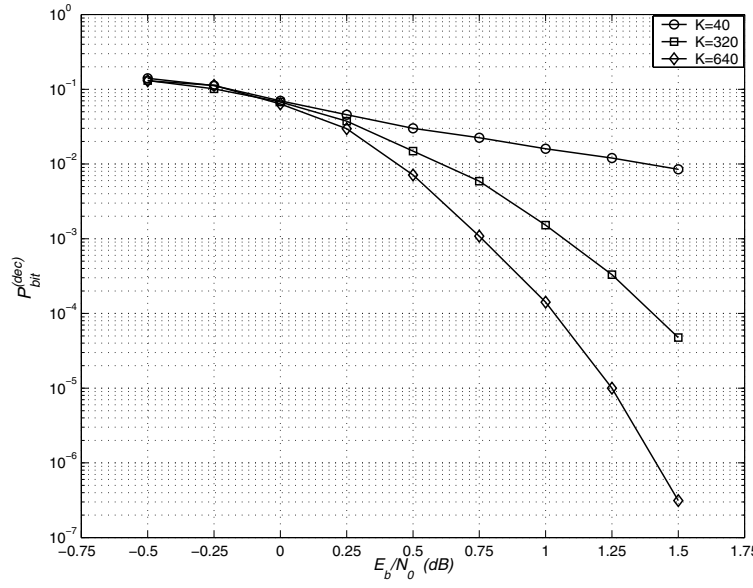


Figure 11.34. Performance of the turbo code defined by the UMTS standard achieved after 12 iterations, for $K = 40$, 320 and 640.

iterations of the iterative decoding process. Note that performance improves as the number of iterations increases; however, the gain between consecutive iterations becomes smaller as the number of iterations increases.

In Figure 11.33, the error probability $P_{bit}^{(dec)}$ is given as a function of the number of iterations, for fixed values of E_b/N_0, and $K = 320$. From the behavior of the error probability we deduce possible criteria for stopping the iterative decoding process at convergence [36]. A timely stop of the iterative decoding process leads to a reduction of the decoding delay and of the overall computational complexity of the system. Note, however, that convergence is not always guaranteed.

The performance of the code depends on the length K of the information sequence. Figure 11.34 illustrates how the bit error probability decreases by increasing K, for a constant E_b/N_0. A higher value of K corresponds to an interleaver on longer sequences and thus the assumption of independence among the inputs of each component decoder is better satisfied. Moreover, the burst errors introduced by the channel are distributed over all the original sequence, increasing the correction capability of the decoder. As the length of the interleaver grows, also the latency of the system increases.

11.6 Iterative detection and decoding

We consider the transmitter of Figure 11.35 composed of a convolutional encoder, interleaver, bit mapper and modulator for 16-PAM. Interpreting the channel as a finite-state machine, the overall structure may be interpreted as a *serial concatenated convolutional*

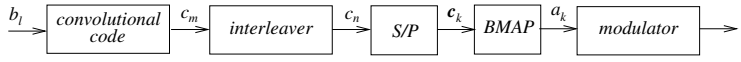


Figure 11.35. Encoder structure, bit mapper, and modulator; for 16-PAM: $\mathbf{c}_k = [c_{4k}, c_{4k-1}, c_{4k-2}, c_{4k-3}]$.

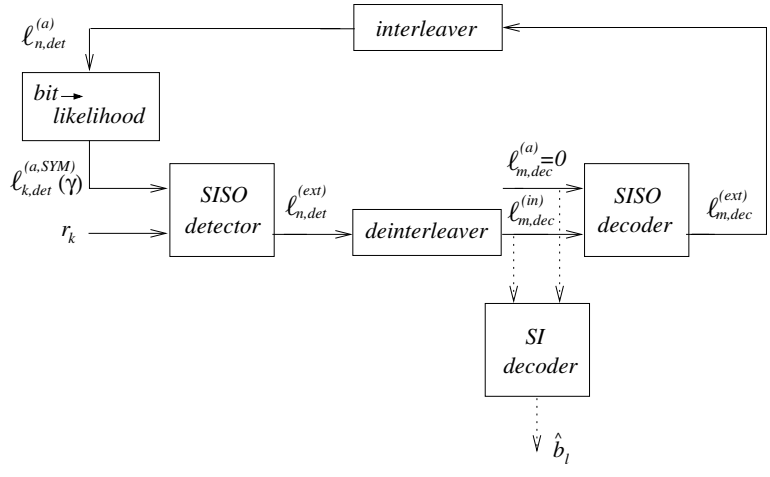


Figure 11.36. Iterative detection and decoding.

code (SCCC). The procedure of SISO detection and SI decoding of page 916 can be made iterative by applying the principles of the previous section, by including a SISO decoding stage. With reference to Figure 11.36, a step by step description follows.

0. *Initialization*
   Suppose we have no information on the *a priori* probability of the code bits, therefore we associate with c_n a zero LLR,

$$\ell^{(a)}_{n,det} = 0 \tag{11.378}$$

1. *Detector*
   First we associate a log-likelihood with the two possible values of $c_n = \eta$, $\eta \in \{-1, 1\}$, according to the rule

$$\ell^{(a)}_{n,det}(1) = \ell^{(a)}_{n,det} \qquad \ell^{(a)}_{n,det}(-1) = 0 \tag{11.379}$$

   Then we express the symbol a_k as a function of the bits $\{c_n\}$ according to the bit mapper, for example, for 16-PAM,

$$a_k = BMAP\{\mathbf{c}_k = [c_{4k}, c_{4k-1}, c_{4k-2}, c_{4k-3}]\} \tag{11.380}$$

Assuming the sequence $\{c_n\}$ is a sequence of i.i.d. binary symbols, we associate with each value of the symbol a_k the *a priori* information expressed by the log-likelihood

$$\ell_{k,det}^{(a,SYM)}(\gamma) = \sum_{t=0}^{3} \ell_{4k-t,det}^{(a)}(\eta_t) \qquad \gamma \in \mathcal{A} \tag{11.381}$$

where $\gamma = BMAP\{[\eta_0, \ldots, \eta_3]\}\eta_t \in \{-1, 1\}$.

For multilevel transmission over a channel with ISI, the FBA of Section 8.10.2 provides a log-likelihood for each value of a_k. The new feature is that now in (8.224) we take into account the *a priori* information on the various values of a_{k+L_1}; then (8.229) becomes

$$C_k(j \mid i) = \frac{1}{\sqrt{2\pi\sigma_I^2}} e^{-\frac{1}{2\sigma_I^2}|\rho_k - u_k|^2} e^{\ell_{k+L_1,det}^{(a,SYM)}(\gamma)} \tag{11.382}$$

where $\gamma = f(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}_i) \in \mathcal{A}$ is the symbol associated with the transition from the state $\boldsymbol{\sigma}_i$ to the state $\boldsymbol{\sigma}_j$ on the trellis determined by the ISI. If $V_k(i), i = 1, \ldots, N_s$, denotes the metric corresponding to the various states of the trellis, we associate with each value of the code bits $\{c_n\}$ the following likelihood:

$$\mathtt{L}_{4(k+L_1)-t,det}(\alpha) = \sum_{\substack{i=1 \\ \boldsymbol{\sigma}_i \text{ such that} \\ c_{4(k+L_1)-t} = \alpha}}^{N_s} V_k(i) \qquad \alpha \in \{-1, 1\} \qquad t = 0, \ldots, 3 \tag{11.383}$$

Taking the logarithm of (11.383), we obtain the LLR

$$\ell_{n,det} = \ell_{n,det}(1) - \ell_{n,det}(-1) \tag{11.384}$$

To determine the extrinsic information associated with $\{c_n\}$, we subtract the *a priori* information from (11.384),

$$\ell_{n,det}^{(ext)} = \ell_{n,det} - \ell_{n,det}^{(a)} \tag{11.385}$$

Note that in this application, the detector considers the bits $\{c_n\}$ as information bits and the log-likelihood associated with c_n at the detector output is due to the channel information[9] in addition to the *a priori* information.

In [38], the quantity $\ell_{n,det}^{(a)}$ in (11.385) is weighted by a coefficient, which is initially chosen small, when the *a priori* information is not reliable, and is increased after each iteration.

2. *Deinterleaver*
The metrics $\ell_{n,det}^{(ext)}$ are re-ordered according to the deinterleaver to provide the sequence $\ell_{m,det}^{(ext)}$.

[9] For the iterative decoding of turbo codes, this information is defined as intrinsic.

3. *Decoder (SISO)*
   As input LLR, we use

$$\ell_{m,dec}^{(in)} = \ell_{m,det}^{(ext)} \tag{11.386}$$

   and we set

$$\ell_{m,dec}^{(a)} = 0 \tag{11.387}$$

   in the lack of an *a priori* information on the code bits $\{c_m\}$. Indeed, we note that in the various formulae the roles of $\ell_{m,dec}^{(a)}$ and $\ell_{m,dec}^{(in)}$ can be interchanged.

   Depending on whether the code is systematic or not, we use the SISO decoding procedure reported in Example 11.5.1 and Example 11.5.2, respectively. In both cases we associate with the encoded bits c_m the quantity

$$\ell_{m,dec}^{(ext)} = \ell_{m,dec} - \ell_{m,dec}^{(in)} \tag{11.388}$$

   that is passed to the SISO detector as *a priori* information, after reordering by the interleaver.

4. *Last iteration*
   After a suitable number of iterations, the various metrics stabilize and from the LLRs $\{\ell_{m,dec}^{(in)}\}$ associated with $\{c_m\}$, the SI decoding of bits $\{b_l\}$ is performed, using the procedure of Example 11.5.1.

11.7 Low-density parity check codes

Low-density parity check (LDPC) codes were introduced by Gallager [6] as a family of linear block codes with parity check matrices containing mostly zeros and only a small number of ones. The "sparsity" of the parity check matrices defining LDPC codes is the key for the efficient decoding of these codes by a message-passing procedure also known as the "sum-product algorithm". LDPC codes and their efficient decoding were "reinvented" by MacKay and Neal [39, 40] in the mid-1990s, shortly after Berrou and Glavieux introduced the turbo-codes discussed in Section 11.5. Subsequently, LDPC codes have generated interest from a theoretical as well as from a practical viewpoint and many new developments have taken place.

It is today well acknowledged that LDPC codes are as good as turbo codes, as they are based on a similar design philosophy. Also the decoding techniques used for both methods can be viewed as different realizations of the same fundamental decoding process. However, the soft input soft output forward backward algorithm of Section 11.5, or suboptimal versions of it, used for turbo decoding is rather complex, whereas the sum-product algorithm used for LDPC decoding lends itself to parallel implementation and is computationally simpler. LDPC codes, on the other hand, may lead to more stringent requirements in terms of storage.

Recall that a linear (n_0, k_0) block code, where n_0 and k_0 denote the transmitted block length and the source block length, respectively, can be described in terms of a parity

check matrix $\mathbf{H}$, such that the equation $\mathbf{Hc} = \mathbf{0}$ is satisfied for all code words $\mathbf{c}$ (see (11.20)). Each row of the $r_0 \times n_0$ parity check matrix, where $r_0 = n_0 - k_0$ is the number of parity check bits, defines a parity check equation that is satisfied by each code word $\mathbf{c}$. For example, the (7,4) Hamming code is defined by the following parity check equations

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{bmatrix} = \mathbf{0} \longleftrightarrow \begin{array}{lll} c_5 = c_1 \oplus c_2 \oplus c_3 & \text{(check 1)} \\ c_6 = c_1 \oplus c_2 \oplus c_4 & \text{(check 2)} \\ c_7 = c_1 \oplus c_3 \oplus c_4 & \text{(check 3)} \end{array} \tag{11.389}$$

LDPC codes differ in major ways with respect to the above simple example; they usually have long block lengths n_0 in order to achieve near Shannon-limit performance, their parity check matrices are defined in nonsystematic form and exhibit a number of ones that is much less than $r_0 \cdot n_0$. A parity check matrix for a (J, K)-*regular LDPC code* has exactly J ones in each of its columns and K ones in each of its rows.

A parity check matrix can generally be represented by a bipartite graph, also called a Tanner graph, with two types of nodes: the *bit nodes* and the *parity check nodes* (or check nodes) [41]. A bit node n, representing the code bit c_n, is connected to the check node m only if the element (m, n) of the parity check matrix is equal to 1. No bit (check) node is connected to a bit (check) node. For example, the (7,4) Hamming code can be represented by the graph shown in Figure 11.37.

We note in this specific case that, because the parity check matrix is given in systematic form, bit nodes c_5, c_6, and c_7 in the associated graph are connected to single distinct check nodes. The parity check matrix of a (J, K)-regular LDPC code leads to a graph where every bit node is connected to precisely J check nodes and every check node is connected to precisely K bit nodes. We emphasize that the performance of an LDPC code depends on the random realization of the parity check matrix $\mathbf{H}$. Hence these codes form a constrained random code ensemble.

Graphical representations of LDPC codes are useful for deriving and implementing the iterative decoding procedure introduced in [6]. Gallager decoder is a message-passing

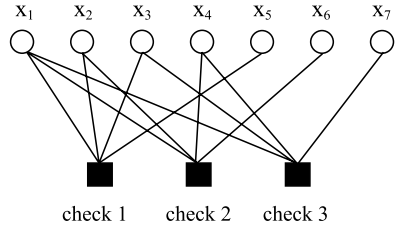


Figure 11.37. Tanner graph corresponding to the parity check matrix of the (7,4) Hamming code.

decoder, in a sense to be made clear below, based on the so-called sum-product algorithm, which is a general decoding algorithm for codes defined on graphs.[10]

Encoding procedure

Encoding is performed by multiplying the vector of k_0 information bits $\mathbf{b}$ by the generator matrix $\mathbf{G}$ of the LDPC code:

$$\mathbf{c^T} = \mathbf{b^T G} \tag{11.390}$$

where the operations are in GF(2). Recall that generator and parity check matrices satisfy the relation

$$\mathbf{H G^T} = \mathbf{0} \tag{11.391}$$

From (11.27), the parity check matrix in systematic form is $\tilde{\mathbf{H}} = [\tilde{\mathbf{A}}, \mathbf{I}]$, where $\mathbf{I}$ is the $r_0 \times r_0$ identity matrix, and $\tilde{\mathbf{A}}$ is a binary matrix. Recall also that any other $r_0 \times n_0$ matrix $\mathbf{H}$ whose rows span the same space as $\tilde{\mathbf{H}}$ is a valid parity check matrix.

Given the block length n_0 of the transmitted sequence and the block length k_0 of the information sequence, we select a column weight J, greater than or equal to 3. To define the code, we generate a rectangular $r_0 \times n_0$ matrix $\mathbf{H} = [\mathbf{A}\ \mathbf{B}]$ at random with exactly J ones per column and, assuming a proper choice of n_0 and k_0, exactly K ones per row. The $r_0 \times k_0$ matrix $\mathbf{A}$ and the square $r_0 \times r_0$ matrix $\mathbf{B}$ are very sparse. If the rows of $\mathbf{H}$ are independent, which is usually true with high probability if J is odd [40], by Gaussian elimination and reordering of columns we determine an equivalent parity check matrix $\tilde{\mathbf{H}}$ in systematic form. From (11.26), we obtain the generator matrix in systematic form as

$$\mathbf{G^T} = \begin{bmatrix} \mathbf{I} \\ \tilde{\mathbf{A}} \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ \mathbf{B^{-1}A} \end{bmatrix} \tag{11.392}$$

where I is the $k_0 \times k_0$ identity matrix.

Assuming initially antipodal linear signaling over an ideal AWGN channel, for the vector of transmitted symbols $\boldsymbol{a} = [a_1, \ldots, a_{n_0}]^T$, $a_k \in \{-1, 1\}$, corresponding to the code word $\boldsymbol{c} = [c_1, \ldots, c_{n_0}]^T$, the received vector is given by

$$\mathbf{z} = \mathbf{a} + \mathbf{w} \tag{11.393}$$

where $\mathbf{w}$ denotes a vector of Gaussian noise samples with variance σ_I^2.

Decoding algorithm

Adopting the MAP criterion (8.221), the optimal decoder returns the components of the vector $\hat{\mathbf{b}} = [\hat{b}_1, \ldots, \hat{b}_{k_0}]$ that maximize the *a posteriori* probabilities

$$\hat{b}_k = \arg \max_{\beta \in \{0,1\}} P[b_k = \beta \mid \mathbf{z} = \boldsymbol{\rho}, \mathbf{G}] \qquad k = 1, \ldots, k_0 \tag{11.394}$$

[10] A wide variety of other algorithms (e.g., the Viterbi algorithm, the forward backward algorithm, the iterative turbo decoding algorithm, the fast Fourier transform, ...) can also be derived as specific instances of the sum-product algorithm [42].

Note that (11.394) is equivalent to the MAP criterion expressed by (11.321). However, an attempt to evaluate (11.394) by the direct computation of the joint probability distribution of the components of **b** given the observation would be impractical. Assuming the probability of **b** uniform, and **w** statistically independent of **b**, we resort to the knowledge of the parity check matrix to simplify the decoding problem. We will find the most likely binary vector **x** such that (see (11.20))

$$\mathbf{s} = \mathbf{H}\mathbf{x} = \mathbf{0} \tag{11.395}$$

given the received noisy vector **z** and a valid parity check matrix **H**.

We call *checks* the elements $s_i, i = 1, \ldots, r_0$, of the vector **s**, which are represented by the check nodes in the corresponding Tanner graph. Then the aim is to compute the marginal *a posteriori* probabilities

$$\mathtt{L}_n(\beta) = P[x_n = \beta \mid \mathbf{z} = \boldsymbol{\rho}, \mathbf{s} = \mathbf{0}, \mathbf{G}] \qquad \beta \in \{0, 1\} \qquad n = 1, \ldots, n_0 \tag{11.396}$$

The detected code bits will then be given by

$$\hat{c}_n = \arg \max_{\beta \in \{0,1\}} \mathtt{L}_n(\beta) \qquad n = 1, \ldots, n_0 \tag{11.397}$$

We define as $H_{i,n}$ the element with indices (i, n) of the parity check matrix **H**. Let $L(i) = \{n : H_{i,n} = 1\}$, $i = 1, \ldots, r_0$, be the set of the bit nodes that participate in the check with index i. Also, let $L(i)\backslash\tilde{n}$ be the set $L(i)$ from which the element with index $\tilde{n}$ has been removed. Similarly, let $M(n) = \{i : H_{i,n} = 1\}$, $n = 1, \ldots, n_0$, be the set of the check nodes in which the bit with index n participates.

The algorithm consists of two alternating steps, illustrated in Figure 11.38, in which quantities $q_{i,n}^{\beta}$ and $r_{i,n}^{\beta}$, associated with each non-zero element of the matrix **H**, are iteratively updated. The quantity $q_{i,n}^{\beta}$ denotes the probability that $x_n = \beta$, $\beta \in \{0, 1\}$, given the information obtained via checks other than check i:

$$q_{i,n}^{\beta} = P[x_n = \beta \mid \{s_{i'} = 0, i' \in M(n)\backslash i\}, \mathbf{z} = \boldsymbol{\rho}] \tag{11.398}$$

Moreover, we define the *a posteriori* probabilities

$$q_n^{\beta} = P[x_n = \beta \mid s = 0, \mathbf{z} = \boldsymbol{\rho}] \tag{11.399}$$

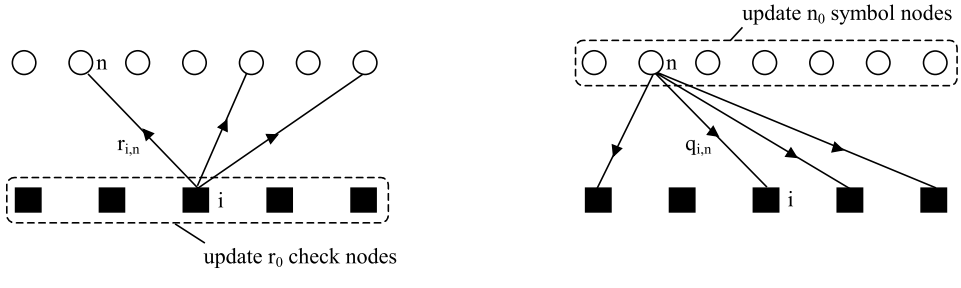


Figure 11.38. Message-passing decoding.

Given $x_n = \beta,\ \beta \in \{0, 1\}$, the quantity $r_{i,n}^{\beta}$ denotes the probability of check i being satisfied and the other bits having a known distribution (given by the probabilities $\{q_{i,n'}^{\beta} : n' \in L(i)\backslash n,\ \beta \in \{0, 1\}\}$):

$$r_{i,n}^{\beta} = P[s_i = 0, \{x_{n'}, n' \in L(i)\backslash n\} \mid x_n = \beta, \mathbf{z} = \boldsymbol{\rho}] \tag{11.400}$$

In the first step, the quantities $r_{i,n}^{\beta}$ associated with check node i are updated and passed as *messages* to the bit nodes checked by check node i. This operation is performed for all check nodes. In the second step, quantities $q_{i,n}^{\beta}$ associated with bit node n are updated and passed as messages to the check nodes that involve bit node n. This operation is performed for all bit nodes.

From (11.395), we note the property of (11.400) that

$$\begin{aligned} r_{i,n}^{0} &= 1 - P[s_i = 1, \{x_{n'}, n' \in L(i)\backslash n\} \mid x_n = 0, \mathbf{z} = \boldsymbol{\rho}] \\ &= 1 - P[s_i = 0, \{x_{n'}, n' \in L(i)\backslash n\} \mid x_n = 1, \mathbf{z} = \boldsymbol{\rho}] \\ &= 1 - r_{i,n}^{1} \end{aligned} \tag{11.401}$$

The algorithm is described as follows.

Initialization. Let $p_n^0 = P[x_n = 0 \mid \mathbf{z} = \boldsymbol{\rho}]$ denote the probability that $x_n = 0$ given the observation, and $p_n^1 = P[x_n = 1 \mid \mathbf{z} = \boldsymbol{\rho}] = 1 - p_n^0$. For the AWGN channel with binary antipodal input symbols considered in this section, we have (see (8.262))

$$p_n^0 = \frac{1}{1 + e^{2\rho_n/\sigma_I^2}} \qquad p_n^1 = \frac{1}{1 + e^{-2\rho_n/\sigma_I^2}} \tag{11.402}$$

Let $q_{i,n}^{\beta} = p_n^{\beta},\ n \in L(i),\ i = 1, \ldots, r_0,\ \beta \in \{0, 1\}$.

First step. We run through the checks, and for the i-th check we compute for each $n \in L(i)$ the probability $r_{i,n}^0$ that, given $x_n = 0$, $s_i = 0$ and the other bits $\{x_{n'} : n' \neq n\}$ have a distribution $\{q_{i,n'}^0, q_{i,n'}^1\}$.

From (11.400) we obtain

$$r_{i,n}^{0} = \sum_{\substack{\alpha_{n'} \in \{0,1\}: \\ n' \in L(i)\backslash n}} P[s_i = 0 \mid x_n = 0,\ \{x_{n'} = \alpha_{n'} : n' \in L(i)\backslash n\}] \prod_{n' \in L(i)\backslash n} q_{i,n'}^{\alpha_{n'}} \tag{11.403}$$

Moreover, $r_{i,n}^1 = 1 - r_{i,n}^0$.

The conditional probabilities in the above expression are either one or zero, depending on whether $s_i = 0$ or $s_i = 1$ is obtained for the hypothesized values of x_n and $\{x_{n'}\}$. The probabilities $\{r_{i,n}^0,\ r_{i,n}^1\}$ can be found efficiently by the FBA, as illustrated by the following example.

Example 11.7.1
Assume $K = 4$ and $L(i) = \{n_1, n_2, n_3, n_4\}$. The *observation* s_i can be expressed in terms of the input *variables* x_k, $k \in L(i)$, as

$$s_i = x_{n_1} + x_{n_2} + x_{n_3} + x_{n_4} = \sum_{l=1}^{K} x_{n_l} \tag{11.404}$$

where the addition is in GF(2). Let us define the state as

$$\boldsymbol{s}_{n_k} = \sum_{l=1}^{k} x_{n_l} = s_{n_{k-1}} + x_{n_k} \tag{11.405}$$

with $\boldsymbol{s}_0 = 0$, and observe that $\boldsymbol{s}_{n_K} = s_i$.

Following the formulation of the FBA in Section 8.10.2 we define the quantities:

1. Forward metric:

$$F_{n_k}(j) = P[\boldsymbol{s}_{n_k} = j] \qquad j \in \{0, 1\} \tag{11.406}$$

From (11.405) we obtain the recursive equation

$$F_{n_k}(j) = P[x_{n_k} = j]F_{n_{k-1}}(\overline{j}) + P[x_{n_k} = \overline{j}]F_{n_{k-1}}(j) \qquad k = 1, \ldots K \tag{11.407}$$

where $\overline{j}$ denotes the one's complement of j, with the initial condition $F_{n_0}(0) = 1$.

2. Backward metric:

$$\begin{aligned} B_{n_k}(j) &= P[s_i = 0 \mid \mathbf{s}_{n_k} = j] \\ &= \sum_{m=0}^{1} P[s_i = 0 \mid \mathbf{s}_{n_{k+1}} = m, \mathbf{s}_{n_k} = j]P[\mathbf{s}_{n_{k+1}} = m \mid \mathbf{s}_{n_k} = j] \\ &= \sum_{m=0}^{1} P[s_i = 0 \mid \mathbf{s}_{n_{k+1}} = m]P[x_{n_{k+1}} = m \oplus j] \qquad j \in \{0, 1\} \end{aligned} \tag{11.408}$$

using (11.405) and the fact that s_i is independent of $\mathbf{s}_{n_k}$ given $\mathbf{s}_{n_{k+1}}$. From (11.408), we obtain the recursive equation

$$B_{n_k}(j) = P[x_{n_{k+1}} = j]B_{n_{k+1}}(0) + P[x_{n_{k+1}} = \overline{j}]B_{n_{k+1}}(1) \qquad k = 1, \ldots K \tag{11.409}$$

with the initial condition $B_{n_{K+1}}(0) = 1$, which is obtained from the observation $s_i = 0$.

Therefore the probabilities r^{β}_{i,n_k}, $\beta \in \{0, 1\}$ are given by (see (8.244))

$$r^0_{i,n_k} = F_{n_k}(0)B_{n_k}(0) + F_{n_k}(1)B_{n_k}(1) \qquad k = 1, \ldots, K \tag{11.410}$$

and $r^1_{i,n_k} = 1 - r^0_{i,n_k}$.

Second step. After computing $r_{i,n}^0$ and $r_{i,n}^1$ we update the values of the probabilities $q_{i,n}^0$ and $q_{i,n}^1$. From (11.398) we find

$$q_{i,n}^{\beta} = \frac{P[x_n = \beta, \{s_{i'} = 0, i' \in M(n)\backslash i\}, \mathbf{z} = \boldsymbol{\rho}]}{P[\{s_{i'} = 0, i' \in M(n)\backslash i\}, \mathbf{z} = \boldsymbol{\rho}]} \tag{11.411}$$

Lumping in $\alpha_{i,n}$ the contribution of the terms that do not depend on β and using the i.i.d. assumption on the code bits, we obtain

$$\begin{aligned} q_{i,n}^{\beta} &= \alpha_{i,n} P[z_n = \rho_n, \{s_{i'} = 0, i' \in M(n)\backslash i\} \mid x_n = \beta] \\ &= \alpha_{i,n} p_n^{\beta} \prod_{i' \in M(n)\backslash i} r_{i',n}^{\beta} \end{aligned} \tag{11.412}$$

where $\alpha_{i,n}$ is chosen such that $q_{i,n}^0 + q_{i,n}^1 = 1$. Taking into account the information from all check nodes, from (11.399) we can also compute the "pseudo *a posteriori* probabilities" q_n^0 and q_n^1 at this iteration, given by

$$q_n^0 = \alpha_n p_n^0 \prod_{i \in M(n)} r_{i,n}^0 \tag{11.413}$$

$$q_n^1 = \alpha_n p_n^1 \prod_{i \in M(n)} r_{i,n}^1 \tag{11.414}$$

where α_n is chosen such that $q_n^0 + q_n^1 = 1$.

At this point, the algorithm repeats from the first step. At the end of the second step, one iteration of the decoding algorithm is completed. At each iteration, it is possible to detect a code word $\hat{\mathbf{c}}$ by the log-MAP criterion (8.277), i.e. detect

$$\hat{c}_k = \operatorname{sgn}\left(\ln \frac{q_n^1}{q_n^0}\right) \qquad n = 1, \ldots, n_0 \tag{11.415}$$

Decoding is stopped if $\mathbf{H}\hat{\mathbf{c}} = 0$, or if some other stopping criterion is met, e.g., maximum number of iterations is achieved.

Messages passed between the nodes need not be probabilities but can be likelihood or log-likelihood ratios. In fact, various simplifications of the decoding algorithm have been explored and can be adopted for practical implementations [43, 44].

We note that the sum–product algorithm for the decoding of LDPC codes has been derived under the assumption that the check nodes s_i, $i = 1, \ldots, r_0$, are statistically independent given the bit nodes x_n, $n = 1, \ldots, n_0$, and vice versa, i.e. the variables of the vectors $\mathbf{s}$ and $\mathbf{x}$ form a Markov field [42]. Although this assumption is not strictly true, it turns out that the algorithm yields very good performance with low computational complexity. However, we note that parity check matrices leading to Tanner graphs that exhibit cycles of length four, such as the one depicted in Figure 11.39, should be avoided. In fact, this structure would introduce non-negligible statistical dependence between nodes. In graph theory, the length of the shortest cycle in a graph is referred to as *girth*. A general method for constructing Tanner graphs with large girth is described in [45].

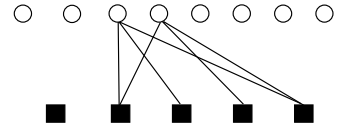


Figure 11.39. Tanner graph with a cycle of length four.

Example of application

We study in this section the application of binary LDPC codes to two-dimensional QAM transmission over an AWGN channel [46]. The block diagrams of the encoding and decoding processes are shown in Figure 11.40.

For bit mapping, $\log_2 M$ code bits are mapped into one QAM symbol taken from an M-point constellation using Gray mapping. At the receiver, the received samples, which represent noisy QAM symbols, are input to a soft detector that provides soft information on individual code bits in the form of *a posteriori* probabilities. These probabilities are employed to carry out the message-passing LDPC decoding procedure described in the previous section.

Assuming that the employed QAM constellation is square, with $\log_2 M$ equal to an even number, and that the in-phase and quadrature noise components are independent, it is computationally advantageous to perform soft detection independently for the real and imaginary parts of the received complex samples. We will therefore consider only square QAM constellations. Bit mapping for the real or the imaginary part of transmitted QAM symbols is performed by mapping a group of $\frac{1}{2}\log_2 M$ code bits $[c_0, c_1, \ldots, c_{\frac{1}{2}(\log_2 M)-1}]$ that are part of a code word into one of the $\sqrt{M}$ real symbols within the set

$$\mathcal{A} = \{-(\sqrt{M}-1), -(\sqrt{M}-3), \ldots, -1, +1, \ldots, +(\sqrt{M}-1)\} \tag{11.416}$$

Denoting by z_n the *real* or the *imaginary* part of a noisy received signal, we have

$$z_n = a_n + w_n \tag{11.417}$$

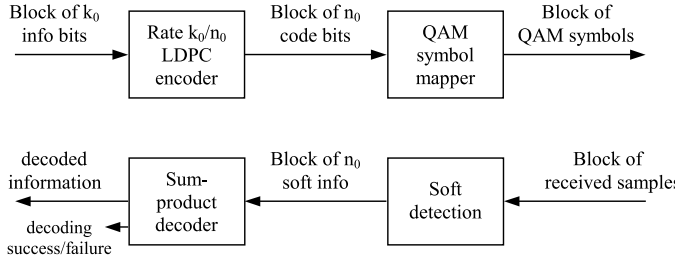


Figure 11.40. Multilevel LDPC encoding and decoding.

where $a_n \in \mathcal{A}$, and w_n is an AWGN sample with variance σ_I^2. The *a posteriori* probability that bit c_ℓ is zero or one is computed as (see (8.262))

$$P[c_\ell = \beta \mid z_n = \rho_n] = \frac{\sum\limits_{\substack{\alpha \in \mathcal{A} \\ c_\ell = \beta}} e^{-\frac{(\rho_n - \alpha)^2}{2\sigma_I^2}}}{\sum\limits_{\alpha \in \mathcal{A}} e^{-\frac{(\rho_n - \alpha)^2}{2\sigma_I^2}}} \qquad \ell = 0, 1, \ldots, \frac{1}{2}(\log_2 M) - 1 \qquad \beta \in \{0, 1\} \tag{11.418}$$

where the summation in the numerator is taken over all symbols $a_n \in \mathcal{A}$ for which $c_\ell = \beta$, $\beta \in \{0, 1\}$.

Performance and coding gain

Recall from (6.197) the expression of the error probability for uncoded M-QAM transmission,

$$P_e \simeq 4Q\left(\sqrt{\frac{3}{M-1}\Gamma}\right) \tag{11.419}$$

where Γ is the signal-to-noise ratio given by (6.105). In general, the relation between M and the rate of the encoder-modulator is given by (11.1),

$$R_I = \frac{k_0}{n_0}\frac{\log_2 M}{2} \tag{11.420}$$

Recall also, from (6.191), that the signal-to-noise ratio per dimension is given by

$$\Gamma_I = \Gamma = 2R_I \frac{E_b}{N_0} \tag{11.421}$$

Using (6.288) we introduce the normalized signal-to-noise ratio

$$\overline{\Gamma}_I = \frac{\Gamma_I}{2^{2R_I} - 1} = \frac{2R_I}{2^{2R_I} - 1}\frac{E_b}{N_0} \tag{11.422}$$

Then for an uncoded M-QAM system we express (11.419) as

$$P_e \simeq 4Q\left(\sqrt{3\overline{\Gamma}_I}\right) \tag{11.423}$$

As illustrated in Figure 6.54, the curve of P_e versus $\overline{\Gamma}_I$ indicates that the "gap to capacity" for uncoded QAM with $M \gg 1$ is equal to $\overline{\Gamma}_{gap,dB} \simeq 9.8$ dB at a symbol error probability of 10^{-7}. We therefore determine the value of the normalized signal-to-noise ratio $\overline{\Gamma}_I^c$ needed for the coded system to achieve a symbol error probability of 10^{-7}, and compute the coding gain at that symbol error probability as

$$G_{code} = 9.8 - 10\log_{10}(\overline{\Gamma}_I^c) \text{ dB} \tag{11.424}$$

Table 11.22 LDPC codes considered for the simulation and coding gains achieved at a symbol error probability of 10^{-7} for different QAM constellations. The spectral efficiencies ν are also indicated.

	k_0	n_0	code rate k_0/n_0	16-QAM	64-QAM	4096-QAM
Code 1	433	495	0.8747	4.9 dB (3.49 bit/s/Hz)	4.6 dB (5.24 bit/s/Hz)	3.5 dB (10.46 bit/s/Hz)
Code 2	1777	1998	0.8894	6.1 dB (3.55 bit/s/Hz)	5.9 dB (5.33 bit/s/Hz)	4.8 dB (10.62 bit/s/Hz)
Code 3	4095	4376	0.9358	6.2 dB (3.74 bit/s/Hz)	6.1 dB (5.61 bit/s/Hz)	5.6 dB (11.22 bit/s/Hz)

From Figure 6.54, as for large signal-to-noise ratios the Shannon limit cannot be approached to within less than 1.53 dB without shaping, we note that an upper limit to the coding gain measured in this manner is about 8.27 dB. Simulation results for three high-rate $(n_0,\ k_0)$ binary LDPC codes are specified in Table 11.22 in terms of the coding gains obtained at a symbol error probability of 10^{-7} for transmission over an AWGN channel for 16, 64, and 4096-QAM modulation formats. Transmitted QAM symbols are obtained from coded bits via Gray mapping. To measure error probabilities, one code word is decoded using the message-passing (sum-product) algorithm for given maximum number of iterations. Figure 11.41 shows the effect on performance of the maximum number of

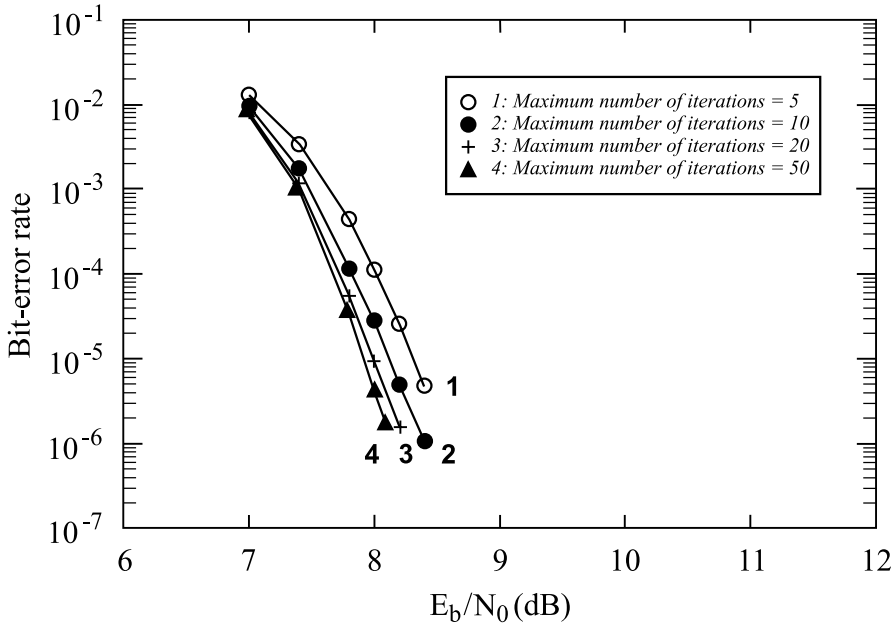


Figure 11.41. Performance of LDPC decoding with Code 2 and 16-QAM for various values of the maximum number of iterations.

iterations allowed in the decoding process for code 2 specified in Table 11.22 and 16-QAM transmission.

The codes given in Table 11.22 are due to MacKay and have been obtained by a random construction method. The results of Table 11.22 indicate that LDPC codes offer net coding gains that are similar to those that have been reported for turbo codes. Furthermore, LDPC codes achieve asymptotically an excellent performance without exhibiting "error floors" and admit a wide range of trade-offs between performance and decoding complexity.

Bibliography

[1] S. Lin and D. J. Costello Jr., *Error control coding*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[2] R. E. Blahut, *Theory and practice of error control codes*. Reading, MA: Addison-Wesley, 1983.

[3] W. W. Peterson and E. J. Weldon Jr., *Error-correcting codes*. Cambridge, MA: MIT Press, 2nd ed., 1972.

[4] J. K. Wolf, *Lecture notes*. San Diego, CA: University of California.

[5] S. B. Wicker and V. K. Bhargava, eds, *Reed-Solomon codes and their applications*. Piscataway, NJ: IEEE Press, 1994.

[6] R. Gallager, *Information theory and reliable communication*. New York: John Wiley & Sons, 1968.

[7] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision output and its applications", in *Proc. GLOBECOM '89*, Dallas, Texas, pp. 2828–2833, Nov. 1989.

[8] M. P. C. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics", *IEEE Trans. on Information Theory*, vol. 41, pp. 1379–1396, Sept. 1995.

[9] M. P. C. Fossorier and S. Lin, "Soft-input soft-output decoding of linear block codes based on ordered statistics", in *Proc. GLOBECOM '98*, Sidney, Australia, pp. 2828–2833, Nov. 1998.

[10] D. J. Costello Jr., J. Hagenauer, H. Imai, and S. B. Wicker, "Applications of error-control coding", *IEEE Trans. on Information Theory*, vol. 44, pp. 2531–2560, Oct. 1998.

[11] R. M. Fano, "A heuristic discussion on probabilistic decoding", *IEEE Trans. on Information Theory*, vol. 9, pp. 64–74, Apr. 1963.

[12] K. Zigangirov, "Some sequential decoding procedures", *Probl. Peredachi Informatsii*, vol. 2, pp. 13–25, 1966.

[13] F. Jelinek, "An upper bound on moments of sequential decoding effort", *IEEE Trans. on Information Theory*, vol. 15, pp. 140–149, Jan. 1969.

[14] F. Jelinek, "Fast sequential decoding algorithm using a stack", *IBM Journal of Research and Development*, vol. 13, pp. 675–685, Nov. 1969.

[15] F. Q. Wang and D. J. Costello, "Erasure-free sequential decoding of trellis codes", *IEEE Trans. on Information Theory*, vol. 40, pp. 1803–1817, Nov. 1994.

[16] C. F. Lin and J. B. Andeerson, "M-algorithm decoding of channel convolutional codes", in *Conf. Rec., Princeton Conf. Inform. Sci. Syst.*, Princeton, NJ, pp. 362–365, Mar. 1986.

[17] S. Benedetto and E. Biglieri, *Principles of digital transmission with wireless applications*. New York: Kluwer Academic Publishers, 1999.

[18] G. D. Forney, Jr., *Concatenated codes*. Cambridge, MA: MIT Press, 1966.

[19] M. P. C. Fossorier, F. Burkert, S. Lin, and J. Hagenauer, "On the equivalence between SOVA and max-log-MAP decodings", *IEEE Communications Letters*, vol. 2, pp. 137–139, May 1998.

[20] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo codes", in *Proc. IEEE Int. Conference on Communications*, Geneva, Switzerland, pp. 1064–1070, May 23–26, 1993.

[21] C. Berrou and A. Glavieux, "Near optimum error-correcting coding and decoding: turbo-codes", *IEEE Trans. on Communications*, vol. 44, pp. 1261–1271, Oct. 1996.

[22] B. Sklar, "A primer on turbo code concepts", *IEEE Communications Magazine*, vol. 35, pp. 94–101, Dec. 1997.

[23] S. Benedetto and G. Montorsi, "Unveiling turbo codes: some results on parallel concatenated coding schemes", *IEEE Trans. on Information Theory*, vol. 42, pp. 409–428, Mar. 1996.

[24] 3-rd Generation Partnership Project (3GPP), Technical Specification Group (TSG), Radio Access Network (RAN), Working Group 1 (WG1), "Multiplexing and channel coding (TDD)", *Document TS 1.22, v.2.0.0*, Apr. 2000.

[25] International Telecommunication Union (ITU), Radiocommunication Study Groups, "A guide to digital terrestrial television broadcasting in the VHF/UHF bands", *Doc. 11-3/3-E*, Mar. 1998.

[26] Consultative Committee for Space Data Systems (CCSDS), Telemetry Systems (Panel 1), "Telemetry channel coding", *Blue Book, CCSDS 101.0-B-4: Issue 4*, May 1999.

[27] S. Benedetto, R. Garello, and G. Montorsi, "A search for good convolutional codes to be used in the construction of turbo codes", *IEEE Trans. on Communications*, vol. 46, pp. 1101–1105, Sept. 1998.

[28] O. Y. Takeshita and D. J. Costello Jr., "New deterministic interleaver designs for turbo codes", *IEEE Trans. on Information Theory*, vol. 46, pp. 1988–2006, Sept. 2000.

[29] D. Divsalar and F. Pollara, "Turbo codes for PCS applications", in *Proc. 1995 IEEE Int. Conference on Communications*, Seattle, U.S.A., pp. 54–59, June 1995.

[30] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design, and iterative decoding", *IEEE Trans. on Information Theory*, vol. 45, pp. 909–926, May 1998.

[31] S. Benedetto and G. Montorsi, "Versatile bandwidth-efficient parallel and serial turbo-trellis-coded modulation", in *Proc. 2000 Intern. Symp. on Turbo Codes & Relat. Topics*, Brest, France, pp. 201–208, Sept. 2000.

[32] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference turbo-equalization", *Europ. Trans. on Telecommun. (ETT)*, vol. 6, pp. 507–511, September/October 1995.

[33] G. Colavolpe, G. Ferrari, and R. Raheli, "Noncoherent iterative (turbo) decoding", *IEEE Trans. on Communications*, vol. 48, pp. 1488–1498, Sept. 2000.

[34] G. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications system", in *Proc. European Personal Mobile Communications Conference*, Bristol, UK, pp. 307–312, 1997.

[35] P. Hoeher and J. Lodge, "Turbo-DPSK": iterative differential PSK demodulation", *IEEE Trans. on Communications*, vol. 47, pp. 837–843, June 1999.

[36] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes", *IEEE Trans. on Information Theory*, vol. 42, pp. 429–445, Mar. 1996.

[37] G. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems", in *Proc. EPMCC*, pp. 307–312, Oct. 1997.

[38] A. Picart, P. Didier, and G. Glavieux, "Turbo-detection: a new approach to combat channel frequency selectivity", in *Proc. IEEE Int. Conference on Communications*, pp. 1498–1502, 1997.

[39] D. MacKay and R. Neal, "Near Shannon limit performance of low density parity check codes", *Electron. Lett.*, vol. 32, pp. 1645–1646, Aug. 1996.

[40] D. MacKay, "Good error-correcting codes based on very sparse matrices", *IEEE Trans. on Information Theory*, vol. 45, pp. 399–431, Mar. 1999.

[41] R. M. Tanner, "A recursive approach to low complexity codes", *IEEE Trans. on Information Theory*, vol. 27, pp. 533–547, Sept. 1981.

[42] B. J. Frey, *Graphical models for machine learning and digital communications*. Cambridge, MA: MIT Press, 1998.

[43] M. P. C. Fossorier, M. Mahaljevic, and H. Imai, "Reduced complexity iterative decoding of low-density parity check codes based on belief propagation", *IEEE Trans. on Computers*, vol. 47, pp. 673–680, May 1999.

[44] X.-Y. Hu, E. Eleftheriou, D. M. Arnold, and A. Dholakia, "Efficient implementations of the sum-product algorithm for decoding LDPC codes", in *Proc. GLOBECOM '01*, San Antonio, TX, Nov. 2001.

[45] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, "Progressive edge-growth Tanner graphs", in *Proc. GLOBECOM '01*, San Antonio, TX, Nov. 2001.

[46] G. Cherubini, E. Eleftheriou, and S. Ölcer, "On advanced signal processing and coding techniques for digital subscriber lines", Records of the Workshop "What is *next* in xDSL?", Vienna, Austria, Sept. 2000.

Appendix 11.A *Nonbinary parity check codes*

Assume that code words are sequences of symbols from the finite field $GF(q)$ (see Section 11.2.2), all of length n. As there are q^n possible sequences, the introduction of redundancy in the transmitted sequences is possible if the number of code words M_c is less than q^n.

We denote by $\mathbf{c}$ a transmitted sequence of n symbols taken from $GF(q)$. We also assume that the symbols of the received sequence $\mathbf{z}$ are from the same alphabet. We define the *error sequence* $\mathbf{e}$ by the equation (see (11.12) for the binary case)

$$\mathbf{z} = \mathbf{c} + \mathbf{e} \tag{11.425}$$

where $+$ denotes a component by component addition of the vectors in accordance with the rules of addition in the field $GF(q)$.

Definition 11.16
The number of non-zero components of a vector $\mathbf{x}$ is defined as the weight of the vector, denoted by $w(\mathbf{x})$.

Then $w(\mathbf{e})$ is equal to the number of errors occurred in transmitting the code word.

Definition 11.17
The *minimum distance* of a code, denoted d^H_{min}, is equal to the minimum Hamming distance between all pairs of code words; i.e. it is the same as for binary codes.

We will give without proof the following propositions, similar to those for binary codes on page 830.

1. A nonbinary block code with minimum distance d^H_{min} can correct all error sequences of weight $\left\lfloor \frac{d^H_{min}-1}{2} \right\rfloor$ or less.
2. A nonbinary block code with minimum distance d^H_{min} can detect all error sequences of weight $(d^H_{min} - 1)$ or less.

As in the binary case, we ask for a relation among the parameters of a code: n, M_c, d^H_{min}, and q. It can be proved that for a block code with length n and minimum distance d^H_{min}, M_c must satisfy the inequality

$$M_c \leq \left\lfloor q^n \Big/ \left\{ 1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{\left\lfloor \frac{d^H_{min}-1}{2} \right\rfloor}(q-1)^{\left\lfloor \frac{d^H_{min}-1}{2} \right\rfloor} \right\} \right\rfloor \tag{11.426}$$

Furthermore, for n and d_{min}^H given, it is always possible to find a code with M_c^* words, where

$$M_c^* = \left\lceil q^n \Big/ \left\{ 1 + \binom{n}{1}(q-1) + \binom{n}{2}(q-1)^2 + \cdots + \binom{n}{d_{min}^H - 1}(q-1)^{d_{min}^H - 1} \right\} \right\rceil \tag{11.427}$$

Linear codes

Definition 11.18
A *linear code* is a block code with symbols from $GF(q)$ for which:

a) the all zero word is a code word;

b) any multiple of a code word is a code word;

c) any linear combination of any two code words is a code word.

Example 11.A.1
A binary group code is a linear code with symbols from $GF(2)$.

Example 11.A.2
Consider a block code of length 5 having symbols from $GF(3)$ with code words

$$\begin{array}{ccccc} 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 1 \\ 0 & 1 & 1 & 2 & 2 \\ 2 & 0 & 0 & 1 & 2 \\ 1 & 1 & 1 & 1 & 0 \\ 2 & 1 & 1 & 0 & 1 \\ 0 & 2 & 2 & 1 & 1 \\ 1 & 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 2 & 0 \end{array} \tag{11.428}$$

It is easily verified that this code is a linear code.

We give the following two properties of a linear code.

1. The minimum distance of the code, d_{min}^H, is given as

$$d_{min}^H = \min\, w(\bar{\mathbf{c}}) \tag{11.429}$$

where $\bar{\mathbf{c}}$ can be any non-zero code word.

Proof. By definition of the Hamming distance between two code words, we get

$$d^H(\mathbf{c}_1, \mathbf{c}_2) = w(\mathbf{c}_1 + (-\mathbf{c}_2)) \tag{11.430}$$

By Property b), $(-\mathbf{c}_2)$ is a code word if $\mathbf{c}_2$ is a code word; by Property c), $\mathbf{c}_1 + (-\mathbf{c}_2)$ must also be a code word. As two code words differ in at least d_{min}^H positions, there is a code word of weight d_{min}^H; if there were a code word of weight less than d_{min}^H, this word would be different from the zero word in fewer than d_{min}^H positions.

2. If all code words in a linear code are written as rows of an $M_c \times n$ matrix, every column is composed of all zeros, or contains all elements of the field, each repeated M_c/q times.

Parity check matrix

Let $\mathbf{H}$ be an $r \times n$ matrix with coefficients from $GF(q)$, expressed as

$$\mathbf{H} = [\mathbf{A}\ \mathbf{B}] \tag{11.431}$$

where the $r \times r$ matrix $\mathbf{B}$ is such that $\det[\mathbf{B}] \neq 0$.

A generalized nonbinary parity check code is a code composed of all vectors $\mathbf{c}$ of length n, with elements from $GF(q)$, that are the solutions of the equation

$$\mathbf{Hc} = \mathbf{0} \tag{11.432}$$

The matrix $\mathbf{H}$ is called the *generalized parity check matrix*.

Propriety 1 of nonbinary generalized parity check codes. A nonbinary generalized parity check code is a linear code.

Proof.

a) The all zero word is a code word, as $\mathbf{H0} = \mathbf{0}$.

b) Any multiple of a code word is a code word, because if $\mathbf{c}$ is a code word, then $\mathbf{Hc} = \mathbf{0}$. But $\mathbf{H}(\alpha\mathbf{c}) = \alpha\mathbf{Hc} = \mathbf{0}$, and therefore $\alpha\mathbf{c}$ is a code word; here α is any element from $GF(q)$.

c) Any linear combination of any two code words is a code word, because if $\mathbf{c}_1$ and $\mathbf{c}_2$ are two code words, then $\mathbf{H}(\alpha\mathbf{c}_1 + \beta\mathbf{c}_2) = \alpha\mathbf{Hc}_1 + \beta\mathbf{Hc}_2 = \alpha\mathbf{0} + \beta\mathbf{0} = \mathbf{0}$, and therefore $\alpha\mathbf{c}_1 + \beta\mathbf{c}_2$ is a code word.

Property 2 of nonbinary generalized parity check codes. The code words corresponding to the matrix $\mathbf{H} = [\mathbf{A}\ \mathbf{B}]$ are identical to the code words corresponding to the parity check matrix $\tilde{\mathbf{H}} = [\mathbf{B}^{-1}\mathbf{A}, \mathbf{I}]$.

Proof. Same as for the binary case.

The matrices in the form $[\mathbf{A}\ \mathbf{I}]$ are said to be in *canonical or systematic form*.

Property 3 of nonbinary generalized parity check codes. A code consists of exactly $q^{n-r} = q^k$ code words.

Proof. Same as for the binary case (see Property 3 on page 834).

The first $k = n - r$ symbols are called *information symbols*, and the last r symbols are called *generalized parity check symbols*.

Property 4 of nonbinary generalized parity check codes. A code word of weight w exists if and only if some linear combination of w columns of the matrix $\mathbf{H}$ is equal to $\mathbf{0}$.

Proof. $\mathbf{c}$ is a code word if and only if $\mathbf{Hc} = \mathbf{0}$. Let c_j be the j-th component of $\mathbf{c}$ and let $\mathbf{h}_i$ be the i-th column of $\mathbf{H}$; then if $\mathbf{c}$ is a code word we have

$$\sum_{j=1}^{n} \mathbf{h}_j \, c_j = \mathbf{0} \tag{11.433}$$

If $\mathbf{c}$ is a code word of weight w, there are exactly w non-zero components of $\mathbf{c}$, say $c_{j_1}, c_{j_2}, \ldots, c_{j_w}$; then

$$c_{j_1}\mathbf{h}_{j_1} + c_{j_2}\mathbf{h}_{j_2} + \cdots + c_{j_w}\mathbf{h}_{j_w} = \mathbf{0} \tag{11.434}$$

thus, a linear combination of w columns of $\mathbf{H}$ is equal to $\mathbf{0}$. Conversely, if (11.434) is true, then $\mathbf{Hc} = \mathbf{0}$, where $\mathbf{c}$ is a vector of weight w with non-zero components $c_{j_1}, c_{j_2}, \ldots, c_{j_w}$.

Combining Property 1 of a linear code and Properties 1 and 4 of a nonbinary generalized parity check code, we obtain the following property.

Property 5 of nonbinary generalized parity check codes. A code has minimum distance d_{min}^H if some linear combination of d_{min}^H columns of $\mathbf{H}$ is equal to $\mathbf{0}$, but no linear combination of fewer than d_{min}^H number of columns of $\mathbf{H}$ is equal to $\mathbf{0}$.

Property 5 is fundamental for the design of nonbinary codes.

Example 11.A.3

Consider the field $GF(4)$, and let α be a primitive element of this field; moreover consider the generalized parity check matrix

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & \alpha & \alpha^2 & 0 & 1 \end{bmatrix} \tag{11.435}$$

We find that no linear combination of two columns is equal to $\mathbf{0}$. However, there are many linear combinations of three columns that are equal to $\mathbf{0}$, for example, $\mathbf{h}_1 + \mathbf{h}_4 + \mathbf{h}_5 = \mathbf{0}$, $\alpha\mathbf{h}_2 + \alpha\mathbf{h}_4 + \alpha^2\mathbf{h}_5 = \mathbf{0}$, ...; hence the minimum distance of this code is $d_{min}^H = 3$.

Code generator matrix

We assume that the parity check matrix is in canonical form; then

$$\mathbf{c}_{n-r+1}^{n} = -\mathbf{A}\mathbf{c}_1^{n-r} \tag{11.436}$$

and

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1^{n-r} \\ \mathbf{c}_{n-r+1}^{n} \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{A} \end{bmatrix} \mathbf{c}_1^{n-r} = \mathbf{G}^T \, \mathbf{c}_1^{n-r} \tag{11.437}$$

The matrix $\mathbf{G}$ is called the *generator matrix* of the code and is expressed as

$$\mathbf{G} = [\mathbf{I}, -\mathbf{A}^T] \tag{11.438}$$

so that

$$\mathbf{c}^T = (\mathbf{c}_1^{n-r})^T \mathbf{G} \tag{11.439}$$

Thus the code words, considered as row vectors, are given as all linear combinations of the rows of the matrix $\mathbf{G}$. A nonbinary generalized parity check code can be specified by giving its generalized parity check matrix or its generator matrix.

Example 11.A.4
Consider the field $GF(4)$ and let α be a primitive element of this field; moreover, consider the generalized parity check matrix (11.435). The generator matrix of this code is given by

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & \alpha \\ 0 & 0 & 1 & 1 & \alpha^2 \end{bmatrix} \tag{11.440}$$

There are 64 code words corresponding to all linear combinations of the rows of the matrix $\mathbf{G}$.

Decoding of nonbinary parity check codes

Methods for the decoding of nonbinary generalized parity check codes are similar to those for the binary case. Conceptually the simplest method consists in comparing the received block of n symbols with each code word and choosing that code word that differs from the received word in the fewest positions. An equivalent method for a linear code consists in partitioning the q^n possible sequences into q^r sets. The partitioning is done as follows.

Step 1: choose the first set as the set of $q^{n-r} = q^k$ code words, $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_{q^k}$.

Step 2: choose any vector, say $\boldsymbol{\eta}_2$, that is not a code word; then choose the second set as $\mathbf{c}_1 + \boldsymbol{\eta}_2, \mathbf{c}_2 + \boldsymbol{\eta}_2, \ldots, \mathbf{c}_{q^k} + \boldsymbol{\eta}_2$.

Step i: choose any vector, say $\boldsymbol{\eta}_i$, not included in any previous set; choose the i-th set as $\mathbf{c}_1 + \boldsymbol{\eta}_i, \mathbf{c}_2 + \boldsymbol{\eta}_i, \ldots, \mathbf{c}_{q^k} + \boldsymbol{\eta}_i$.

The partitioning continues until all q^n vectors are used; each set is called a *coset*, and the vectors $\boldsymbol{\eta}_i$ are called *coset leaders*. The all zero vector is the *coset leader* for the first set.

Coset

We give the following properties of the cosets omitting the proofs.

1. Every one of the q^n vectors occurs in one and only one coset.
2. Suppose that, instead of choosing $\boldsymbol{\eta}_i$ as coset leader of the i-th coset, we choose another element of that coset as the coset leader; then the coset formed by using the new coset leader contains exactly the same vectors as the old coset.
3. There are q^r cosets.

Two conceptually simple decoding methods

We now form a *coset table* by choosing as coset leader for each coset the vector of minimum weight in that coset. The table consists of an array of vectors, with the i-th row in the array being the i-th coset; the coset leaders make up the first column, and the j-th column consists of the vectors $\mathbf{c}_j, \mathbf{c}_j + \boldsymbol{\eta}_2, \mathbf{c}_j + \boldsymbol{\eta}_3, \ldots, \mathbf{c}_j + \boldsymbol{\eta}_{q^r}$.

A method for decoding consists of the following steps.

Step 1: locate the received vector in the coset table.

Step 2: choose the code word that appears as the first vector in the column containing the received vector.

This decoding method decodes to the closest code word to the received word and the coset leaders are the correctable error patterns.

A modified version of the described decoding method is:

Step $1'$*:* locate the received vector in the coset table and then identify the coset leader of the coset containing this vector.

Step $2'$*:* subtract the coset leader from the received vector to find the decoded code word.

Syndrome decoding

Another method of decoding is the syndrome decoding. For any generalized parity check matrix $\mathbf{H}$ and all vectors $\mathbf{z}$ of length n, we define the syndrome of $\mathbf{z}$, $\mathbf{s}(\mathbf{z})$, as

$$\mathbf{s}(\mathbf{z}) = \mathbf{H}\mathbf{z} \tag{11.441}$$

We can show that all vectors in the same coset have the same syndrome and vectors in different cosets have different syndromes. This leads to the following decoding method:

Step $1''$*:* compute the syndrome of the received vector, as this syndrome identifies the coset in which the received vector is in, and so identifies the leader of that coset.

Step $2''$*:* subtract the coset leader from the received vector to find the decoded code word.

The difficulty with this decoding method is in the second part of step $1''$, that is identifying the coset leader that corresponds to the computed syndrome; this step is equivalent to finding a linear combination of the columns of $\mathbf{H}$ which is equal to that syndrome, using the smallest number of columns. The algebraic structure of the generalized parity check matrix for certain classes of codes allows for algebraic means of finding the coset leader from the syndrome.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 12

Trellis coded modulation

During the 1980s an evolution in the methods to transmit data over channels with limited bandwidth took place, giving origin to techniques for joint coding and modulation that are generally known by the name of *trellis coded modulation* (TCM). The main characteristic of TCM lies in the fact that it yields coding gains with respect to conventional modulation techniques without requiring that the channel bandwidth be increased. The first article on TCM appeared in 1976 by Ungerboeck; later, a more detailed publication by the same author on the principles of TCM [1] spurred considerable interest in this topic [2, 3, 4, 5, 6, 7, 8], leading to a full development of the theory of TCM.

TCM techniques use multilevel modulation with a set of signals from a one, two, or multi-dimensional space. The choice of the signals that generate a code sequence is determined by a finite-state encoder. In TCM, the set of modulation signals is expanded with respect to the set used by an uncoded, i.e. without redundancy, system; in this manner, it is possible to introduce redundancy in the transmitted signal without widening the bandwidth. At the receiver, the signals in the presence of additive noise and channel distortion are decoded by a maximum likelihood sequence decoder. By simple TCM techniques using a four-state encoder, it is possible to obtain a coding gain of 3 dB with respect to conventional uncoded modulation; with more sophisticated TCM techniques, coding gains of 6 dB or more can be achieved (see Chapter 6).

Errors in the decoding of the received signal sequence are less likely to occur if the waveforms, which represent the code sequences, are easily distinguishable from each other; in mathematical terms, the signal sequences, represented in the Euclidean multidimensional space, need to be separated by large distances. The novelty of TCM is in postulating the expansion of the set of symbols[1] in order to provide the redundancy necessary for the encoding process. The construction of modulation code sequences that are characterized by a *free distance*, i.e. the minimum Euclidean distance between code sequences, that is much larger than the minimum distance between uncoded modulation symbols, with the same information bit rate, and the same bandwidth and power of the modulated signal, is obtained by the joint design of encoder and bit mapper. The term trellis derives from the similarity between state transition diagrams of a TCM encoder and trellis diagrams

[1] In the first part of this chapter we mainly use the notion of *symbols* of an alphabet with cardinality M, although the analysis could be conducted by referring to vectors in the signal space as *modulation signals*. We will use the term "signals" instead of "symbols" only in the multidimensional case.

of binary convolutional codes; the difference lies in the fact that, in TCM schemes, the branches of the trellis are labeled with modulation symbols rather than binary symbols.

Thanks to the use of sophisticated TCM schemes, it was possible to achieve reliable data transmission over telephone channels at rates much higher than 9.6 kbit/s, which for years was considered the practical limit. In the mid-1980s, the rate of 14.4 kbit/s was reached. Transmission at a maximum bit rate of 28.8 kbit/s was later specified in the standard CCITT V.34, and extensions were proposed to achieve the rates of 31.2 kbit/s and 33.6 kbit/s.

12.1 Linear TCM for one- and two-dimensional signal sets

12.1.1 Fundamental elements

Consider the transmission system illustrated in Figure 6.20, that consists of the modulator, transmission channel, demodulator and data detector. Errors occasionally occur in the symbol detection, and $\hat{a}_k \neq a_k$. Usually the simplest data detector is a threshold detector that takes an instantaneous hard decision on the value $\hat{a}_k$ of the transmitted symbol, based on the observation of the sample z_k at the demodulator output. Detection is of the nearest-neighbor type, i.e. the detector decides for the symbol of the constellation that is at the minimum Euclidean distance from the received sample z_k. The objective of traditional channel coding techniques consists in detecting and/or correcting the errors present in the detected sequence of bits $\{\tilde{c}_m\}$.

In the approach followed in Chapter 11, a binary encoder was used to map k_0 information binary symbols $\{b_\ell\}$ in n_0 code binary symbols $\{c_m\}$. As mentioned in Section 11.1, we note that, if we want to maintain the effective rate $1/T_b$ of the information message and at the same time the modulation rate $1/T$ of the system, we need to increase the cardinality of the modulation alphabet. If we do not consider the joint design of encoder and bit mapper, however, a reduction of the bit error probability cannot be efficiently achieved, as we see from the following example.

Example 12.1.1
Consider an uncoded 4-PSK system and an 8-PSK system that uses a binary error correcting code with rate 2/3; both systems transmit two information bits per modulation interval, which corresponds to a spectral efficiency of 2 bit/s/Hz. If the 4-PSK system works with an error probability of 10^{-5}, for a given signal-to-noise ratio Γ, the 8-PSK system works with an error probability larger than 10^{-2}, due to the smaller Euclidean distance between signals of the 8-PSK system. We must use an error correcting code with minimum Hamming distance $d^H_{min} \geq 7$ to reduce the error probability to the same value of the uncoded 4-PSK system. A binary convolutional code with rate 2/3 and constraint length 6 has the required value of $d^H_{free} = 7$. Decoding requires a decoder with 64 states that implements the Viterbi algorithm. However, even after increasing the complexity of the 8-PSK system, we have obtained an error probability only equal to that of the uncoded 4-PSK system.

Two problems determine the unsatisfactory result obtained with the traditional approach. The first is originated by the use of independent hard decisions taken by the detector before

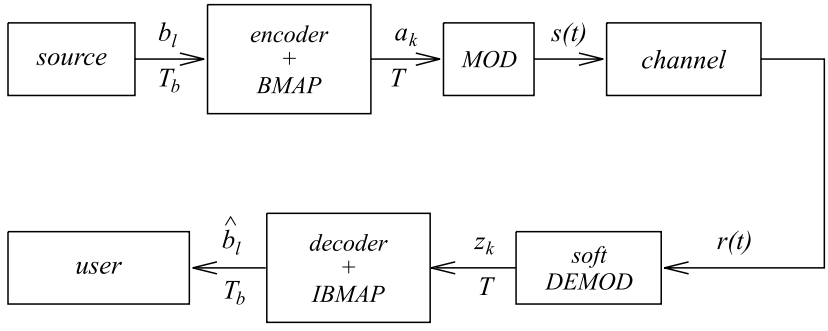


Figure 12.1. Block diagram of a transmission system with trellis coded modulation.

decoding; hard input decoding leads to an irreversible loss of information; the remedy is the use of soft decoding (see page 912), whereby the decoder directly operates on the samples at the demodulator output. The second derives from the independent design of encoder and bit mapper.

We now consider the transmission system of Figure 12.1, where the transmitted symbol sequence $\{a_k\}$ is produced by a finite-state machine having the information bit sequence $\{b_\ell\}$ as input, possibly with a number of information bits per modulation interval larger than one. We denote by $1/T$ the modulation rate and by $\mathcal{A}$ the alphabet of a_k. For an AWGN channel, at the decision point the received samples in the absence of ISI are given by (see (8.173))

$$z_k = a_k + w_k \tag{12.1}$$

where $\{w_k\}$ is a sequence of white Gaussian noise samples. Maximum likelihood sequence detection represents the optimum strategy for decoding a sequence transmitted over a dispersive noisy channel. The decision rule consists in determining the sequence $\{\hat{a}_k\}$ closest to the received sequence in terms of Euclidean distance (see (8.190)) in the set $\mathcal{S}$ of all possible code symbol sequences. MLSD is efficiently implemented by the Viterbi algorithm, provided that the generation of the code symbol sequences follows the rules of a finite-state machine.

In relation to (8.194), we define as *free distance*, d_{free}, the minimum Euclidean distance between two code symbol sequences $\{\alpha_k\}$ and $\{\beta_k\}$, that belong to the set $\mathcal{S}$, given by

$$d^2_{free} = \min_{\{\alpha_k\} \neq \{\beta_k\}} \sum_k |\alpha_k - \beta_k|^2 \qquad \{\alpha_k\}, \{\beta_k\} \in \mathcal{S} \tag{12.2}$$

The most probable error event is determined by two code symbol sequences of the set $\mathcal{S}$ at the minimum distance. The assignment of symbol sequences using a code that is optimized for Hamming distance does not guarantee an acceptable structure in terms of Euclidean distance, as in general the relation between the Hamming distance and the Euclidean distance is not monotonic.

Encoder and modulator must then be jointly designed for the purpose of assigning to symbol sequences waveforms that are separated in the Euclidean signal space by a distance

equal to at least d_{free}, where d_{free} is greater than the minimum distance between the symbols of an uncoded system.

At the receiver, the demodulator-decoder does not make errors if the received signal in the Euclidean signal space is at a distance smaller than $d_{free}/2$ from the transmitted sequence.

Basic TCM scheme

The objective of TCM is to obtain an error probability lower than that achievable with uncoded modulation, for the same bit rate of the system, channel bandwidth, transmitted signal power and noise power spectral density.

The generation of code symbol sequences by a sequential finite-state machine (FSM) sets some constraints on the symbols of a sequence, thus introducing interdependence among them (see Appendix 8.D). The transmitted symbol at instant kT depends not only on the information bits generated by the source at the same instant, as in the case of memoryless modulation, but also on the previous symbols.

We define as $\mathbf{b}_k$ the vector of $\log_2 M$ information bits at instant kT. We recall that for M-ary uncoded transmission there exists a one-to-one correspondence between $\mathbf{b}_k$ and the symbol $a_k \in \mathcal{A}$. We also introduce the state $\mathbf{s}_k$ at the instant kT. According to the model of Appendix 8.D, the generation of a sequence of encoded symbols is obtained by the two functions

$$\begin{aligned} a_k &= f(\mathbf{b}_k, \mathbf{s}_{k-1}) \\ \mathbf{s}_k &= g(\mathbf{b}_k, \mathbf{s}_{k-1}) \end{aligned} \tag{12.3}$$

For an input vector $\mathbf{b}_k$ and a state $\mathbf{s}_{k-1}$, the first equation describes the choice of the transmitted symbol a_k from a certain constellation, the second the choice of the next state $\mathbf{s}_k$.

Interdependence between the symbols $\{a_k\}$ is introduced without a reduction of the bit rate by increasing the cardinality of the alphabet. For example, for a length K of the sequence of input vectors, if we change $\mathcal{A}$ of cardinality M with $\mathcal{A}' \supset \mathcal{A}$ of cardinality $M' > M$, and we select M^K sequences as a subset of $(\mathcal{A}')^K$, a better separation of the code sequences in the Euclidean space may be obtained. Hence, we can obtain a minimum distance d_{free} between any two sequences larger than the minimum distance between signals in $\mathcal{A}^K$.

Note that this operation may cause an increase in the average symbol energy from $E_{s,u}$ for uncoded transmission to $E_{s,c}$ for coded transmission, and hence a loss in efficiency given by $E_{s,c}/E_{s,u}$.

Furthermore, we define as N_{free} the number of sequences that a code sequence has, *on average*, at the distance d_{free} in the Euclidean multidimensional space.

Example

Suppose we want to transmit two bits of information per symbol. Instead of using QPSK modulation, we can use the scheme illustrated in Figure 12.2.

The scheme has two parts. The first is a finite-state sequential machine with 8 states, where the state $\mathbf{s}_k$ is defined by the content of the memory cells $\mathbf{s}_k = [s_k^{(2)}, s_k^{(1)}, s_k^{(0)}]$. The

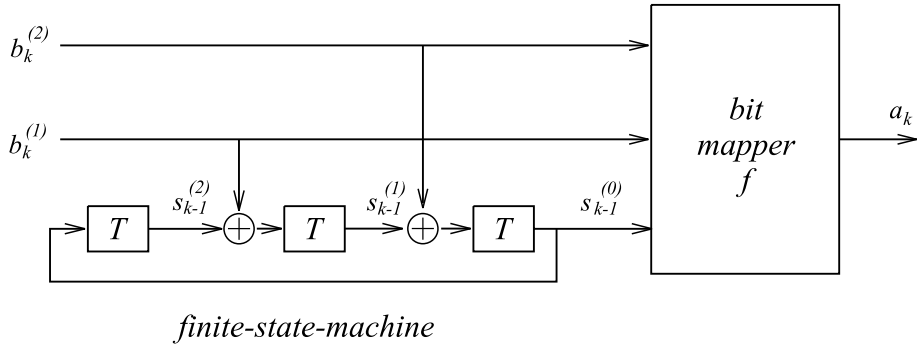


Figure 12.2. Eight-state trellis encoder and bit mapper for the transmission of 2 bits per modulation interval by 8-PSK.

two bits $\mathbf{b}_k = [b_k^{(2)}, b_k^{(1)}]$ are input to the FSM, which undergoes a transition from state $\mathbf{s}_{k-1}$ to one of four next possible states, $\mathbf{s}_k$, according to the function g. The second part is the bit mapper, which maps the two information bits and one bit that depends on the state, i.e. the three bits $[b_k^{(2)}, b_k^{(1)}, s_{k-1}^{(0)}]$, in one of the symbols of an eight-ary constellation according to the function f, for example, an 8-PSK constellation using the map of Figure 12.5. Note that the transmission of two information bits per modulation interval is achieved. Therefore the constellation of the system is expanded by a factor 2 with respect to uncoded QPSK transmission. Recall from the discussion in Section 6.10 that most of the achievable coding gain for transmission over an ideal AWGN channel of two bits per modulation interval can be obtained by doubling the cardinality of the constellation from four to eight symbols. We will see that trellis coded modulation using the simple scheme of Figure 12.2 allows to achieve a coding gain of 3.6 dB.

For the graphical representation of the functions f and g, it is convenient to use a trellis diagram; the nodes of the trellis represent the FSM states and the branches represent the possible transitions between states. For a given state $\mathbf{s}_{k-1}$, a branch is associated with each possible vector $\mathbf{b}_k$ by the function g, that reaches a next state $\mathbf{s}_k$. Each branch is labeled with the corresponding value of the transmitted symbol a_k. For the encoder of Figure 12.2 and the map of Figure 12.5, the corresponding trellis is shown in Figure 12.3, where the trellis is terminated by forcing the state of the FSM to zero at the instant $k = 4$. For a general representation of the trellis, see Figure 12.13.

Each path of the trellis corresponds to only one message sequence $\{b_\ell\}$ and is associated with only one sequence of code symbols $\{a_k\}$. The optimum decoder searches the trellis for the most probable path, given the received sequence $\{z_k\}$ is observed at the output of the demodulator. This search is usually realized by the Viterbi algorithm (see Section 8.10). Because of the presence of noise, the chosen path may not coincide with the correct one, but diverge from it at the instant $k = i$ and rejoin it at the instant $k = i + L$; in this case we say that an *error event of length L* has occurred, as illustrated in the example in Figure 12.4 for an error event of length two (see Definition 8.1 on page 683).

Note that in a trellis diagram more branches may connect the same pair of nodes. In this case we speak of parallel transitions, and by the term free distance of the code

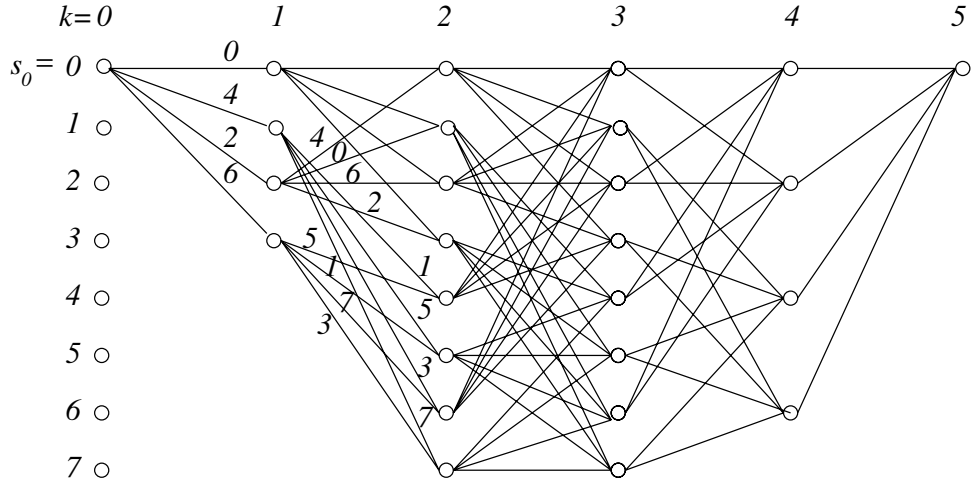


Figure 12.3. Trellis diagram for the encoder of Figure 12.2 and the map of Figure 12.5. Each branch is labeled with the corresponding value of a_k.

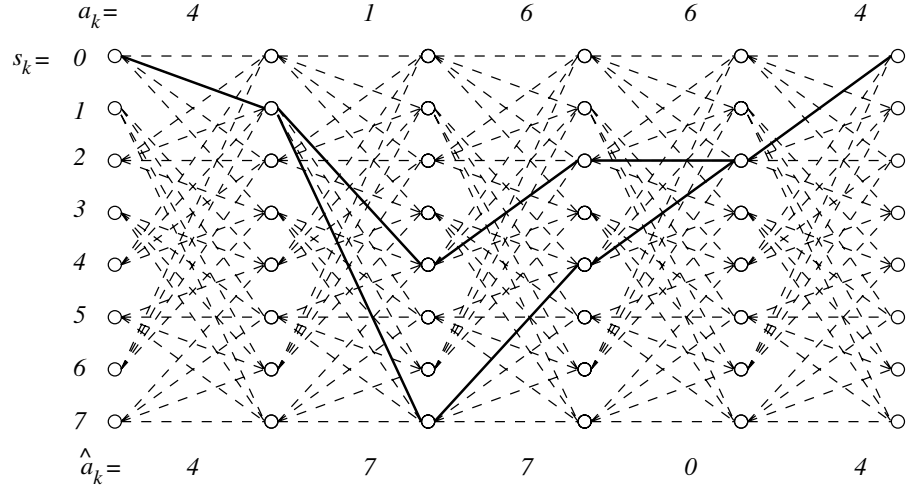


Figure 12.4. Section of the trellis for the decoder of an eight-state trellis code. The two continuous lines indicate two possible paths relative to two 8-PSK signal sequences, $\{a_k\}$ and $\{\hat{a}_k\}$.

we denote the minimum among the distances between symbols on parallel transitions and the distances between code sequences associated with pairs of paths in the trellis that originate from a common node and merge into a common node after L transitions, $L > 1$.

By utilizing the sequence of samples $\{z_k\}$, the decoding of a TCM signal is done in two phases. In the first phase, called *subset decoding*, within each subset of symbols assigned

to the parallel transitions in the trellis diagram, the receiver determines the symbol closest to the received sample; these symbols are then memorized together with their squared distances from the received sample. In the second phase we apply the Viterbi algorithm to find the code sequence $\{\hat{a}_k\}$ along the trellis such that the sum of the squared distances between the code sequence and the sequence $\{z_k\}$ is minimum. Recalling that the signal is obtained at the output of the demodulator in the presence of additive white Gaussian noise with variance σ_I^2 per dimension, the probability of an error event for large values of the signal-to-noise ratio is approximated by (see (8.195))

$$P_e \simeq N_{free}\, Q\left(\frac{d_{free}}{2\sigma_I}\right) \tag{12.4}$$

where d_{free} is defined in (12.2).

From the Definition 6.2 on page 508 and the relation (12.4) between Euclidean distance and error probability, we give the definition of *asymptotic coding gain*, G_{code},[2] as the ratio between the minimum distance, d_{free}, between code sequences and the minimum Euclidean distance for uncoded sequences, equal to the minimum distance between symbols of the constellation of an uncoded system, $\tilde{\Delta}_0$, normalized by the ratio between the average energy of the coded sequence, $E_{s,c}$, and the average energy of the uncoded sequence, $E_{s,u}$. The coding gain is then expressed in dB as

$$G_{code} = 10\log_{10}\frac{d_{free}^2/\tilde{\Delta}_0^2}{E_{s,c}/E_{s,u}} \tag{12.5}$$

12.1.2 Set partitioning

The design of trellis codes is based on a method called mapping by set partitioning. This method requires that the bit mapper assign symbol values to the input binary vectors so that the minimum Euclidean distance between possible code sequences $\{a_k\}$ is maximum. For a given encoder the search of the optimum assignment is made by taking into consideration subsets of the symbol set $\mathcal{A}$. These subsets are obtained by successive partitioning of the set $\mathcal{A}$, and are characterized by the property that the minimum Euclidean distance between symbols in a subset corresponding to a certain level of partitioning is larger than or equal to the minimum distance obtained at the previous level.

Consider the symbol alphabet $\mathcal{A} = \mathcal{A}_0$ with 2^n elements, that corresponds to level zero of partitioning. At the first level of partitioning, that is characterized by the index $q = 1$, the set $\mathcal{A}_0$ is subdivided into two disjoint subsets $\mathcal{A}_1(0)$ and $\mathcal{A}_1(1)$ with 2^{n-1} elements each. Let $\Delta_1(0)$ and $\Delta_1(1)$ be the minimum Euclidean distances between elements of the subsets $\mathcal{A}_1(0)$ and $\mathcal{A}_1(1)$, respectively; define Δ_1 as the minimum between the two Euclidean distances $\Delta_1(0)$ and $\Delta_1(1)$; we choose a partition for which Δ_1 is maximum. At the level of partitioning characterized by the index $q > 1$, each of the 2^{q-1} subsets $\mathcal{A}_{q-1}(\ell)$, $\ell = 0, 1, \ldots, 2^{q-1} - 1$, is subdivided into two subsets, thus originating 2^q subsets. During

[2] To emphasize the dependence of the asymptotic coding gain on the choice of the symbol constellations of the coded and uncoded systems, sometimes the information on the considered modulation schemes is included as a subscript in the symbol used to denote the coding gain, e.g. $G_{8PSK/4PSK}$ for the introductory example.

the procedure, it is required that the minimum Euclidean distance at the q-th level of partitioning,

$$\Delta_q = \min_{\ell \in \{0,1,\ldots,2^q-1\}} \Delta_q(\ell) \qquad \text{with} \qquad \Delta_q(\ell) = \min_{\substack{\alpha_i,\, \alpha_m \in \mathcal{A}_q(\ell) \\ \alpha_i \neq \alpha_m}} |\alpha_i - \alpha_m| \tag{12.6}$$

is maximum. At the n-th level of partitioning the subsets $\mathcal{A}_n(\ell)$ consist of only one element each; to subsets with only one element we assign the minimum distance $\Delta_n = \infty$; at the end of the procedure we obtain a tree diagram of binary partitioning for the symbol set. At the q-th level of partitioning, to the two subsets obtained by a subset at the $(q-1)$-th level we assign the binary symbols $y^{(q-1)} = 0$ and $y^{(q-1)} = 1$, respectively; in this manner, an n-tuple of binary symbols $\mathbf{y}_i = (y_i^{(n-1)}, \ldots, y_i^{(1)}, y_i^{(0)})$ is associated with each element α_i found at an end node of the tree diagram.[3]

Therefore the Euclidean distance between two elements of $\mathcal{A}$, α_i and α_m, indicated by the binary vectors $\mathbf{y}_i$ and $\mathbf{y}_m$ that are equal in the first q components, satisfies the relation

$$|\alpha_i - \alpha_m| \geq \Delta_q \quad \text{for} \quad y_i^{(p)} = y_m^{(p)} \qquad p = 0, \ldots, q-1 \qquad i \neq m \tag{12.7}$$

In fact, because of the equality of the components in the positions from (0) up to $(q-1)$, we have that the two elements are in the same subset $\mathcal{A}_q(\ell)$ at the q-th level of partitioning. Therefore their Euclidean distance is at least equal to Δ_q.

Example 12.1.2

The partitioning of the set $\mathcal{A}_0$ of symbols with statistical power $E[|a_k|^2] = 1$ for an 8-PSK system is illustrated in Figure 12.5. The minimum Euclidean distance between elements of the set $\mathcal{A}_0$ is given by $\Delta_0 = 2\sin(\pi/8) = 0.765$. At the first level of partitioning the two subsets $\mathcal{B}_0 = \{(y^{(2)}, y^{(1)}, 0); y^{(i)} = 0, 1\}$ and $\mathcal{B}_1 = \{(y^{(2)}, y^{(1)}, 1); y^{(i)} = 0, 1\}$ are found, with four elements each and minimum Euclidean distance $\Delta_1 = \sqrt{2}$. At the second level of partitioning four subsets $\mathcal{C}_0 = \{(y^{(2)}, 0, 0); y^{(2)} = 0, 1\}$, $\mathcal{C}_2 = \{(y^{(2)}, 1, 0); y^{(2)} = 0, 1\}$, $\mathcal{C}_1 = \{(y^{(2)}, 0, 1); y^{(2)} = 0, 1\}$, and $\mathcal{C}_3 = \{(y^{(2)}, 1, 1); y^{(2)} = 0, 1\}$ are found with two elements each and minimum Euclidean distance $\Delta_2 = 2$. Finally, at the last level eight subsets $D_0, \ldots, D_7$ are found, with one element each and minimum Euclidean distance $\Delta_3 = \infty$.

Example 12.1.3

The partitioning of the set $\mathcal{A}_0$ of symbols with statistical power $E[|a_k|^2] = 1$ for a 16-QAM system is illustrated in Figure 12.6. The minimum Euclidean distance between the elements of $\mathcal{A}_0$ is given by $\Delta_0 = 2/\sqrt{10} = 0.632$. Note that at each successive partitioning level the minimum Euclidean distance among the elements of a subset increases by a factor equal to $\sqrt{2}$. Therefore at the third level of partitioning the minimum Euclidean distance between the elements of each of the subsets D_i, $i = 0, 1, \ldots, 7$, is given by $\Delta_3 = \sqrt{8}\Delta_0$.

[3] For TCM encoders, the n-tuples of binary code symbols will be indicated by $\mathbf{y} = (y^{(n-1)}, \ldots, y^{(0)})$ rather than by the notation $\mathbf{c}$ employed in the previous chapter.

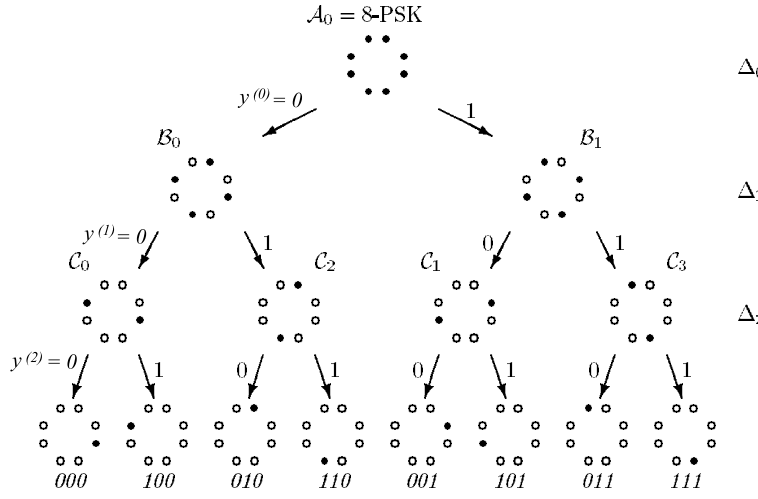


Figure 12.5. Partitioning of the symbol set for an 8-PSK system. [*From Ungerboeck (1982).* © *1982 IEEE.*]

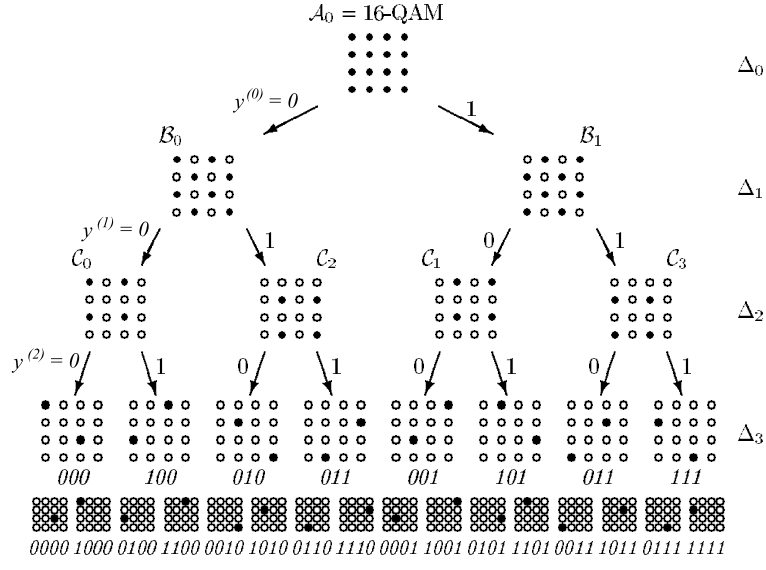


Figure 12.6. Partitioning of the symbol set for a 16-QAM system. [*From Ungerboeck (1982).* © *1982 IEEE.*]

12.1.3 Lattices

Several constellations and the relative partitioning can be effectively described by lattices; furthermore, as we will see in the following sections, the formulation based on lattices is particularly convenient in the discussion on multidimensional trellis codes.

In general, let $\mathbf{Z}_D = \mathbf{Z}^D$, where $\mathbf{Z}$ denotes the set of integers;[4] a lattice Λ in $\Re^D$ is defined by the relation

$$\Lambda = \{(i_1, \ldots, i_D)\,\mathbf{G} \mid (i_1, \ldots, i_D) \in \mathbf{Z}_D\} \tag{12.8}$$

where $\mathbf{G}$ is a non-singular $D \times D$ matrix, *called lattice generator matrix*, by means of which we obtain a correspondence $\mathbf{Z}_D \rightarrow \Lambda$. The vectors given by the rows of $\mathbf{G}$ form a basis for the lattice Λ; the vectors of the basis define a parallelepiped whose volume $V_0 = |\det(\mathbf{G})|$ represents the *characteristic volume* of the lattice. The volume V_0 is equivalent to the volume of a Voronoi cell associated with an element or *point of lattice* Λ and defined as the set of points in $\Re^D$ whose distance from a given point of Λ is smaller than the distance from any other point of Λ. The set of Voronoi cells associated with the points of Λ is equivalent to the space $\Re^D$. A lattice is characterized by two parameters:

1. d_{min}, defined as the minimum distance between points of the lattice;
2. the *kissing number*, defined as the number of lattice points at minimum distance from a given point.

We obtain a subgroup $\Lambda_q(0)$ if points of the lattice Λ are chosen as basis vectors in a matrix $\mathbf{G}_q$, such that they give rise to a characteristic volume $V_q = |\det(\mathbf{G}_q)| > V_0$.

Example 12.1.4 ($\mathbf{Z}_p$ lattice)
In general, as already mentioned, the notation $\mathbf{Z}_p$ is used to define a lattice with an infinite number of points in the p-dimensional Euclidean space with coordinates given by integers. The generator matrix $\mathbf{G}$ for the lattice $\mathbf{Z}_p$ is the $p \times p$ identity matrix; the minimum distance is $d_{min} = 1$ and the kissing number is equal to $2p$. The $\mathbf{Z}_2$ type constellations (see Figure 12.7a) for QAM systems are finite subsets of $\mathbf{Z}_2$, with center at the origin and minimum Euclidean distance equal to Δ_0.

Example 12.1.5 ($\mathbf{D}_n$ lattice)
$\mathbf{D}_n$ is the set of all n-dimensional points whose coordinates are integers that sum to an even number; it may be regarded as a version of the $\mathbf{Z}_n$ lattice from which the points whose coordinates are integers that sum to an odd number were removed. The minimum distance is $d_{min} = \sqrt{2}$ and the kissing number is $2n(n-1)$. The lattice $\mathbf{D}_2$ is represented in Figure 12.7b.

$\mathbf{D}_4$, called the *Schläfli lattice*, constitutes the densest lattice in $\Re^4$; this means that if four-dimensional spheres with centers given by the points of the lattice are used to fill $\Re^4$, then $\mathbf{D}_4$ is the lattice having the largest number of spheres per unit of volume.

Example 12.1.6 ($\mathbf{E}_8$ lattice)
$\mathbf{E}_8$ is given by points

$$\left\{(x_1, \ldots, x_8) \mid \forall i : x_i \in \mathbf{Z} \text{ or } \forall i : x_i \in \mathbf{Z} + \frac{1}{2}, \quad \sum_{i=1}^{8} x_i = 0 \bmod 2\right\} \tag{12.9}$$

[4] In this chapter we use $\mathbf{Z}$ rather than $\mathcal{Z}$ to denote the set of integers.

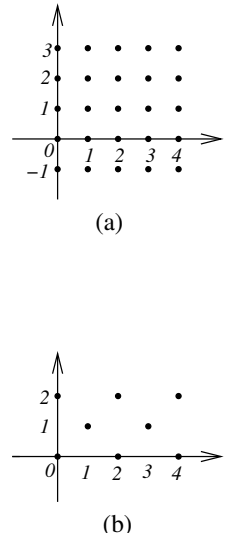


Figure 12.7. (a) $\mathbf{Z}_2$ lattice; (b) $\mathbf{D}_2$ lattice.

In other words $\mathbf{E}_8$ is the set of eight-dimensional points whose components are all integers, or all halves of odd integers, that sum to an even number. $\mathbf{E}_8$ is called the *Gosset lattice*.

We now discuss set partitioning with the aid of lattices. First we recall the properties of subsets obtained by partitioning.

If the set $\mathcal{A}$ has a group structure with respect to a certain operation (see page 844), the partitioning can be done so that the sequence of subsets $\mathcal{A}_0, \mathcal{A}_1(0), \ldots, \mathcal{A}_n(0)$, with $\mathcal{A}_q(0) \subset \mathcal{A}_{q-1}(0)$, form a *chain of subgroups* of $\mathcal{A}_0$; in this case the subsets $\mathcal{A}_q(\ell)$, $\ell \in \{1, \ldots, 2^q - 1\}$, are called *cosets* of the subgroup $\mathcal{A}_q(0)$ with respect to $\mathcal{A}_0$ (see page 837), and are obtained from the subgroup $\mathcal{A}_q(0)$ by translations. The distribution of Euclidean distances between elements of a coset $\mathcal{A}_q(\ell)$ is equal to the distribution of the Euclidean distances between elements of the subgroup $\mathcal{A}_q(0)$, as the "difference" between two elements of a coset yields an element of the subgroup; in particular, for the minimum Euclidean distance in subsets at a certain level of partitioning it holds

$$\Delta_q(\ell) = \Delta_q \qquad \forall \ell \in \{0, \ldots, 2^q - 1\} \tag{12.10}$$

The lattice Λ in $\Re^D$ defined as in (12.8) has *group structure with respect to the addition*. With a suitable translation and normalization, we obtain that the set $\mathcal{A}_0$ for PAM or QAM is represented by a subset of $\mathbf{Z}$ or $\mathbf{Z}_2$. To get a QAM constellation from $\mathbf{Z}_2$, we define, for example, the translated and normalized lattice $\mathbf{Q} = c(\mathbf{Z}_2 + \{1/2, 1/2\})$, where c is an arbitrary scaling factor, generally chosen to normalize the statistical power of the symbols to 1. Figure 12.8 illustrates how QAM constellations are obtained from $\mathbf{Z}_2$.

If we apply binary partitioning to the set $\mathbf{Z}$ or $\mathbf{Z}_2$, we still get infinite lattices in $\Re$ or $\Re^2$, in which the minimum Euclidean distance increases with respect to the original lattice. Formally, we can assign the binary representations of the tree diagram obtained by

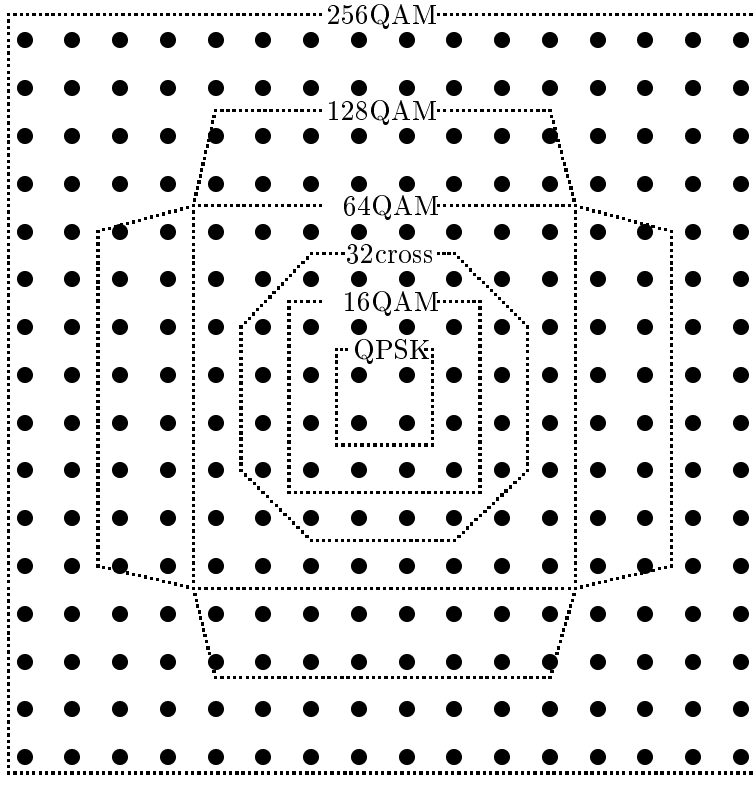


Figure 12.8. The integer lattice $\mathbf{Z}_2$ as template for QAM constellations.

partitioning to the lattices; for transmission, a symbol is chosen as representative for each lattice at an end node of the tree diagram.

Definition 12.1
The notation $\mathbf{X}/\mathbf{X}'$ denotes the set of subsets obtained from the decomposition of the group $\mathbf{X}$ in the subgroup $\mathbf{X}'$ and its cosets. The set $\mathbf{X}/\mathbf{X}'$ forms in turn a group, called the *quotient group* of $\mathbf{X}$ with respect to $\mathbf{X}'$. It is called *binary* if the number of elements is a power of two.

PAM. In this case the subgroups of the lattice $\mathbf{Z}$ are expressed by

$$\mathcal{A}_q(0) = \{2^q i \mid i \in \mathbf{Z}\} = 2^q \mathbf{Z} \tag{12.11}$$

Let

$$t(\ell) = \ell \in \{0, \ldots, 2^q - 1\} \tag{12.12}$$

then the cosets of a subgroup, expressed as

$$\mathcal{A}_q(\ell) = \{a + t(\ell) \mid a \in \mathcal{A}_q(0)\} \tag{12.13}$$

are obtained by translations of the subgroup. The sequence

$$\mathbf{Z} / 2\mathbf{Z} / 2^2\mathbf{Z} / \ \ldots \ / 2^q\mathbf{Z} / \ \ldots \tag{12.14}$$

forms a binary *partitioning chain* of the lattice $\mathbf{Z}$, with increasing minimum distances given by

$$\Delta_q = 2^q \tag{12.15}$$

QAM. The first subgroup in the binary partitioning chain of the two-dimensional lattice $\mathbf{Z}_2$ is a lattice that is obtained from $\mathbf{Z}_2$ by rotation of $\pi/4$ and multiplication by $\sqrt{2}$. The matrix of this linear transformation is

$$\mathbf{R} = \begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \tag{12.16}$$

Successive subgroups in the binary partitioning chain are obtained by repeated application of the linear transformation $\mathbf{R}$,

$$\mathcal{A}_q(0) = \mathbf{Z}_2\mathbf{R}^q \tag{12.17}$$

Let

$$i\mathbf{Z}_2\mathbf{R}^q = \{(ik, im)\mathbf{R}^q \mid k, m \in \mathbf{Z}\} \qquad i \in \mathbf{N} \tag{12.18}$$

where $\mathbf{N}$ is the set of natural numbers, then the sequence

$$\mathbf{Z}_2 / \mathbf{Z}_2\mathbf{R} / \mathbf{Z}_2\mathbf{R}^2 / \mathbf{Z}_2\mathbf{R}^3 / \cdots = \mathbf{Z}_2 / \mathbf{Z}_2\mathbf{R} / 2\mathbf{Z}_2 / 2\mathbf{Z}_2\mathbf{R} / \ \ldots \tag{12.19}$$

forms a binary partitioning chain of the lattice $\mathbf{Z}_2$, with increasing minimum distances given by

$$\Delta_q = 2^{q/2} \tag{12.20}$$

This binary partitioning chain is illustrated in Figure 12.9.

The cosets $\mathcal{A}_q(\ell)$ are obtained by translations of the subgroup $\mathcal{A}_q(0)$ as in the one-dimensional case (12.13), with

$$\mathbf{t}(\ell) = (i, m) \tag{12.21}$$

where, as it can be observed in Figure 12.10 for the case $q = 2$ with $\mathcal{A}_2(0) = 2\mathbf{Z}_2$,

$$\begin{aligned} &i \in \left\{0, \ldots, 2^{\frac{q}{2}} - 1\right\} \qquad m \in \left\{0, \ldots, 2^{\frac{q}{2}} - 1\right\} \qquad \ell = 2^{\frac{q}{2}} i + m \qquad q \text{ even} \\ &i \in \left\{0, \ldots, 2^{\frac{q+1}{2}} - 1\right\} \quad m \in \left\{0, \ldots, 2^{\frac{q-1}{2}} - 1\right\} \quad \ell = 2^{\frac{q-1}{2}} i + m \quad q \text{ odd} \end{aligned} \tag{12.22}$$

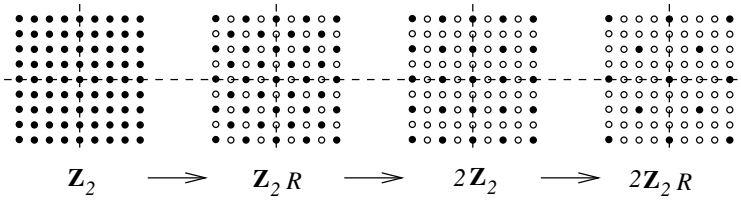


Figure 12.9. Binary partitioning chain of the lattice $\mathbf{Z}_2$.

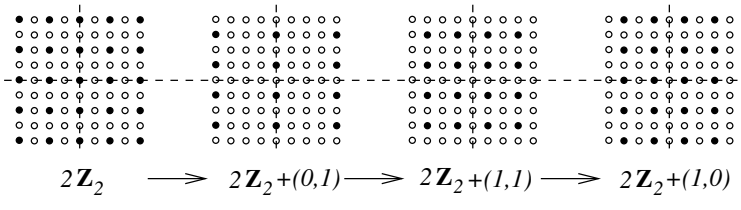


Figure 12.10. The four cosets of $2\mathbf{Z}_2$ in the partition $\mathbf{Z}_2/2\mathbf{Z}_2$.

M-PSK. In this case the set of symbols $\mathcal{A}_0 = \{e^{j2\pi(k/M)} \mid k \in \mathbf{Z}\}$ on the unit circle of the complex plane forms a *group with respect to the multiplication*. If the number of elements M is a power of two, the sequence

$$\mathcal{A}_0 \,/\, \mathcal{A}_1(0) \,/\, \mathcal{A}_2(0) \,/\, \ldots \,/\, \mathcal{A}_{\log_2 M}(0) \qquad \text{with} \qquad \mathcal{A}_q(0) = \left\{ e^{j2\pi \frac{2^q}{M} k} \middle| k \in \mathbf{Z} \right\} \tag{12.23}$$

forms a binary partitioning chain of the set $\mathcal{A}_0$, with increasing minimum distances

$$\Delta_q = \begin{cases} 2\sin\left(\pi \dfrac{2^q}{M}\right) & \text{for } 0 \le q < \log_2 M \\ \infty & \text{for } q = \log_2 M \end{cases} \tag{12.24}$$

The cosets of the subgroups $\mathcal{A}_q(0)$ are given by

$$\mathcal{A}_q(\ell) = \left\{ a\, e^{j2\pi \frac{\ell}{M}} \middle| a \in \mathcal{A}_q(0) \right\} \qquad \ell \in \{0, \ldots, 2^q - 1\} \tag{12.25}$$

12.1.4 Assignment of symbols to the transitions in the trellis

As discussed in the previous subsections, an encoder can be modeled as a finite-state machine with a given number of states and well-defined state transitions. If the encoder input consists of m binary symbols per modulation interval,[5] then there are 2^m possible transitions from a state to the next state; there may be parallel transitions between pairs of states; furthermore for reasons of symmetry we take into consideration only encoders with a uniform structure. After selecting a diagram having the desired characteristics in terms

[5] In this chapter encoding and bit mapping are jointly optimized and m represents the number of information bits L_b per modulation interval (see (6.93)). For example, for QAM the rate of the encoder-modulator is $R_I = \frac{m}{2}$.

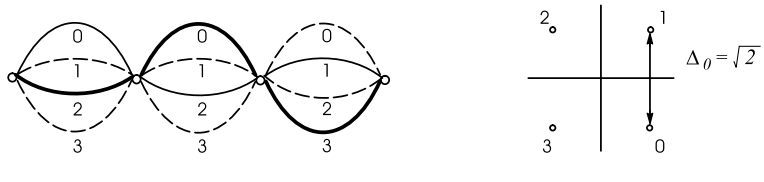
of state transitions, the design of a code is completed by assigning symbols to the state transitions, such that d_{free} is maximum. Following the indications of information theory (see Section 6.10), the symbols are chosen from a redundant set $\mathcal{A}$ of 2^{m+1} elements.

Example 12.1.7 (Uncoded transmission of 2 bits per modulation interval by 8-PSK)
Consider an uncoded 4-PSK system as reference system. The uncoded transmission of 2 bits per modulation interval by 4-PSK can be viewed as the result of the application of a trivial encoder with only one state, and trellis diagram with four parallel transitions, as illustrated in Figure 12.11. A distinct symbol of the alphabet of a 4-PSK system is assigned to each parallel transition. Note from Figure 12.5 that the alphabet of a 4-PSK system is obtained choosing the subset $\mathcal{B}_0$ (or $\mathcal{B}_1$) obtained by partitioning the alphabet $\mathcal{A}_0$ of an 8-PSK system. Therefore $d_{min} = \Delta_1 = \sqrt{2}$. In the trellis diagram, any sequence on the trellis represents a possible symbol sequence. The optimum receiver decides for the symbol of the subset $\mathcal{B}_0$ (or $\mathcal{B}_1$) that is found at the minimum distance from the received signal.

Now consider the two-state trellis diagram of Figure 12.12. The symbols of the subsets $\mathcal{B}_0 = \mathcal{C}_0 \cup \mathcal{C}_2$ and $\mathcal{B}_1 = \mathcal{C}_1 \cup \mathcal{C}_3$ are assigned to the transitions that originate from the first and second state, respectively; this guarantees that the minimum d_{free} between the code symbol sequences is at least equal to that obtained for uncoded 4-PSK system. With a trellis diagram with only two states it is impossible to have only signals of $\mathcal{B}_0$ or $\mathcal{B}_1$ assigned to the transitions that originate from a certain state, and also to all transitions that lead to the same state; therefore we find that the minimum Euclidean distance between code symbol sequences is in this case equal to $d_{free} = \sqrt{\Delta_1^2 + \Delta_0^2} = 1.608$, greater than that obtained for uncoded 4-PSK transmission. As $E_{s,c} = E_{s,u}$ and $\tilde{\Delta}_0 = \Delta_1$, the coding gain is $G_{8PSK/4PSK} = 20\log_{10}(\sqrt{\Delta_1^2 + \Delta_0^2}/\sqrt{\Delta_1^2}) = 1.1$ dB. Furthermore $N_{free} = 2$.

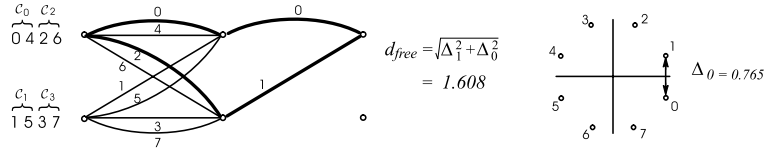
High coding gains for the transmission of 2 bits per modulation interval by 8-PSK are obtained by the codes represented in Figure 12.13, with trellis diagrams having 4, 8, and 16 states. For the heuristic design of these codes with moderate complexity we resort to the following rules proposed by Ungerboeck:

1. all symbols of the set $\mathcal{A}_0$ must be assigned equally likely to the state transitions in the trellis diagram, using criteria of regularity and symmetry;



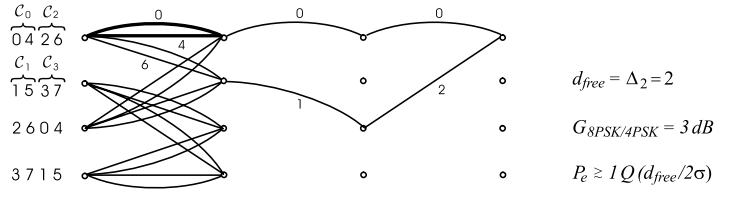
Trellis diagram with a single state. 4-PSK constellation

Figure 12.11. Uncoded transmission of 2 bits per modulation interval by 4-PSK.

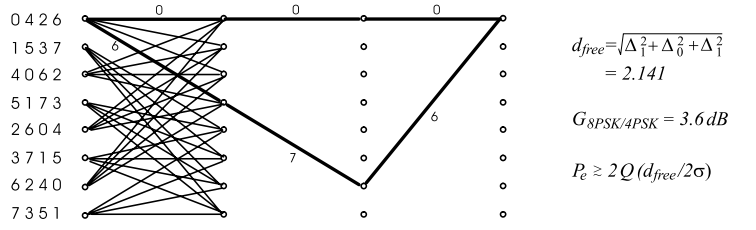


Trellis diagram with two states. 8-PSK constellation

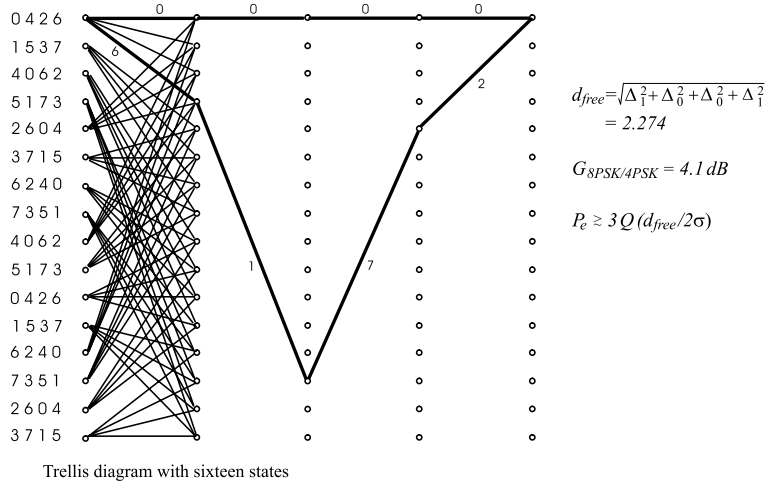
Figure 12.12. Transmission of 2 bits per modulation interval using a two-state trellis code and 8-PSK. For each state the values of the symbols assigned to the transitions that originate from that state are indicated.



Trellis diagram with four states



Trellis diagram with eight states



Trellis diagram with sixteen states

Figure 12.13. Trellis codes with 4, 8, and 16 states for transmission of 2 bits per modulation interval by 8-PSK. [*From Ungerboeck (1982).* © *1982 IEEE.*]

2. to transitions that originate from the same state are assigned symbols of the subset $\mathcal{B}_0$, or symbols of the subset $\mathcal{B}_1$;

3. to transitions that merge to the same state are assigned symbols of the subset $\mathcal{B}_0$, or symbols of the subset $\mathcal{B}_1$;

4. to parallel transitions between two states we assign the symbols of one of the subsets $\mathcal{C}_0$, $\mathcal{C}_1$, $\mathcal{C}_2$, or $\mathcal{C}_3$.

Rule 1 intuitively points to the fact that good trellis codes exhibit a regular structure. Rules 2, 3, and 4 guarantee that the minimum Euclidean distance between code symbol sequences that differ in one or more elements is at least twice the minimum Euclidean distance between uncoded 4-PSK symbols, so that the coding gain is greater than or equal to 3 dB, as we will see in the next examples.

Example 12.1.8 (Four-state trellis code for the transmission of 2 bit/s/Hz by 8-PSK)
Consider the code with four states represented in Figure 12.13. Between each pair of code symbol sequences in the trellis diagram that diverge at a certain state and merge after more than one transition, the Euclidean distance is greater than or equal to $\sqrt{\Delta_1^2 + \Delta_0^2 + \Delta_1^2} = 2.141$. For example, this distance exists between sequences in the trellis diagram labeled by the symbols 0–0–0 and 2–1–2; on the other hand, the Euclidean distance between symbols assigned to parallel transitions is equal to $\Delta_2 = 2$. Therefore the minimum Euclidean distance between code symbol sequences is equal to 2; hence with a four-state trellis code we obtain a gain equal to 3 dB over uncoded 4-PSK transmission. Note that, as the minimum distance d_{free} is determined by parallel transitions, the sequence at minimum distance from a transmitted sequence differs only by one element that corresponds to the transmitted symbol rotated by 180°.

A possible implementation of the encoder/bit-mapper for a four-state trellis code is illustrated in Figure 12.14.

The 4-state trellis code for transmission with spectral efficiency of 2 bit/s/Hz by 8-PSK was described in detail as an introductory example. In Figure 12.13 the values of d_{free} and N_{free} for codes with 4, 8, and 16 states are reported.

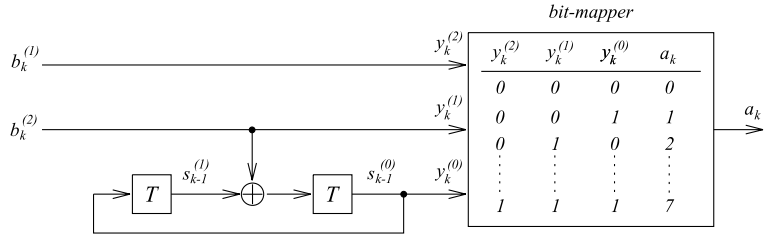


Figure 12.14. Encoder/bit-mapper for a 4-state trellis code for the transmission of 2 bits per modulation interval by 8-PSK.

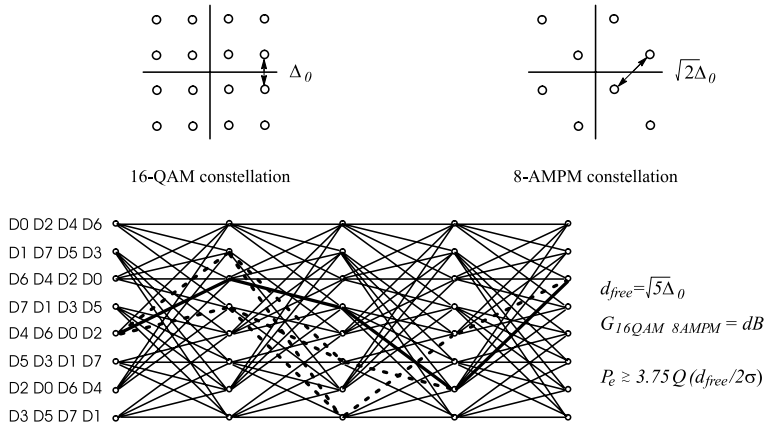


Figure 12.15. Eight-state trellis code for the transmission of 3 bits per modulation interval by 16-QAM.

Consider now a 16-QAM system for the transmission of 3 bits per modulation interval; in this case the reference system uses uncoded 8-PSK or 8-AM-PM, as illustrated in Figure 12.15.

Example 12.1.9 (Eight-state trellis code for the transmission of 3 bit/s/Hz by 16-QAM)
The partitioning of a symbol set with unit statistical power for a 16-QAM system is shown in Figure 12.6. For the assignment of symbols to the transitions on the trellis consider the subsets of the symbol set $\mathcal{A}_0$ denoted by $D_0, D_1, \ldots, D_7$, that contain two elements each. The minimum Euclidean distance between the signals in $\mathcal{A}_0$ is $\Delta_0 = 2/\sqrt{10} = 0.632$; the minimum Euclidean distance between elements of a subset D_i, $i = 0, 1, \ldots, 7$, is $\Delta_3 = \sqrt{8}\Delta_0$.

In the 8-state trellis code illustrated in Figure 12.15, four transitions diverge from each state and four merge to each state. To each transition one of the subsets D_i, $i = 0, 1, \ldots, 7$, is assigned; therefore a transition in the trellis corresponds to a pair of parallel transitions. The assignment of subsets to the transitions satisfies Ungerboeck rules. The subsets D_0, D_4, D_2, D_6, or D_1, D_5, D_3, D_7 are assigned to the four transitions from or to the same state. In evaluating d_{free}, this choice guarantees a squared Euclidean distance equal to at least $2\Delta_0^2$ between sequences that diverge from a state and merge after L transitions, $L > 1$. The squared distance between sequences that diverge from a state and merge after two transitions is equal to $6\Delta_0^2$. If two sequences diverge and merge again after three or more transitions, at least one intermediate transition contributes to an incremental squared Euclidean distance equal to Δ_0^2; thus the minimum Euclidean distance between code symbol sequences that do not differ only for one symbol is given by $\sqrt{5}\Delta_0$. As the Euclidean distance between symbols assigned to parallel transitions is equal to $\sqrt{8}\Delta_0$, the free distance of the code is $d_{free} = \sqrt{5}\Delta_0$. Because the minimum Euclidean distance for an uncoded 8-AM-PM reference system with the same average symbol energy is $\tilde{\Delta}_0 = \sqrt{2}\Delta_0$, the coding gain is $G_{16QAM/8AM\text{-}PM} = 20\log_{10}\{\sqrt{5/2}\} = 4\,\text{dB}$.

In the trellis diagram of Figure 12.15 four paths are shown that represent error events at minimum distance from the code sequence, having symbols taken from the subsets $D_0 - D_0 - D_3 - D_6$; the sequences in error diverge from the same state and merge after three or four transitions. It can be shown that for each code sequence and for each state there are two paths leading to error events of length three and two of length four. The number of code sequences at the minimum distance depends on the code sequence being considered, hence N_{free} represents a mean value. The proof is simple in the case of uncoded 16-QAM, where the number of symbols at the minimum distance from a symbol that is found at the center of the constellation is larger than the number of symbols at the minimum distance from a symbol found at the edge of the constellation; in this case we obtain $N_{free} = 3$. For the eight-state trellis code of Figure 12.15, we get $N_{free} = 3.75$. For constellations of type $\mathbf{Z}_2$ with a number of signals that tends to infinity, the limit is $N_{free} = 4$ for uncoded modulation, and $N_{free} = 16$ for coded modulation with an eight-state code.

12.1.5 General structure of the encoder/bit-mapper

The structure of trellis codes for TCM can be described in general by the combination of a convolutional encoder and a special bit mapper, as illustrated in Figure 12.16. A code symbol sequence is generated as follows. Of the m information bits $[b_k^{(m)}, \ldots, b_k^{(1)}]$ that must be transmitted during a cycle of the encoder/bit-mapper operations, $\tilde{m} \leq m$ are input to a convolutional encoder with rate $R_c = \tilde{m}/(\tilde{m}+1)$; the $\tilde{m}+1$ bits at the encoder output, $\tilde{\mathbf{y}}_k = [y_k^{(\tilde{m})}, \ldots, y_k^{(0)}]^T$, are used to *select one of the* $2^{\tilde{m}+1}$ *subsets* of the symbol set $\mathcal{A}$ with 2^{m+1} elements, and to determine the next state of the encoder. The remaining $(m-\tilde{m})$ uncoded bits *determine which of the* $2^{m-\tilde{m}}$ *symbols* of the selected subset must be transmitted. For example, in the encoder/bit-mapper for a four-state code illustrated in Figure 12.14, the two bits $y_k^{(0)}$ and $y_k^{(1)}$ select one of the four subsets $\mathcal{C}_0, \mathcal{C}_2, \mathcal{C}_1, \mathcal{C}_3$ of the set $\mathcal{A}_0$ with 8 elements. The uncoded bit $y_k^{(2)}$ determines which of the two symbols in the subset $\mathcal{C}_i$ is transmitted.

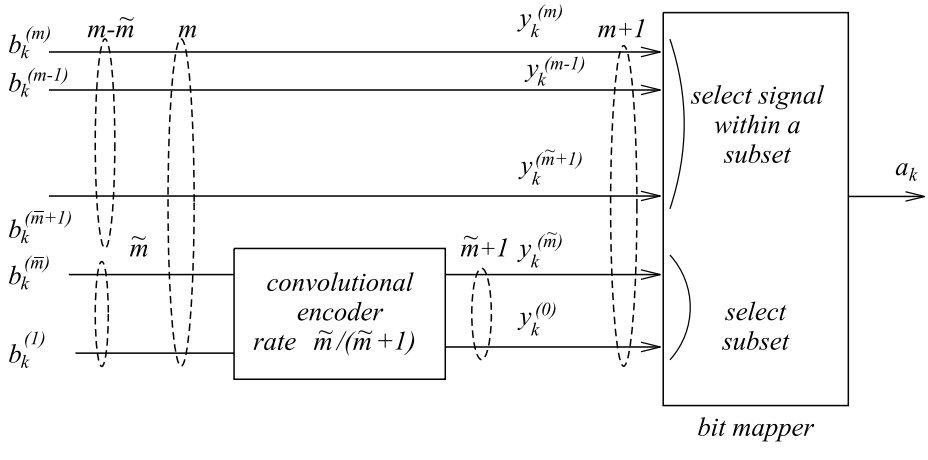


Figure 12.16. General structure of the encoder/bit-mapper for TCM.

Let $\mathbf{y}_k = [y_k^{(m)}, \ldots, y_k^{(1)}, y_k^{(0)}]$ be the $(m+1)$-dimensional binary vector at the input of the bit-mapper at the k-th instant, then the selected symbol is expressed as $a_k = a[\mathbf{y}_k]$. Note that in the trellis the symbols of each subset are associated with $2^{m-\tilde{m}}$ parallel transitions.

The free Euclidean distance of a trellis code, given by (12.2), can be expressed as

$$d_{free} = \min\{\Delta_{\tilde{m}+1}, d_{free}(\tilde{m})\} \tag{12.26}$$

where $\Delta_{\tilde{m}+1}$ is the minimum distance between symbols assigned to parallel transitions and $d_{free}(\tilde{m})$ denotes the minimum distance between code sequences that differ in more than one symbol. In the particular case $\tilde{m} = m$, each subset has only one element and therefore there are no parallel transitions; this occurs, for example, for the encoder/ bit-mapper for an 8-state code illustrated in Figure 12.2.

From Figure 12.16, observe that the vector sequence $\{\tilde{\mathbf{y}}_k\}$ is the output sequence of a convolutional encoder. Recalling (11.263), for a convolutional code with rate $\tilde{m}/(\tilde{m}+1)$ and constraint length ν, we have the following constraints on the bits of the sequence $\{\tilde{\mathbf{y}}_k\}$:

$$\sum_{i=0}^{\tilde{m}} h_\nu^{(i)} y_{k-\nu}^{(i)} \oplus h_{\nu-1}^{(i)} y_{k-\nu+1}^{(i)} \oplus \cdots \oplus h_0^{(i)} y_k^{(i)} = 0 \qquad \forall k \tag{12.27}$$

where $\{h_j^{(i)}\}$, $0 \le j \le \nu$, $0 \le i \le \tilde{m}$, are the parity check binary coefficients of the encoder. For an encoder having ν binary memory cells, a trellis diagram is generated with 2^ν states.

Note that (12.27) defines only the constraints on the code bits, but not the input/output relation of the encoder.

Using polynomial notation, for the binary vector sequence $\mathbf{y}(D)$ (12.27) becomes

$$[y^{(m)}(D), \ldots, y^{(1)}(D), y^{(0)}(D)]\,[h^{(m)}(D), \ldots, h^{(1)}(D), h^{(0)}(D)]^T = 0 \tag{12.28}$$

where $h^{(i)}(D) = h_\nu^{(i)} D^\nu + h_{\nu-1}^{(i)} D^{\nu-1} + \cdots + h_1^{(i)} D + h_0^{(i)}$, for $i = 0, 1, \ldots, \tilde{m}$, and $h^{(i)}(D) = 0$ for $\tilde{m} < i \le m$. From (12.28) we observe that the code sequences $\mathbf{y}(D)$ can be obtained by a systematic encoder with feedback as[6]

$$\begin{bmatrix} y^{(m)}(D) \\ \vdots \\ y^{(1)}(D) \\ y^{(0)}(D) \end{bmatrix} = \begin{bmatrix} & & 0 \\ & & \vdots \\ & & 0 \\ \mathbf{I}_m & & h^{(\tilde{m})}(D)/h^{(0)}(D) \\ & & \vdots \\ & & h^{(1)}(D)/h^{(0)}(D) \end{bmatrix}^T \begin{bmatrix} b^{(m)}(D) \\ \vdots \\ b^{(1)}(D) \end{bmatrix} \tag{12.29}$$

The rational functions $h^{(i)}(D)/h^{(0)}(D)$, $i = 1, \ldots, \tilde{m}$, are realizable if the following condition is satisfied [9]:

$$h_0^{(i)} = h_\nu^{(i)} = \begin{cases} 0 & i \neq 0 \\ 1 & i = 0 \end{cases} \qquad \nu \ge 2 \tag{12.30}$$

---

[6] Note that (12.29) is analogous to (11.304); here the parity check coefficients are used.

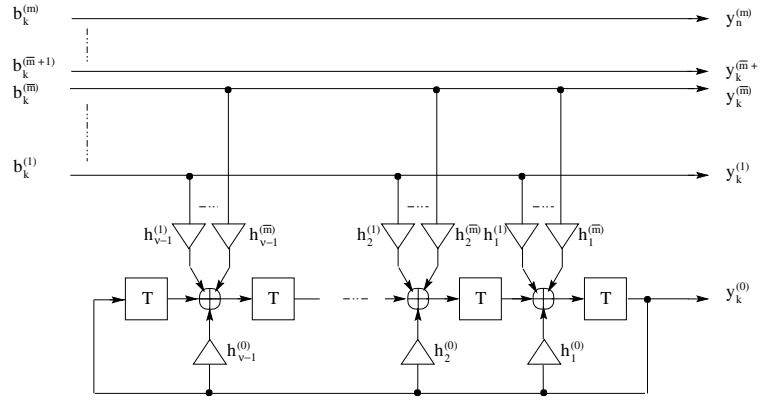


Figure 12.17. Block diagram of a systematic convolutional encoder with feedback. [*From Ungerboeck (1982). © 1982 IEEE.*]

The implementation of a systematic encoder with feedback having ν binary memory elements is illustrated in Figure 12.17.

Computation of d_{free}

Consider the two code sequences

$$\mathbf{y}_1(D) = [y_1^{(m)}(D), \ldots, y_1^{(0)}(D)]^T \tag{12.31}$$

$$\mathbf{y}_2(D) = [y_2^{(m)}(D), \ldots, y_2^{(0)}(D)]^T \tag{12.32}$$

related by $\mathbf{y}_2(D) = \mathbf{y}_1(D) \oplus \mathbf{e}(D)$, as the code is linear. Let $\mathbf{e}_k = [e_k^{(m)}, \ldots, e_k^{(0)}]$, then the error sequence $\mathbf{e}(D)$ is given by

$$\mathbf{e}(D) = \mathbf{e}_k D^k + \mathbf{e}_{k+1} D^{k+1} + \cdots + \mathbf{e}_{k+L} D^{k+L} \tag{12.33}$$

where $\mathbf{e}_i, \mathbf{e}_{i+L} \neq 0$, and $L > 0$. Note that $\mathbf{e}(D)$ is also a valid code sequence as the code is linear. To find a lower bound on the Euclidean distance between the code symbol sequences $a_1(D) = a[\mathbf{y}_1(D)]$ and $a_2(D) = a[\mathbf{y}_2(D)]$ obtained from $\mathbf{y}_1(D)$ and $\mathbf{y}_2(D)$, we define the function $d[\mathbf{e}_k] = \min_{\mathbf{z}_k} d(a[\mathbf{z}_k], a[\mathbf{z}_k \oplus \mathbf{e}_k])$, where minimization takes place in the space of the binary vectors $\mathbf{z}_k = [z_k^{(m)}, \ldots, z_k^{(1)}, z_k^{(0)}]^T$, and $d(\cdot,\cdot)$ denotes the Euclidean distance between specified symbols. For the squared distance between the sequences $a_1(D)$ and $a_2(D)$ then the relation holds

$$\sum_{k=i}^{i+L} d^2(a[\mathbf{y}_k], a[\mathbf{y}_k \oplus \mathbf{e}_k]) \geq \sum_{k=i}^{i+L} d^2[\mathbf{e}_k] = d^2[\mathbf{e}(D)] \tag{12.34}$$

We give the following fundamental theorem [1].

Theorem 12.1
For each sequence $\mathbf{e}(D)$ there exists a pair of symbol sequences $a_1(D)$ and $a_2(D)$ for which relation (12.34) is satisfied with the equal sign.

Proof. Due to the symmetry in the subsets of symbols obtained by partitioning, $d[\mathbf{e}_k] = \min_{\mathbf{z}_k} d(a[\mathbf{z}_k], a[\mathbf{z}_k \oplus \mathbf{e}_k])$ can arbitrarily be obtained by letting the component $z_k^{(0)}$ of the vector $\mathbf{z_k}$ equal to 0 or to 1 and performing the minimization only with respect to components $[z_k^{(m)}, \ldots, z_k^{(1)}]$. As encoding does not impose any constraint on the component sequence $[y_k^{(m)}, \ldots, y_k^{(1)}]$,[7] for each sequence $\mathbf{e}(D)$ a code sequence $\mathbf{y}(D)$ exists such that the relation (12.34) is satisfied as equality for every value of the index k.

The free Euclidean distance between code symbol sequences can therefore be determined by a method similar to that used to find the free Hamming distance between binary code sequences $\mathbf{y}(D)$ (see (11.266)). We need to find an efficient algorithm to examine all possible error sequences $\mathbf{e}(D)$ (12.33) and to substitute the squared Euclidean distance $d^2[\mathbf{e}_k]$ to the Hamming weight of $\mathbf{e}_k$; thus

$$d_{free}^2(\tilde{m}) = \min_{\mathbf{e}(D)=\mathbf{y}_2(D)-\mathbf{y}_1(D)\neq\mathbf{0}} \sum_{k=i}^{i+L} d^2[\mathbf{e}_k] \tag{12.35}$$

Let $q(\mathbf{e}_k)$ be the number of consecutive components equal to zero in the vector $\mathbf{e}_k$, starting with component $e_k^{(0)}$. For example, if $\mathbf{e}_k = [e_k^{(m)}, \ldots, e_k^{(3)}, 1, 0, 0]^T$, then $q(\mathbf{e}_k) = 2$. From the definition of the indices assigned to symbols by partitioning, we obtain that $d[\mathbf{e}_k] \geq \Delta_{q(\mathbf{e}_k)}$; moreover this relation is satisfied as equality for almost all vectors $\mathbf{e}_k$. Note that $d[\mathbf{0}] = \Delta_{q(\mathbf{0})} = 0$. Therefore

$$d_{free}^2(\tilde{m}) \geq \min_{\mathbf{e}(D)\neq\mathbf{0}} \sum_{k=i}^{i+L} \Delta_{q(\mathbf{e}_k)}^2 = \Delta_{free}^2(\tilde{m}) \tag{12.36}$$

If we assume $d_{free}(\tilde{m}) = \Delta_{free}(\tilde{m})$, the risk of committing an error in evaluating the free distance of the code is low, as the minimum is usually reached by more than one error sequence. By the definition of free distance in terms of the minimum distances between elements of the subsets of the symbol set, the computation of $d_{free}(\tilde{m})$ will be independent of the particular assignment of the symbols to the binary vectors with $(m+1)$ components, provided that the values of the minimum distances among elements of the subsets are not changed.

At this point it is possible to identify a further important consequence of the constraint (12.30) on the binary coefficients of the systematic convolutional encoder. We can show that an error sequence $\mathbf{e}(D)$ begins with $\mathbf{e}_i = (e_i^{(m)}, \ldots, e_i^{(1)}, 0)$ and ends with $\mathbf{e}_{i+L} = (e_{i+L}^{(m)}, \ldots, e_{i+L}^{(1)}, 0)$. It is therefore guaranteed that all transitions that originate from the same state and to the transitions that merge at the same state are assigned signals of the subset $\mathcal{B}_0$ or those of the subset $\mathcal{B}_1$. The squared Euclidean distance associated with an error sequence is therefore greater than or equal to $2\Delta_1^2$. The constraint on the parity check

[7] From the parity equation (12.29) we observe that a code sequence $\{\mathbf{z}_k\}$ can have arbitrary values for each m-tuple $[z_k^{(m)}, \ldots, z_k^{(1)}]$.

coefficients allows us, however, to determine only a lower bound for $d_{free}(\tilde{m})$. For a given sequence $\Delta_0 \leq \Delta_1 \leq \cdots \leq \Delta_{\tilde{m}+1}$ of minimum distances between elements of subsets and a code with constraint length ν, a convolutional code that yields the maximum value of $d_{free}(\tilde{m})$ is usually found by a computer program for code search. The search of the $(\nu-1)(\tilde{m}+1)$ parity check binary coefficients is performed by means such that the explicit computation of $d_{free}(\tilde{m})$ is often avoided.

Tables 12.1 and 12.2 report the optimum codes for TCM with symbols of the type $\mathbf{Z}_1$ and $\mathbf{Z}_2$, respectively [2]. For 8-PSK, the optimum codes are given in Table 12.3 [2].

Table 12.1 Codes for one-dimensional modulation. [*From Ungerboeck (1987). © 1987 IEEE.*]

2^ν	$\tilde{m}$	$\mathbf{h^1}$	$\mathbf{h^0}$	$d^2_{free}/\tilde{\Delta}^2_0$	$G_{4AM/2AM}$ $(m=1)$	$G_{8AM/4AM}$ $(m=2)$	G_{code} $(m\to\infty)$	N_{free} $(m\to\infty)$
4	1	2	5	9.0	2.55	3.31	3.52	4
8	1	04	13	10.0	3.01	3.77	3.97	4
16	1	04	23	11.0	3.42	4.18	4.39	8
32	1	10	45	13.0	4.15	4.91	5.11	12
64	1	024	103	14.0	4.47	5.23	5.44	36
128	1	126	235	16.0	5.05	5.81	6.02	66

Table 12.2 Codes for two-dimensional modulation. [*From Ungerboeck (1987). © 1987 IEEE.*]

2^ν	$\tilde{m}$	$\mathbf{h^2}$	$\mathbf{h^1}$	$\mathbf{h^0}$	$d^2_{free}/\tilde{\Delta}^2_0$	$G_{16QAM/8PSK}$ $(m=3)$	$G_{32QAM/16QAM}$ $(m=4)$	$G_{64QAM/32QAM}$ $(m=5)$	G_{code} $(m\to\infty)$	N_{free} $(m\to\infty)$
4	1	—	2	5	4.0*	4.36	3.01	2.80	3.01	4
8	2	04	02	11	5.0	5.33	3.98	3.77	3.98	16
16	2	16	04	23	6.0	6.12	4.77	4.56	4.77	56
32	2	10	06	41	6.0	6.12	4.77	4.56	4.77	16
64	2	064	016	101	7.0	6.79	5.44	5.23	5.44	56
128	2	042	014	203	8.0	7.37	6.02	5.81	6.02	344
256	2	304	056	401	8.0	7.37	6.02	5.81	6.02	44
512	2	0510	0346	1001	8.0*	7.37	6.02	5.81	6.02	4

Table 12.3 Codes for 8-PSK. [*From Ungerboeck (1987). © 1987 IEEE.*]

2^ν	$\tilde{m}$	$\mathbf{h^2}$	$\mathbf{h^1}$	$\mathbf{h^0}$	$d^2_{free}/\tilde{\Delta}^2_0$	$G_{8PSK/4PSK}$ $(m=2)$	N_{free}
4	1	—	2	5	4.0*	3.01	1
8	2	04	02	11	4.586	3.60	2
16	2	16	04	23	5.172	4.13	$\simeq 2.3$
32	2	34	16	45	5.758	4.59	4
64	2	066	030	103	6.343	5.01	$\simeq 5.3$
128	2	122	054	277	6.586	5.17	$\simeq 0.5$

Parity check coefficients are specified in octal notation; for example, the binary vector $[h_6^{(0)}, \ldots, h_0^{(0)}] = [1, 0, 0, 0, 1, 0, 1]$ is represented by $\mathbf{h}^{(0)} = 105_8$. In the tables, an asterisk next to the value d_{free} indicates that the free distance is determined by the parallel transitions, that is $d_{free}(\tilde{m}) > \Delta_{\tilde{m}+1}$.

12.2 Multidimensional TCM

So far we have dealt with trellis coded modulation schemes that use two-dimensional (2D) constellations, that is to send m information bits per modulation interval we employ a 2D constellation of $2^{(m+1)}$ points; the intrinsic cost is represented by doubling the cardinality of the 2D constellation with respect to uncoded schemes, as a bit of redundancy is generated at each modulation interval. Consequently the minimum distance within points of the constellation is reduced, for the same average power of the transmitted signal; without this cost the coding gain would be 3 dB higher.

An advantage of using a multidimensional constellation to generate code symbol sequences is that doubling the cardinality of a constellation does not lead to a 3 dB loss; in other words, the signals are spaced by a larger Euclidean distance d_{min} and therefore the margin against noise is increased.

A simple way to generate a multidimensional constellation is obtained by time division. If, for example, ℓ two-dimensional symbols are transmitted over a time interval of duration T_s, and each of them has a duration T_s/ℓ, we may regard the ℓ 2D symbols as an element of a 2ℓ-dimensional constellation. Therefore in practice multidimensional signals can be transmitted as sequences of one or two-dimensional symbols. An example of multidimensional signaling using binary PAM transmission is illustrated in Section 6.8.

In this section we describe the construction of 2ℓ-dimensional TCM schemes for the transmission of m bits per 2D symbol, and thus $m\ell$ bits per 2ℓ-dimensional signal. We maintain the principle of using a redundant signal set, with a number of elements doubled with respect to that used for uncoded modulation; therefore the 2ℓ-dimensional TCM schemes use sets of $2^{m\ell+1}$ 2ℓ-dimensional signals. With respect to two-dimensional TCM schemes, this implies a lower redundancy in the two-dimensional component sets. For example, in the 4D case doubling the number of elements causes an expansion of the 2D component constellations by a factor $\sqrt{2}$; this corresponds to a half bit of redundancy per 2D component constellation. The cost of the expansion of the signal set is reduced by 1.5 dB in the 4D case and by 0.75 dB in the 8D case. A further advantage of multidimensional constellations is that the design of schemes invariant to phase rotation is simplified (see Section 12.3).

Encoding

Starting with a constellation $\mathcal{A}_0$ of the type $\mathbf{Z}_1$ or $\mathbf{Z}_2$ in the one or two-dimensional signal space, we consider multidimensional trellis codes where the signals to be assigned to the transitions in the trellis diagram come from a constellation $\mathcal{A}_I^0$, $I > 2$, in the multidimensional space $\Re^I$. In practice, if $(m + 1)$ binary output symbols of the finite state encoder determine the assignment of modulation signals, it is possible to associate with these binary symbols a sequence of ℓ modulation signals, each in the constellation $\mathcal{A}_0$, transmitted during ℓ modulation intervals, each of duration T; this sequence can be

considered as an element of the space $\Re^I$, as signals transmitted in different modulation intervals are assumed orthogonal. If we have a constellation $\mathcal{A}_0$ with M elements, the relation $M^\ell \geq 2^{m+1}$ holds.

Possible sequences of ℓ symbols of a constellation $\mathcal{A}_0$ give origin to a block code B of length ℓ in the space $\Re^I$. Hence, we can consider the multidimensional TCM as an encoding method in which the binary output symbols of the finite state encoder represent the information symbols of the block code. Part of the whole coding gain is obtained by choosing a multidimensional constellation $\mathcal{A}_I^0$ with a large minimum Euclidean distance, that is equivalent to the choice of an adequate block code; therefore it is necessary to identify block codes in the Euclidean space $\Re^I$ that admit a partitioning of code sequences in subsets such that the minimum Euclidean distance among elements of a subset is the largest possible. For a linear block code, it can be shown that this partitioning yields as subsets a subgroup of the code and its cosets. Consider the trellis diagram of a code for the multidimensional TCM where each state has $2^{\tilde{m}+1}$ transitions to adjacent states, and between pairs of adjacent states there exist $2^{m-\tilde{m}}$ parallel transitions. Then the construction of the trellis code requires that the constellation $\mathcal{A}_I^0$ is partitioned into $2^{\tilde{m}+1}$ block codes $B_{\tilde{m}+1}^j$, $j \in \{0, \ldots, 2^{\tilde{m}+1} - 1\}$, each of length ℓ and *rate* equal to $(m - \tilde{m})/I$ bits/dimension. In any period equivalent to ℓ modulation intervals, $(\tilde{m} + 1)$ binary output symbols of the finite state encoder select one of the code blocks, and the remaining $(m - \tilde{m})$ binary symbols determine the sequence to be transmitted among those belonging to the selected code. From (6.103), for a constellation $\mathcal{A}_0$ of type $\mathbf{Z}_2$, $B_{min} T_s = \ell$, and $M^\ell = 2^{\ell m+1}$, the spectral efficiency of a system with multidimensional TCM is equal to

$$\nu = \frac{m}{\ell} = \frac{\ell \log_2 M - 1}{\ell} = \log_2 M - \frac{1}{\ell} \quad \text{bit/s/Hz} \tag{12.37}$$

The optimum constellation in the space $\Re^I$ is obtained by solving the problem of finding a lattice such that, given the minimum Euclidean distance d_{min} between two points, the number of points per unit of volume is maximum; for $I = 2$, the solution is given by the hexagonal lattice. For a number of constellation points $M \gg 1$, the ratio between the statistical power of signals of a constellation of the type $\mathbf{Z}_2$ and that of signals of a constellation chosen as a subset of the hexagonal lattice is equal to 0.62 dB; hence, a constellation subset of the hexagonal lattice yields a "coding gain" equal to 0.62 dB with respect to a constellation of the type $\mathbf{Z}_2$.

For $I = 4$ and $I = 8$ the solutions are given by the Schläfli lattice $\mathbf{D}_4$ and the Gosset lattice $\mathbf{E}_8$, respectively, defined in Examples 12.1.5 and 12.1.6 of page 976. Note that to an increase in the number of dimensions and in the density of the optimum lattice also corresponds an increase in the number of lattice points with minimum distance from a given point.

To design codes with a set of modulation signals whose elements are represented by symbol sequences, we have to address the problem of partitioning a lattice in a multidimensional space.

In the multidimensional TCM, if $\tilde{m} + 1$ binary output symbols of the finite state encoder determine the next state to a given state, the lattice Λ must be partitioned into $2^{\tilde{m}+1}$ subsets $\Lambda_{\tilde{m}+1}(j)$, $j = 0, \ldots, 2^{\tilde{m}+1} - 1$, such that the minimum Euclidean distance $\Delta_{\tilde{m}+1}$ between the elements of each subset is maximum; hence, the problem consists in determining the subsets of Λ so that the density of the points of $\Lambda_{\tilde{m}+1}(j)$, $j = 0, \ldots, 2^{\tilde{m}+1} - 1$, is maximum.

In the case of an I-dimensional lattice Λ that can be expressed as the Cartesian product of ℓ terms all equal to a lattice in the space $\Re^{I/\ell}$, the partitioning of Λ can be derived from the partitioning of the I/ℓ-dimensional lattice.

Example 12.2.1 (Partitioning of the lattice $\mathbf{Z}_4$*)*
The lattice $\mathcal{A}_4^0 = \mathbf{Z}_4$ can be expressed in terms of the Cartesian product of the lattice $\mathbf{Z}_2$ with itself, $\mathbf{Z}_4 = \mathbf{Z}_2 \otimes \mathbf{Z}_2$; the subsets of the lattice $\mathcal{A}_4^0$ are therefore characterized by two sets of signals belonging to a $\mathbf{Z}_2$ lattice and by their subsets. The partitioning of a signal set belonging to a $\mathbf{Z}_2$ lattice is represented in Figure 12.6. Thus

$$\begin{aligned} \mathbf{Z}_4 = \mathcal{A}_4^0 &= \mathcal{A}_0 \otimes \mathcal{A}_0 = (\mathcal{B}_0 \cup \mathcal{B}_1) \otimes (\mathcal{B}_0 \cup \mathcal{B}_1) \\ &= (\mathcal{B}_0 \otimes \mathcal{B}_0) \cup (\mathcal{B}_0 \otimes \mathcal{B}_1) \cup (\mathcal{B}_1 \otimes \mathcal{B}_0) \cup (\mathcal{B}_1 \otimes \mathcal{B}_1) \end{aligned} \tag{12.38}$$

At the first level of partitioning the two optimum subsets are

$$\mathcal{B}_4^0 = (\mathcal{B}_0 \otimes \mathcal{B}_0) \cup (\mathcal{B}_1 \otimes \mathcal{B}_1) \tag{12.39}$$

$$\mathcal{B}_4^1 = (\mathcal{B}_0 \otimes \mathcal{B}_1) \cup (\mathcal{B}_1 \otimes \mathcal{B}_0) \tag{12.40}$$

In terms of the four-dimensional TCM, the assignment of pairs of two-dimensional modulation signals to the transitions in the trellis diagram is such that in the first modulation interval all the points of the component QAM constellation are admissible. If two pairs assigned to adjacent transitions are such that the signals in the first modulation interval are separated by the minimum Euclidean distance Δ_0, the signals in the second modulation interval will also have Euclidean distance at least equal to Δ_0; in this way the minimum Euclidean distance among points of $\mathcal{B}_4^0$ or $\mathcal{B}_4^1$ is equal to $\Delta_1 = \sqrt{2}\Delta_0$. The subset $\mathcal{B}_4^0$ is the Schläfli lattice $\mathbf{D}_4$, that is the densest lattice in the space $\Re^4$; the subset $\mathcal{B}_4^1$ is instead the coset of $\mathbf{D}_4$ with respect to $\mathbf{Z}_4$. The next partitioning in the four subsets

$$\mathcal{B}_0 \otimes \mathcal{B}_0 \qquad \mathcal{B}_1 \otimes \mathcal{B}_1 \qquad \mathcal{B}_0 \otimes \mathcal{B}_1 \qquad \mathcal{B}_0 \otimes \mathcal{B}_1 \tag{12.41}$$

does not yield any increase in the minimum Euclidean distance. Note that the four subsets at the second level of partitioning differ from the $\mathbf{Z}_4$ lattice only by the position with respect to the origin, direction, and scale; therefore the subsets at successive levels of partitioning are obtained by iterating the same procedure described for the first two. Thus we have the following partitioning chain

$$\mathbf{Z}_4 \,/\, \mathbf{D}_4 \,/\, (\mathbf{Z}_2\mathbf{R})^2 \,/\, 4\mathbf{D}_4 \,/\, \ldots \tag{12.42}$$

where $\mathbf{R}$ is the 2×2 matrix given by (12.16). The partitioning of the $\mathbf{Z}_4$ lattice is illustrated in Figure 12.18 [2].

Optimum codes for the multidimensional TCM are found in a similar manner to that described for the one- and two-dimensional TCM codes. Codes and relative asymptotic

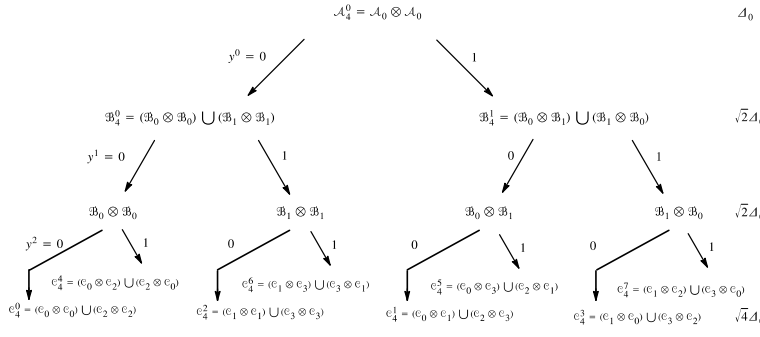


Figure 12.18. Partitioning of the lattice $\mathcal{A}_4^0 = \mathbf{Z}_4$. [*From Ungerboeck (1987). © 1987 IEEE.*]

Table 12.4 Codes for four-dimensional modulation. [*From Ungerboeck (1987). © 1987 IEEE.*]

2^ν	$\tilde{m}$	$\mathbf{h}^4$	$\mathbf{h}^3$	$\mathbf{h}^2$	$\mathbf{h}^1$	$\mathbf{h}^0$	$d^2_{free}/\tilde{\Delta}_0^2$	G_{code} $(m \to \infty)$	N_{free} $(m \to \infty)$
8	2	—	—	04	02	11	4.0	4.52	88
16	2	—	—	14	02	21	4.0	4.52	24
32	3	—	30	14	02	41	4.0	4.52	8
64	4	050	030	014	002	101	5.0	5.48	144
128	4	120	050	022	006	203	6.0	6.28	—

coding gains for the four-dimensional TCM, with respect to uncoded modulation with signals of the type $\mathbf{Z}_2$, are reported in Table 12.4 [2]. These gains are obtained for signal sets with a large number of elements that, in the signal space, take up the same volume of elements of the signal set used for uncoded modulation; then the comparison is made for the same statistical power and the same peak power of the two-dimensional signals utilized for uncoded modulation.

Decoding

The decoding of signals generated by multidimensional TCM is achieved by a sequence of operations that is the inverse with respect to the encoding procedure described in the previous subsection. The first stage of decoding consists in determining, for each modulation interval, the Euclidean distance between the received sample and all M signals of the constellation $\mathcal{A}_0$ of symbols and also, within each subset $\mathcal{A}_q(i)$ of the constellation $\mathcal{A}_0$, the signal $\hat{a}_k(i)$ that has the minimum Euclidean distance from the received sample. The second stage consists in the decoding, by a maximum likelihood decoder, of $2^{\tilde{m}+1}$ block codes $B_{\tilde{m}+1}(\ell')$. Due to the large number M^ℓ of signals in the multidimensional space $\Re^I$, in general the block codes have a number of elements such that the complexity of a maximum likelihood decoder that should compute the metric for each element would result excessive. The task of the decoder can be greatly simplified thanks to the method followed

for the construction of block codes $B_{\tilde{m}+1}(\ell')$. Block codes are identified by the subsets of the multidimensional constellation $\mathcal{A}_I^0$, which are expressed in terms of the Cartesian product of subsets $\mathcal{A}_q(i)$ of the one- or two-dimensional constellation $\mathcal{A}_0$. Decoding of the block codes is jointly carried out by a trellis diagram defined on a finite number ℓ of modulation intervals, where Cartesian products of subsets of the constellation $\mathcal{A}_0$ in different modulation intervals are represented as sequences of branches, and the union of subsets as the union of branches. Figure 12.19 illustrates the trellis diagram for the decoding of block codes obtained by the partitioning into two, four, and eight subsets of a constellation $\mathcal{A}_4^0 \subset \mathbf{Z}_4$.

As the decisions taken in different modulation intervals are independent, the branch metrics of the trellis diagram is the squared Euclidean distance $d^2(z_k, \hat{a}_k(i))$ between the received sample during the k-th modulation interval and the signal $\hat{a}_k(i)$ of the subset $\mathcal{A}_q(i)$ at the minimum Euclidean distance from z_k. The maximum likelihood decoder for block codes can then be implemented by the Viterbi algorithm, which is applied for ℓ iterations from the initial node of the trellis diagram to the $2^{\tilde{m}+1}$ terminal nodes; the procedure to decode block codes by a trellis diagram is due to Wolf [10]. The third and last decoding stage consists in using the metrics, obtained at the $2^{\tilde{m}+1}$ terminal nodes of the trellis diagram for the decoding of the block codes, in the Viterbi decoder for the trellis code, that yields the sequence of detected binary vectors $(\hat{b}_k^{(m)}, \ldots, \hat{b}_k^{(1)})$.

To evaluate the complexity of each iteration in multidimensional TCM decoding, it is necessary to consider the overall number of branches in the trellis diagrams considered at the different decoding stages; this number includes the ℓM branches for decisions on signals within the subsets $\mathcal{A}_q(i)$ in ℓ modulation intervals, the N_R branches in the trellis diagram for the decoding of the block codes, and the $2^{\nu+\tilde{m}}$ branches in the trellis diagram

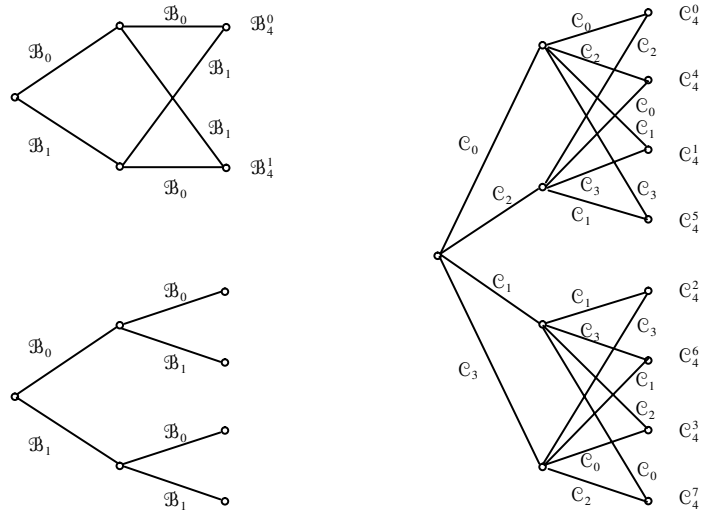


Figure 12.19. Trellis diagram for the decoding of block codes obtained by the partitioning the lattice $\mathcal{A}_4^0 = \mathbf{Z}_4$ into two, four, and eight subsets.

for the decoding of the trellis code. For a code with efficiency equal to $(\log_2 M - 1/\ell)$ bits per modulation interval, the complexity expressed as the number of branches in the trellis diagram per transmitted information bit is thus given by

$$\frac{\ell M + N_R + 2^{\nu+\tilde{m}}}{\ell \log_2 M - 1} \tag{12.43}$$

The number N_R can be computed on the basis of the particular choice of the multidimensional constellation partitioning chain; for example, in the case of partitioning of the lattice $\mathbf{Z}_4$ into four or eight subsets, for the four-dimensional TCM we obtain $N_R = 20$. Whereas in the case of two-dimensional TCM the decoding complexity essentially lies in the implementation of the Viterbi algorithm to decode the trellis code, in the four-dimensional TCM most of the complexity is due to the decoding of the block codes.

In multidimensional TCM it is common to find codes with a large number N_{free} of error events at the minimum Euclidean distance; this characteristic is due to the fact that in dense multidimensional lattices the number of points at the minimum Euclidean distance rapidly increases with the increase in the number of dimensions. In this case the minimum Euclidean distance is not sufficient to completely characterize code performance, as the difference between asymptotic coding gain and effective coding gain, for values of interest of the error probability, cannot be neglected.

12.3 Rotationally invariant TCM schemes

In PSK or QAM transmission schemes with coherent demodulation, an ambiguity occurs at the receiver whenever we ignore the value of the carrier phase used at the transmitter. In BPSK, for example, there are two equilibrium values of the carrier phase synchronization system at the receiver, corresponding to a rotation of the phase equal to 0° and 180°, respectively; if the rotation of the phase is equal to 180°, the binary symbols at the demodulator output come out inverted. In QAM, instead, because of the symmetry of the $\mathbf{Z}_2$ lattice points, there are four possible equilibrium values. The solution to the problem is represented by the insertion in the transmitted sequence of symbols for the synchronization, or by differential encoding (see Section 6.5.2). We recall that with differential encoding the information symbols are assigned to the phase difference between consecutive elements in the symbol sequence; in this way, the absolute value of phase in the signal space becomes irrelevant to the receiver.

With TCM we obtain code signal sequences that, in general, do not present symmetries with respect to the phase rotation in the signal space. This means that, for a code symbol sequence, after a phase rotation determined by the phase synchronization system, we may obtain a sequence that does not belong to the code; therefore trellis coding not invariant to phase rotation can be seen as a method for the construction of signal sequences that allow the recovery of the absolute value of the carrier phase. Then, for the demodulation, we choose the value of the carrier phase corresponding to an equilibrium value of the synchronization system, for which the Euclidean distance between the received sequence and the code sequence is minimum.

For fast carrier phase synchronization, various rotationally invariant trellis codes have been developed; these codes are characterized by the property that code signal sequences

continue to belong to the code even after the largest possible number of phase rotations. In this case, with differential encoding of the information symbols, the independence of demodulation and decoding from the carrier phase recovered by the synchronization system is guaranteed. A differential decoder is then applied to the sequence of binary symbols $\{\tilde{b}_\ell\}$ at the output of the decoder for the trellis code to obtain the desired detection of the sequence of information symbols $\{\hat{b}_\ell\}$. Rotationally invariant trellis codes were initially proposed by Wei [11, 12]. In general, the invariance to phase rotation is more easily obtained with multidimensional TCM; in the case of two-dimensional TCM for PSK or QAM systems, it is necessary to use non-linear codes in GF(2).

In the case of TCM for PSK systems, the invariance to phase rotation can be directly obtained using PSK signals with differential encoding. The elements of the symbol sets are not assigned to the binary vectors of the convolutional code but rather to the phase differences relative to the previous symbols.

Bibliography

[1] G. Ungerboeck, "Channel coding with multilevel/phase signals", *IEEE Trans. on Information Theory*, vol. 28, pp. 55–67, Jan. 1982.

[2] G. Ungerboeck, "Trellis coded modulation with redundant signal sets. Part I and Part II", *IEEE Communications Magazine*, vol. 25, pp. 6–21, Feb. 1987.

[3] S. S. Pietrobon, R. H. Deng, A. Lafanechere, G. Ungerboeck, and D. J. Costello Jr., "Trellis-coded multidimensional phase modulation", *IEEE Trans. on Information Theory*, vol. 36, pp. 63–89, Jan. 1990.

[4] E. Biglieri, D. Divsalar, P. J. McLane, and M. K. Simon, *Introduction to trellis-coded modulation with applications*. New York: Macmillan Publishing Company, 1991.

[5] S. S. Pietrobon and D. J. Costello Jr., "Trellis coding with multidimensional QAM signal sets", *IEEE Trans. on Information Theory*, vol. 39, pp. 325–336, Mar. 1993.

[6] J. Huber, *Trelliscodierung*. Heidelberg, Germany: Springer-Verlag, 1992.

[7] C. Schlegel, *Trellis coding*. New York: IEEE Press, 1997.

[8] G. D. Forney, Jr. and G. Ungerboeck, "Modulation and coding for linear Gaussian channels", *IEEE Trans. on Information Theory*, vol. 44, pp. 2384–2415, Oct. 1998.

[9] G. D. Forney, Jr., "Convolutional codes I: algebraic structure", *IEEE Trans. on Information Theory*, vol. IT–16, pp. 720–738, Nov. 1970.

[10] J. K. Wolf, "Efficient maximum likelihood decoding of linear block codes using a trellis", *IEEE Trans. on Information Theory*, vol. IT–24, pp. 76–80, Jan. 1978.

[11] L. F. Wei, "Trellis-coded modulation with multidimensional constellations", *IEEE Trans. on Information Theory*, vol. 33, pp. 483–501, July 1987.

[12] L. F. Wei, "Rotationally invariant trellis-coded modulations with multidimensional M-PSK", *IEEE Journal on Selected Areas in Communications*, vol. 7, pp. 1281–1295, Dec. 1989.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 13

Precoding and coding techniques for dispersive channels

In this chapter we first extend the study of the capacity of ideal AWGN channels introduced in Section 6.10 to the case of band limited dispersive channels with additive Gaussian noise. We find that the optimum power spectral density of the transmitted signal is inversely proportional to the spectral signal-to-noise ratio, and is determined by a *water pouring* criterion. Then we discuss practical methods to approximate this capacity. In particular, we consider OFDM or multicarrier modulation described in Chapter 9, as well as single-carrier modulation in the form of CAP or QAM as described in Chapter 7, combined with joint precoding and coding techniques.

13.1 Capacity of a dispersive channel

In general, a non-ideal channel (linear dispersive channel) with additive Gaussian noise is characterized as a transmission medium with impulse response $g_{Ch}(t)$ and additive Gaussian noise $w(t)$ having power spectral density $\mathcal{P}_w(f)$. Let $s(t)$ denote the channel input signal; then the receiver input signal is given by $r(t) = s * g_{Ch}(t) + w(t)$. In practice, the transmitted signal $s(t)$ must satisfy a constraint on the statistical power expressed as

$$\int_0^{+\infty} \mathcal{P}_s(f)\, df \leq V_P \tag{13.1}$$

where $\mathcal{P}_s(f)$ is the power spectral density of $s(t)$.

The signal-to-noise ratio of the channel as a function of frequency is defined as

$$\Gamma_{Ch}(f) = \frac{|\mathcal{G}_{Ch}(f)|^2}{\mathcal{P}_w(f)} \tag{13.2}$$

The transmission passband $\mathcal{B}$, defined in (13.8), is intuitively given by the interval of frequencies characterized by large values of $\Gamma_{Ch}(f)$.

First, we want to determine an expression of the capacity $C_{[b/s]}$ that extends (6.280), valid for an ideal AWGN channel, to the case of a dispersive channel. To this end we divide the passband $\mathcal{B}$, having measure B, into N sub-bands $\mathcal{B}_i$, $i = 1, \ldots, N$, of width $\Delta f = B/N$, where Δf is chosen sufficiently small so that $\mathcal{P}_s(f)$ and $\Gamma_{Ch}(f)$ are, to a first approximation, equal to constants within the generic sub-band $\mathcal{B}_i$, that is we assume $\exists f_i \in \mathcal{B}_i$ such that $\mathcal{P}_s(f) = \mathcal{P}_s(f_i)$ and $\Gamma_{Ch}(f) = \Gamma_{Ch}(f_i)$, $\forall f \in \mathcal{B}_i$.

The subchannel i has a capacity $C_{[b/s],i}$ given by (6.280),

$$C_{[b/s],i} = \Delta f \log_2 \left[1 + \frac{\Delta f \, \mathcal{P}_s(f_i) \, |\mathcal{G}_{Ch}(f_i)|^2}{\Delta f \, \mathcal{P}_w(f_i)} \right] \tag{13.3}$$

The total capacity $C_{[b/s]}$ is obtained by summing the terms $C_{[b/s],i}$, that is

$$\sum_{i=1}^{N} C_{[b/s],i} = \Delta f \sum_{i=1}^{N} \log_2 \left[1 + \frac{\Delta f \, \mathcal{P}_s(f_i) \, |\mathcal{G}_{Ch}(f_i)|^2}{\Delta f \, \mathcal{P}_w(f_i)} \right] \tag{13.4}$$

and letting Δf tend to zero, to obtain

$$C_{[b/s]} = \int_{\mathcal{B}} \log_2 \left[1 + \mathcal{P}_s(f) \, \frac{|\mathcal{G}_{Ch}(f)|^2}{\mathcal{P}_w(f)} \right] df = \int_{\mathcal{B}} \log_2 [1 + \mathcal{P}_s(f) \, \Gamma_{Ch}(f)] \, df \tag{13.5}$$

We now find the optimum power spectral density $\mathcal{P}_s(f)$ that maximizes the capacity (13.5), under the constraint (13.1) and the condition $\mathcal{P}_s(f) \geq 0$. Applying the method of Lagrange multipliers, the optimum $\mathcal{P}_s(f)$ maximizes the integral

$$\int_{\mathcal{B}} \{\log_2 [1 + \mathcal{P}_s(f) \, \Gamma_{Ch}(f)] + \lambda \, \mathcal{P}_s(f)\} \, df \tag{13.6}$$

where λ is a Lagrange multiplier. Using the calculus of variations (see Appendix 8.A) we find that the optimum PSD must satisfy the following condition:

$$\frac{\Gamma_{Ch}(f)}{1 + \mathcal{P}_s(f) \, \Gamma_{Ch}(f)} + \lambda \ln 2 = 0 \tag{13.7}$$

Therefore the solution is given for $f \geq 0$ by

$$\mathcal{P}_{s,opt}(f) = \begin{cases} K - \dfrac{1}{\Gamma_{Ch}(f)} & f \in \mathcal{B} \\ 0 & \text{otherwise} \end{cases} \tag{13.8}$$

where $\mathcal{B} = \{f \in [0, +\infty) \colon \mathcal{P}_{s,opt}(f) > 0\}$ is the passband that allows achieving the capacity in (13.5), and K is a constant such that (13.1) is satisfied with the equal sign. This result is due to Shannon, and is valid for non-ideal linear channels in the presence of additive Gaussian noise. The function $\mathcal{P}_{s,opt}(f)$ is illustrated in Figure 13.1 for a typical behavior of the function $\Gamma_{Ch}(f)$; as the channel impulse response is assumed real valued, for $f < 0$ we get $\mathcal{P}_{s,opt}(f) = \mathcal{P}_{s,opt}(-f)$.

In fact, it results that $\mathcal{P}_s(f)$ should assume large values (small values) at frequencies for which $\Gamma_{Ch}(f)$ assumes large values (small values). From Figure 13.1 we note that if $1/\Gamma_{Ch}(f)$ is the profile of a cup in which we pour a quantity of water equivalent to V_P, the distribution of the water in the cup takes place according to the behavior predicted by (13.8); this observation leads to the *water pouring interpretation* of the optimum distribution of $\mathcal{P}_s(f)$ as a function of frequency.

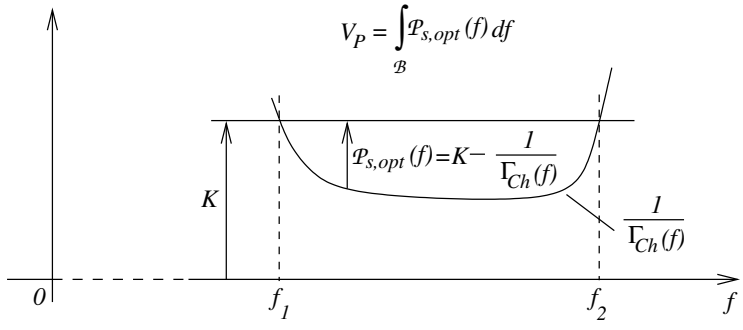


Figure 13.1. Illustration of $\mathcal{P}_{s,opt}(f)$ for a typical behavior of the function $\Gamma_{Ch}(f)$.

Shannon also demonstrated that capacity is achieved if $s(t)$ is a *Gaussian process* with power spectral density $\mathcal{P}_{s,opt}(f)$; the capacity in *bits per second* is given by

$$C_{[b/s]} = \int_{\mathcal{B}} \log_2[1 + \mathcal{P}_{s,opt}(f)\,\Gamma_{Ch}(f)]\,df \tag{13.9}$$

where $\mathcal{B}$ is defined by (13.8).

By analogy with the case of an ideal (non-dispersive) AWGN channel with limited bandwidth (see (6.280)), a linear dispersive channel can be roughly characterized by two parameters: the bandwidth $B = \int_{\mathcal{B}} df$ and the effective signal-to-noise ratio Γ_{eff}, implicitly defined so that

$$C_{[b/s]} = B\,\log_2(1 + \Gamma_{eff}) \tag{13.10}$$

The comparison with (13.9) yields

$$\Gamma_{eff} = \exp\left\{\frac{1}{B}\int_{\mathcal{B}} \ln[1 + \mathcal{P}_{s,opt}(f)\Gamma_{Ch}(f)]\,df\right\} - 1 \tag{13.11}$$

Note that $1 + \Gamma_{eff}$ corresponds to the geometric mean of the function $1 + \mathcal{P}_{s,opt}(f)\Gamma_{Ch}(f)$ in the band $\mathcal{B}$. By analogy with the case of an ideal AWGN channel with limited bandwidth analyzed in Chapter 6, it is useful to define a normalized signal-to-noise ratio for a transmission system over a linear dispersive channel that operates at a rate of the encoder-modulator equal to R_I, in bits per dimension, as

$$\bar{\Gamma}_{eff} = \frac{\Gamma_{eff}}{2^{2R_I} - 1} \tag{13.12}$$

where $\bar{\Gamma}_{eff} > 1$ measures the gap that separates the system being considered from capacity.

The passband $\mathcal{B}$ that allows achieving capacity represents the most important parameter of the spectrum obtained by water pouring. For many channels utilized in practice, the passband $\mathcal{B}$ is composed of only one frequency interval $[f_1, f_2]$, as illustrated in Figure 13.1; in other cases, for example, in the presence of high-power narrowband interference signals,

$\mathcal{B}$ may be formed by the union of disjoint frequency intervals. In practice, we find that the dependence of the capacity on $\mathcal{P}_s(f)$ is not as critical as the dependence on $\mathcal{B}$; a constant power spectral density in the band $\mathcal{B}$ usually allows a system to closely approach capacity [1, 2]. Therefore the application of the water pouring criterion may be limited to the determination of the passband.

13.2 Techniques to achieve capacity

A method to approach capacity is directly suggested by water pouring: the passband $\mathcal{B}$ that allows achieving capacity can be subdivided into disjoint sub-bands with sufficiently small bandwidth Δf so that both $\Gamma_{Ch}(f)$ and $\mathcal{P}_{s,opt}(f)$ are approximately constant in each of the resulting subchannels; then each subchannel can be modeled as an ideal AWGN channel with bandwidth Δf. This procedure is characteristic of OFDM systems, as discussed in Chapter 9.

The power to be assigned to a signal transmitted over a subchannel having center frequency f is given by $\mathcal{P}_{s,opt}(f)\Delta f$. Then, with good approximation, the signal-to-noise ratio in the subchannel is given by $\mathcal{P}_{s,opt}(f)\Gamma_{Ch}(f)$, and the corresponding capacity is given by

$$C_{[b/s]}(f) = \Delta f \log_2(1 + \mathcal{P}_{s,opt}(f)\,\Gamma_{Ch}(f)) \tag{13.13}$$

Observing that $\int_{\mathcal{B}} \mathcal{P}_{s,opt}(f)\,df = V_P$, as Δf tends to zero, the total power in all sub-bands approximates V_P and the total capacity approximates $C_{[b/s]}$, as indicated by (13.9). Indeed, to achieve capacity, powerful coding techniques must be applied so that transmission over each subchannel may take place at a bit rate close to the subchannel capacity $C_{[b/s]}(f)$.

Bit loading for OFDM

In practice, the number of bits per second that are transmitted over a given subchannel i is smaller than that indicated by (13.13), and is determined as follows. We assume the signal-to-noise ratio at the detection point of subchannel i is given by

$$\Gamma[i] = \frac{\mathcal{P}_s(f_i)\,|\mathcal{G}_{Ch}(f_i)|^2}{\mathcal{P}_w(f_i)} \tag{13.14}$$

We recall the definitions of the signal-to-noise ratio gap to capacity $\overline{\Gamma}_{gap,dB}$ for an uncoded M–QAM system, with $M \gg 1$ and $P_e = 10^{-7}$, and of coding gain $G_{code}[i]$ achieved by coding at $P_e = 10^{-7}$, given by (12.5) in dB. By including in (13.13) the parameters $\overline{\Gamma}_{gap,dB}$ and $G_{code}[i]$, we find that the number of bits per modulation interval that can be transmitted over subchannel i with $P_e = 10^{-7}$ is given by

$$\mathfrak{b}[i] = \log_2(1 + \Gamma[i]\,10^{(G_{code}[i] - \overline{\Gamma}_{gap,dB})/10}) \tag{13.15}$$

The *achievable bit rate* is obtained by summing the values given by (13.15) over all N subchannels, and multiplying the result by the modulation rate $1/T = \Delta f$, that is

$$R_b = \frac{1}{T}\sum_{i=1}^{N} \mathfrak{b}[i] \text{ bit/s} \tag{13.16}$$

In practice we resort to a technique called *bit loading* to determine the number of bits to be transmitted over each subchannel per modulation interval, under the constraints: 1) $\mathtt{b}[i]$ can take only a finite number of values determined by the signal constellations; and 2) the total transmitted power is fixed [3, 4, 5, 6].

Note that the modulation interval increases as the number of sub-bands increases. To reduce the delay in the recovery of the information, coding is usually applied "across the subchannels", see, for example, [7].

Discrete-time model of a single carrier system

If the passband $\mathcal{B}$ consists of only one frequency interval, as an alternative to OFDM transmission the capacity of a linear dispersive channel can be achieved by single carrier transmission. For a comparison of the characteristics of the two systems we refer to page 781.

First, we examine the equivalence between a continuous-time channel and a discrete-time channel with ISI; this equivalence is obtained by referring to a transmit filter that shapes the spectrum of the transmit signal as indicated by water pouring, and a receiver that implements a matched filter (MF) or a whitened matched filter (WMF), as illustrated in Section 8.8 (Figure 8.20). By the WMF we obtain a canonical form of the discrete-time channel with *trailing ISI*, that is ISI due only to postcursors, and additive white Gaussian noise (see Figure 13.2). As will be shown in this section, the contribution of the ISI to the capacity of the channel becomes negligible for high signal-to-noise ratios. This suggests that capacity can be achieved by combining ISI cancellation techniques with channel coding and shaping.

Assume that the passband $\mathcal{B}$ that allows achieving capacity consists of only one frequency interval $[f_1, f_2]$, with $0 < f_1 < f_2$ and bandwidth $B = f_2 - f_1$. If $\mathcal{B}$ consists of several intervals, then the same procedure can be separately applied to each interval, although in this case multicarrier transmission is usually preferable.

Consider passband transmission with modulation interval T and with minimum bandwidth of a complex-valued symbol sequence, that is we choose $T = 1/B$.

We recall the signal analysis that led to the scheme of Figure 8.20, extending it to the case of noise $w(t)$ with PSD $\mathcal{P}_w(f)$ not necessarily constant. With reference to Figure 13.2, from (7.29) the transmitted signal is given by

$$s(t) = Re\left(\sum_k a_k h_{Tx}(t - kT)\, e^{j2\pi f_0 t}\right) \tag{13.17}$$

where $\{a_k\}$ is modeled as a sequence of complex-valued i.i.d. symbols with *Gaussian* distribution and variance σ_a^2, and $h_{Tx}(t)$ denotes the transmit filter impulse response, with Fourier transform $\mathcal{H}_{Tx}(f)$. The transmit filter is chosen such that the PSD of $s(t)$, given by (7.27) and (7.28), is equal to $\mathcal{P}_{s,opt}(f)$, that is

$$\frac{\sigma_a^2}{4T}\,|\mathcal{H}_{Tx}(f)|^2 = \mathcal{P}_{s,opt}(f + f_0)\,1(f + f_0) \tag{13.18}$$

At the receiver, the signal $r(t)$ is first demodulated and filtered by a filter with impulse response $\overline{g}_{\mathrm{w}}$ that *suppresses* the signal components around $-2f_0$ and *whitens* the noise.

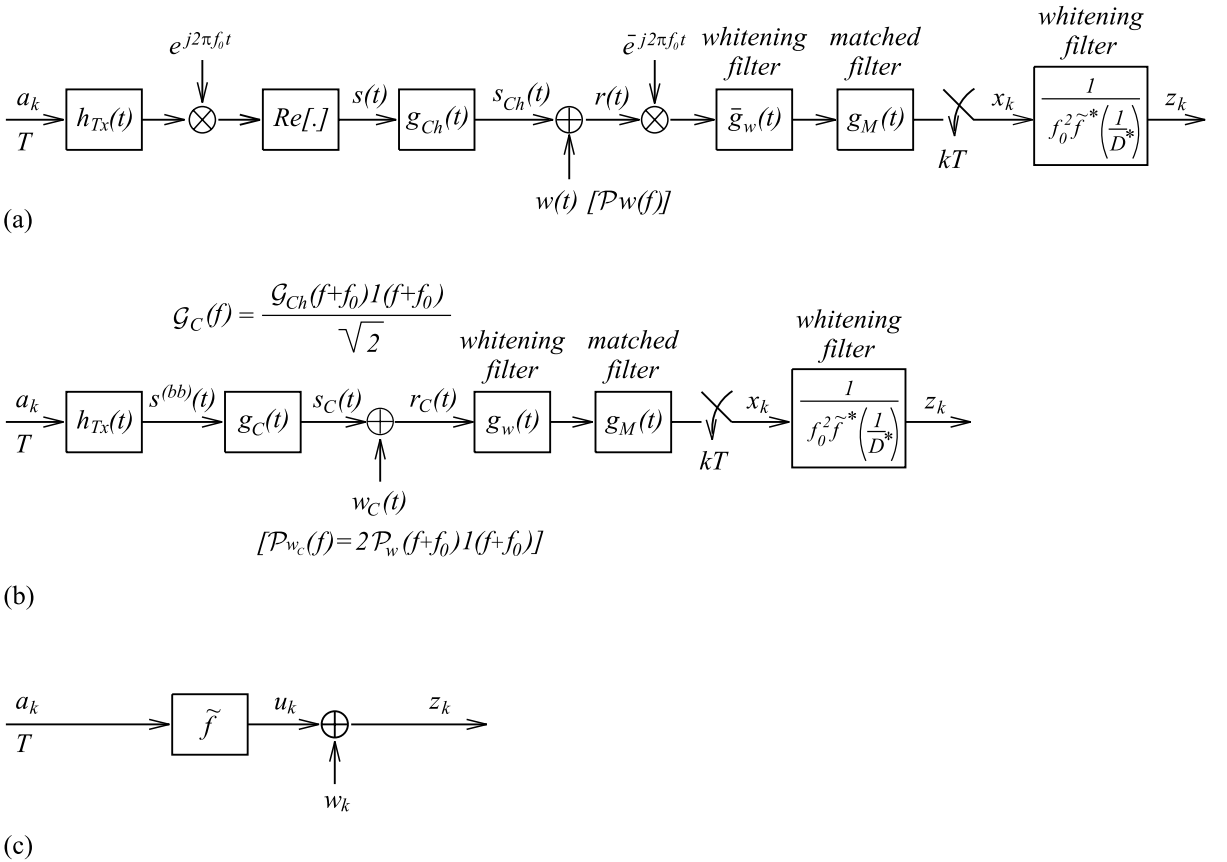


Figure 13.2. Equivalence between a continuous time system, (a) passband model, (b) baseband equivalent model with $g_w(t) = \frac{1}{\sqrt{2}}\bar{g}_w(t)$, and (c) a discrete-time system, for transmission over a linear dispersive channel.

As an alternative we could use a passband whitening filter phase splitter that suppresses the signal components with negative frequency and whitens the noise in the passband, see Section 8.14.1.

Consider the baseband equivalent model of Figure 13.2b, where $\mathcal{G}_C(f) = \frac{1}{\sqrt{2}}\,\mathcal{G}_{Ch}(f + f_0)\,1(f + f_0)$ and $\mathcal{P}_{w_C}(f) = 2\mathcal{P}_w(f + f_0)\,1(f + f_0)$; we define

$$\mathcal{B}_0 = [f_1 - f_0,\, f_2 - f_0] \tag{13.19}$$

as the new passband of the desired signal at the receiver. The whitening filter $g_{\mathrm{w}}(t) = \frac{1}{\sqrt{2}}\overline{g}_{\mathrm{w}}(t)$ has frequency response given by

$$|\mathcal{G}_{\mathrm{w}}(f)|^2 = \begin{cases} \dfrac{1}{\mathcal{P}_{w_C}(f)} & f \in \mathcal{B}_0 \\ 0 & \text{elsewhere} \end{cases} \tag{13.20}$$

From the scheme of Figure 8.20, the whitening filter is then followed by a matched filter g_M with frequency response

$$\mathcal{G}_M(f) = [\mathcal{H}_{Tx}(f)\,\mathcal{G}_C(f)\,\mathcal{G}_{\mathrm{w}}(f)]^* \tag{13.21}$$

The cascade of whitening filter g_{w} and MF g_M yields the composite filter $g_{Rc}^{MF}(f)$ with frequency response

$$\begin{aligned} \mathcal{G}_{Rc}^{MF}(f) &= \mathcal{H}_{Tx}^*(f)\,\mathcal{G}_C^*(f)\,|\mathcal{G}_{\mathrm{w}}(f)|^2 \\ &= \frac{\mathcal{H}_{Tx}^*(f)\,\mathcal{G}_C^*(f)}{\mathcal{P}_{w_C}(f)} \end{aligned} \tag{13.22}$$

The overall QAM pulse q at the output of the MF has frequency response given by

$$\mathcal{Q}(f) = \mathcal{H}_{Tx}(f)\,\mathcal{G}_C(f)\,\mathcal{G}_{Rc}^{MF}(f) = \frac{|\mathcal{H}_{Tx}(f)\,\mathcal{G}_C(f)|^2}{\mathcal{P}_{w_C}(f)} \tag{13.23}$$

Note that $\mathcal{Q}(f)$ has the properties of a PSD with passband $\mathcal{B}_0$; in particular, $Q(f)$ is equal to the noise PSD $\mathcal{P}_{w_R}(f)$ at the MF output, as

$$\mathcal{P}_{w_R}(f) = \mathcal{P}_{w_C}(f)\,|\mathcal{G}_{Rc}^{MF}(f)|^2 = \frac{|\mathcal{H}_{Tx}(f)\,\mathcal{G}_C(f)|^2}{\mathcal{P}_{w_C}(f)} = \mathcal{Q}(f) \tag{13.24}$$

Therefore the sequence of samples at the MF output can be expressed as

$$x_k = \sum_{i=-\infty}^{+\infty} a_i\, h_{k-i} + \tilde{w}_k \tag{13.25}$$

where the coefficients $h_i = q(iT)$ are given by the samples of the overall impulse response $q(t)$, and

$$\tilde{w}_k = w_R(kT) = [w_C(t') * g_{Rc}^{MF}(t')](t)\Big|_{t=kT} \tag{13.26}$$

In general, the Fourier transform of the discrete-time response $\{h_i\}$ is given by

$$\mathcal{H}(f) = \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \mathcal{Q}\left(f - \frac{\ell}{T}\right) \tag{13.27}$$

In this case, because $\mathcal{Q}(f)$ is limited to the passband $\mathcal{B}_0$ with bandwidth $B = 1/T$, there is no aliasing; the function $\mathcal{H}(f)$, periodic of period $1/T$, is therefore equal to $(1/T)\mathcal{Q}(f)$ in the band $\mathcal{B}_0$. As $\mathcal{P}_{w_R}(f) = \mathcal{Q}(f)$, $\{\tilde{w}_k\}$ is a sequence of Gaussian noise samples with autocorrelation sequence $\{\mathrm{r}_{\tilde{w}_k}(n)\} = h_n$. Note, moreover, that $\{h_i\}$ satisfies the Hermitian property, as $\mathcal{H}$ is real valued.

We have thus obtained a discrete-time equivalent channel that can be described using the D transform as

$$x(D) = a(D)h(D) + \tilde{w}(D) \tag{13.28}$$

where $h(D)$ has Hermitian symmetry.

We now proceed to develop an alternative model of discrete-time equivalent channel with causal, monic, and minimum-phase response $\tilde{f}(D)$, and additive white Gaussian noise $w(D)$. In this regard, we recall the theorem of spectral factorization for discrete time systems (see page 53). If $\mathcal{H}(f)$ satisfies the Paley–Wiener condition, then the function $h(D)$ can be factorized as follows:

$$h(D) = \tilde{f}^*\left(\frac{1}{D^*}\right) f_0^2 \tilde{f}(D) \tag{13.29}$$

$$\mathcal{H}(f) = \tilde{\mathcal{F}}^*(f)\, f_0^2\, \tilde{\mathcal{F}}(f) \tag{13.30}$$

where the function $\tilde{f}(D) = 1 + \tilde{f}_1 D + \cdots$ is associated with a causal ($\tilde{f}_i = 0$ for $i < 0$), monic and minimum-phase sequence $\tilde{f}_i$, and $\tilde{\mathcal{F}}(f) = \tilde{f}(e^{-j2\pi fT})$ is the Fourier transform of the sequence $\{\tilde{f}_i\}$. The factor f_0^2 is the geometric mean of $\mathcal{H}(f)$ over an interval of measure $1/T$, that is

$$\log f_0^2 = T \int_{1/T} \log \mathcal{H}(f)\, df \tag{13.31}$$

where logarithms may have any common base.

Then (13.28) can be written as

$$x(D) = a(D)\, f_0^2\, \tilde{f}(D)\, \tilde{f}^*\left(\frac{1}{D^*}\right) + w'(D)\, f_0\, \tilde{f}^*\left(\frac{1}{D^*}\right) \tag{13.32}$$

where $w'(D)$ is a sequence of i.i.d. Gaussian noise samples with unit variance. Filtering $x(D)$ by a filter having transfer function $1/[f_0^2 \tilde{f}^*(1/D^*)]$, we obtain the discrete-time equivalent canonical model of the dispersive channel

$$z(D) = a(D)\, \tilde{f}(D) + w(D) \tag{13.33}$$

where $w(D)$ is a sequence of i.i.d. Gaussian noise samples with variance $1/f_0^2$. Equation (13.33) is obtained under the assumption that $\tilde{f}(D)$ has a stable reciprocal function, and

hence $\tilde{f}^*(1/D^*)$ has an anticausal stable reciprocal function; this condition is verified if $h(D)$ has no spectral zeros.

However, to obtain the reciprocal of $\tilde{f}(D)$ does not represent a problem, as $z(D)$ can be indirectly obtained from the sequence of samples at the output of the WMF. The transfer function of the composite filter that consists of the whitening filter g_w and the WMF has a transfer function given by

$$\mathcal{G}_{Rc}^{WMF}(f) = \frac{\mathcal{G}_{Rc}^{MF}(f)}{f_0^2 \tilde{\mathcal{F}}^*(f)} = \frac{\mathcal{H}_{Tx}^*(f)\,\mathcal{G}_C^*(f)}{\mathcal{P}_{w_C}(f)}\,\frac{\tilde{\mathcal{F}}(f)}{\mathcal{H}(f)} \tag{13.34}$$

The only condition for the stability of the filter (13.34) is given by the Paley–Wiener criterion.

Achieving capacity with a single carrier system

Note that the model (13.33) expresses the output sequence as the sum of the noiseless sequence $a(D)\,\tilde{f}(D)$ and additive white Gaussian noise $w(D)$.

From (13.33), if $a(D)$ is an uncoded sequence with symbols taken from a finite constellation, and $\tilde{f}(D)$ has finite length, then the received sequence in the absence of noise $a(D)\,\tilde{f}(D)$ can be viewed as the output of a finite state machine, and the sequence $a(D)$ can be optimally detected by the Viterbi algorithm, as discussed in Chapter 8. As an alternative, MLSD can be directly performed by considering the MF output sequence $x(D)$, using a trellis of the same complexity but with a different metric (see Section 8.11).

In fact the MF output sequence $x(D)$ can be obtained from the WMF output sequence $z(D)$ by filtering $z(D)$ with a stable filter having transfer function $f_0^2 \tilde{f}^*(1/D^*)$. As $x(D)$ is a sufficient statistic (see note 1 on page 440) for the detection of $a(D)$, also $z(D)$ is a sufficient statistic; therefore the capacity of the overall channel including the MF or the WMF is equal to the capacity $C_{[b/s]}$ given by (13.9). Therefore capacity can be achieved by coding in combination with the cancellation of ISI.

We now evaluate the capacity. Using (13.23), (13.18), and (13.2) with the definitions of $\mathcal{G}_C$ and $\mathcal{P}_{w_C}$, yields

$$\mathcal{Q}(f) = \frac{|\mathcal{H}_{Tx}(f)\,\mathcal{G}_C(f)|^2}{\mathcal{P}_{w_C}(f)} = \frac{4T}{\sigma_a^2}\,\mathcal{P}_{s,opt}(f+f_0)\,\frac{1}{4}\,\Gamma_{Ch}(f+f_0)\,1(f+f_0) \tag{13.35}$$

and capacity can be expressed as

$$\begin{aligned} C_{[b/s]} &= \int_{\mathcal{B}} \log_2[1 + \mathcal{P}_{s,opt}(f)\,\Gamma_{Ch}(f)]\,df \\ &= \int_{\mathcal{B}_0} \log_2[1 + \mathcal{P}_{s,opt}(f+f_0)\,\Gamma_{Ch}(f+f_0)]\,df \\ &= \int_{\mathcal{B}_0} \log_2\left[1 + \frac{\sigma_a^2}{T}\,\mathcal{Q}(f)\right] df \end{aligned} \tag{13.36}$$

Recall from (13.27) that $\mathcal{H}(f)$, periodic of period $1/T$, is equal to $(1/T)\mathcal{Q}(f)$ in the band $\mathcal{B}_0$; therefore using (13.31) for $B = 1/T$, the capacity $C_{[b/s]}$ and its approximation

for large values of the signal-to-noise ratio Γ can be expressed as

$$C_{[b/s]} = \int_{1/T} \log_2(1 + \sigma_a^2 \mathcal{H}(f))\, df \simeq \int_{1/T} \log_2(\sigma_a^2 \mathcal{H}(f))\, df = B \log_2(\sigma_a^2 f_0^2) \quad (13.37)$$

Assume that the tail of the impulse response that causes ISI can be in some way eliminated, so that at the receiver we observe the sequence $a(D) + w(D)$ rather than the sequence (13.33). The signal-to-noise ratio of the resultant ideal AWGN channel becomes $\Gamma_{ISI-free} = \sigma_a^2 f_0^2$; thus, from (6.280) the capacity of the ISI-free channel and its approximation for large values of Γ become

$$C_{ISI-free\,[b/s]} = B \log_2(1 + \sigma_a^2 f_0^2) \simeq B \log_2(\sigma_a^2 f_0^2) \quad (13.38)$$

Comparing (13.37) and (13.38) we finally obtain

$$C_{[b/s]} \simeq C_{ISI-free\,[b/s]} \quad (13.39)$$

Price was the first to observe that for large values of Γ we obtain (13.39), that is for high signal-to-noise ratios the capacity $C_{[b/s]}$ of the linear dispersive channel is approximately equal to the capacity of the ideal ISI-free channel obtained assuming that ISI can be in some way eliminated from the sequence $x(D)$; in other words, ISI does not significantly contribute to the capacity [1].

13.3 Precoding and coding for dispersive channels

The convolution $u(D) = a(D)\, \tilde{f}(D)$ that determines the output sequence in the absence of noise can be viewed as the transform of a vector by the channel matrix, that is $\mathbf{u} = \tilde{\mathbf{F}}\mathbf{a}$, where $\mathbf{a}$ and $\mathbf{u}$ are vectors whose components are given by the transmitted symbols and channel output samples, respectively. As the canonical response $\tilde{f}(D)$ is causal and monic, the matrix $\tilde{\mathbf{F}}$ is triangular with all elements equal to 1 on the diagonal and therefore has determinant equal to 1; thus it follows that the matrix $\tilde{\mathbf{F}}$ identifies a linear transformation that preserves the volume between the input and output spaces. In other words, the channel matrix $\tilde{\mathbf{F}}$ transforms a hypersphere into a hyperellipsoid having the same volume and containing the same number of constellation points.

Coding methods for linear dispersive channels [8, 9, 10, 11, 12, 13, 14] that yield high values of coding gain can be obtained by requiring that the channel output vectors in the absence of noise $\mathbf{u}$ are points of a set Λ' with good properties in terms of Euclidean distance, for example, a lattice identified by integers. From the model of Figure 13.2c, an intuitive explanation of the objectives of coding for linear dispersive channels is obtained by considering the signal sequences $\mathbf{a}$, $\mathbf{u}$, $\mathbf{w}$, and $\mathbf{z}$ as vectors with a finite number of components. If the matrix $\tilde{\mathbf{F}}$ is known at the transmitter, the input vector $\mathbf{a}$ can be predistorted so that the points of the vector $\mathbf{a}$ correspond to points of a signal set $\tilde{\mathbf{F}}^{-1}\Lambda$; the volumes $V(\Lambda)$ in the output signal space, and $V(\tilde{\mathbf{F}}^{-1}\Lambda)$ in the input signal space are equal. In any case, the output channel vectors are observed in the presence of additive white Gaussian noise vectors $\mathbf{w}$. If a detector chooses the point of the lattice Λ' with minimum distance from the output vector we obtain a coding gain, relative to Λ, as in the case of an ideal AWGN channel (see Section 6.10).

Recall that to achieve capacity it is necessary that the distribution of the transmitted signal approximates a Gaussian distribution. We mention that commonly used methods of shaping, which also minimize the transmitted power, require that the points of the input constellation are uniformly distributed within a hypersphere [11]. Coding and shaping thus occur in two Euclidean spaces related by a known linear transformation that preserves the volume. Coding and shaping can be separately optimized, by choosing a method for predistorting the signal set in conjunction with a coding scheme that leads to a large coding gain for an ideal AWGN channel in the signal space where coding takes place, and a method that leads to a large shaping gain in the signal space where shaping takes place. In the remaining part of this chapter we focus on precoding and coding methods to achieve large coding gains for transmission over channels with ISI, assuming the channel impulse response is known.

13.3.1 Tomlinson–Harashima (TH) precoding

Precoding is a method of pre-equalization that allows also for transmission over linear dispersive channels the objectives of coding illustrated in Section 6.10 for transmission over ideal AWGN channels. We now extend the method discussed in Appendix 7.A for partial response systems. Let $\tilde{f}(D) = 1 + \tilde{f}_1 D + \tilde{f}_2 D^2 + \cdots$ be the response of a discrete-time equivalent channel, with ISI and additive noise, and assume that the canonical response of the channel $\tilde{f}(D)$ is known at the transmitter. Let $\tilde{f}(D) = 1 + D\tilde{f}'(D)$, and furthermore assume that for every pair of symbols of the input constellation $\alpha_i, \alpha_j \in \mathcal{A}$ the following relation holds:[1] $\alpha_i = \alpha_j \bmod \Lambda_0$, where $\mathcal{A} \subset \Lambda_0 + \lambda$ is a finite set of symbols (constellation) to which the information bits are mapped, Λ_0 represents the lattice associated with $\mathcal{A}$, and λ is a given offset value, possibly non-zero (see Section 12.1.3).

The objective of all precoding methods, with and without channel coding, consists of transmitting a pre-equalized sequence

$$a^{(p)}(D) = \frac{u^{(p)}(D)}{\tilde{f}(D)} \tag{13.40}$$

so that, in the absence of noise, the channel output sequence $u(D) = a^{(p)}(D)\,\tilde{f}(D) = u^{(p)}(D)$ represents the output of an ideal channel apparently ISI-free, with input sequence $u^{(p)}(D)$; $u^{(p)}(D)$ is a sequence of symbols belonging to a set $\mathcal{A}^{(p)} \subset \Lambda_0 + \lambda$. To achieve this objective with channel input signal samples that belong to a given finite region, the cardinality of the set $\mathcal{A}^{(p)}$ must be greater than that of the set $\mathcal{A}$. The redundancy in $u^{(p)}(D)$ can therefore be used to minimize the average power of the sequence $a^{(p)}(D)$ given by (13.40), or to obtain other desirable characteristics of $a^{(p)}(D)$, for example, a low peak-to-average ratio. In the case of systems that adopt trellis coding, the channel output sequence in the absence of noise $u(D)$ must be a valid code sequence and can then be decoded by a decoder designed for an ideal channel. Note that, in a system with precoding, the elements of the transmitted sequence $a^{(p)}(D)$ are not in general symbols with discrete values.

[1] The expression $\alpha_i = \alpha_j \bmod \Lambda_0$ denotes that the two symbols α_i and α_j differ by a quantity that belongs to Λ_0.

The first precoding method was independently proposed for uncoded systems by Tomlinson [15] and Harashima [16] (TH precoding). Initially, TH precoding was not used in practice because in an uncoded transmission system the preferred method to cancel ISI employs a DFE, as it does not require to send information on the channel impulse response to the transmitter. However, if trellis coding is adopted, decision-feedback equalization is no longer a very attractive solution, as reliable decisions are made available by the Viterbi decoder only with a certain delay.

TH precoding, illustrated in Figure 13.3, uses memoryless operations at the transmitter and at the receiver to obtain samples of both the transmitted sequence $a^{(p)}(D)$ and the detected sequence $\hat{a}(D)$ within a finite region that contains $\mathcal{A}$. In principle, TH precoding can be applied to arbitrary symbol sets $\mathcal{A}$; however, unless it is possible to define an efficient extension of the region containing $\mathcal{A}$, the advantages of TH precoding are reduced by the increase of the transmit signal power (*transmit power penalty*). An efficient extension exists only if the signal space of $a^{(p)}(D)$ can be "tiled", that is completely covered without overlapping with translated versions of a finite region containing $\mathcal{A}$, given by the union of the Voronoi regions of symbols of $\mathcal{A}$, and defined as $R(\mathcal{A})$. Figure 13.4 illustrates the efficient extension of a two-dimensional 16-QAM constellation, where Λ_T denotes the sublattice of Λ_0 that identifies the efficient extension.

With reference to Figure 13.3, the precoder computes the sequence of channel input signal samples $a^{(p)}(D)$ as

$$a^{(p)}(D) = a(D) - p(D) + c(D) \tag{13.41}$$

where the sequence

$$p(D) = [\tilde{f}(D) - 1]\, a^{(p)}(D) = D\tilde{f}'(D)\, a^{(p)}(D) \tag{13.42}$$

represents the ISI at the channel output that must be compensated at the transmitter. The elements of the sequence $c(D)$ are points of the sublattice Λ_T used for the efficient extension of the region $R(\mathcal{A})$ that contains $\mathcal{A}$.[2] The k-th element $c_k \in \Lambda_T$ of the sequence $c(D)$ is chosen so that the statistical power of the channel input sample $a_k^{(p)}$ is minimum; in other words, the element c_k is chosen so that $a_k^{(p)}$ belongs to the region $R = R(\mathcal{A})$, as illustrated in Figure 13.4. From (13.40), (13.41), and (13.42), the channel output sequence in the

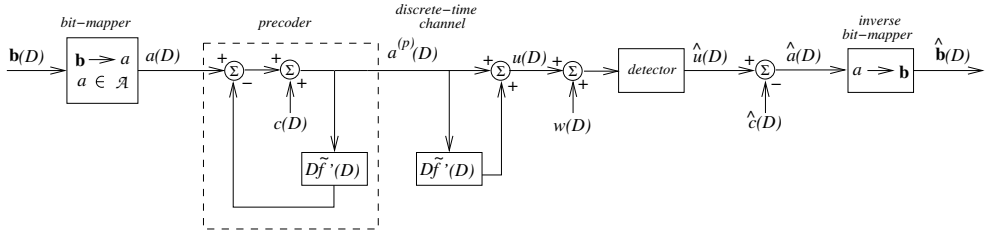


Figure 13.3. Block diagram of a system with TH precoding.

[2] Equation (13.41) represents the extension of (7.198) to the general case, in which the operation mod M is substituted by the addition of the sequence $c(D)$.

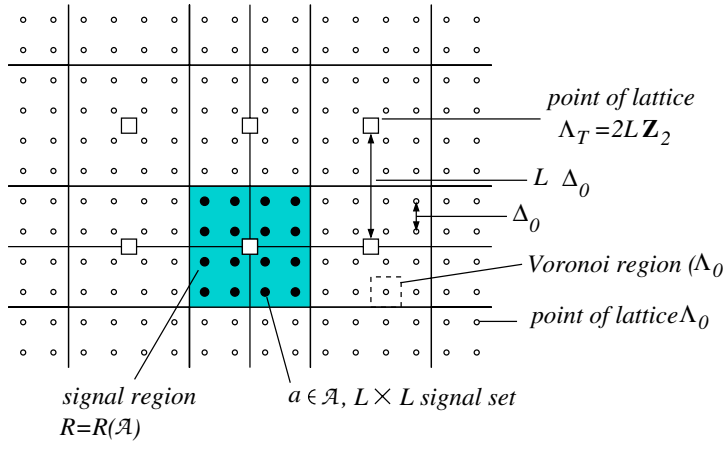


Figure 13.4. Illustration of the efficient extension of a two-dimensional 16-QAM constellation ($L = \sqrt{M} = 4$).

absence of noise is given by

$$\begin{aligned} u(D) &= a^{(p)}(D)\, \tilde{f}(D) = a^{(p)}[1 + D\, \tilde{f}'(D)] \\ &= a^{(p)}(D) + p(D) = a(D) + c(D) \end{aligned} \tag{13.43}$$

Note that from (13.43) we get the relation $u_k = a_k \mod \Lambda_T$, which is equivalent to (7.201).

The samples of the sequence $a^{(p)}(D)$ can be considered, with a good approximation, uniformly distributed in the region R. Assuming a constellation with $M = L \times L$ points for a QAM system, the power of the transmitted sequence is equal to that of a complex-valued signal with both real and imaginary parts that are uniformly distributed in $[-(L/2)\,\Delta_0, (L/2)\,\Delta_0]$, where Δ_0 denotes the minimum distance between points of the lattice Λ_0. Therefore using (5.34) it follows that

$$E[|a_k^{(p)}|^2] \simeq 2\,\frac{L^2}{12}\,\Delta_0^2 \tag{13.44}$$

Recalling that the statistical power of a transmitted symbol in a QAM system is given by $2\frac{L^2-1}{12}\Delta_0^2$ (see (6.182) for $\Delta_0 = 2$), we find that the transmit power penalty in a system that applies TH precoding is equal to $\Delta_0^2/12$ per dimension.

From (13.33), the channel output signal is given by

$$z(D) = u(D) + w(D) \tag{13.45}$$

where $w(D)$ represents a sequence of additive white Gaussian noise samples. In the case of TH precoding for an uncoded system, the detector yields a sequence $\hat{u}(D)$ of symbols belonging to the constellation $\mathcal{A}^{(p)}$; from (13.43) the detected sequence $\hat{a}(D)$ of transmitted symbols is therefore given by the memoryless operation

$$\hat{a}(D) = \hat{u}(D) - \hat{c}(D) \tag{13.46}$$

The k-th element $\hat{c}_k \in \Lambda_T$ of the sequence $\hat{c}(D)$ is chosen so that the symbol $\hat{a}_k = \hat{u}_k - \hat{c}_k$ belongs to the constellation $\mathcal{A}$. As the inverse operation of precoding is memoryless, error propagation at the receiver is completely avoided. Moreover, as the inversion of $\tilde{f}(D)$ is not required at the receiver, $\tilde{f}(D)$ may exhibit spectral nulls, that is it can contain factors of the form $(1 \pm D)$.

13.3.2 TH precoding and TCM

For the combination of trellis coding and TH precoding, the sequence

$$u(D) = a(D) + c(D) \tag{13.47}$$

must be a valid code sequence.

In practice, for all trellis codes that use a one-dimensional constellation with L points or a two-dimensional constellation with $L \times L$ points, if L is a multiple of 4 and $a(D)$ is a code sequence, then also $u(D)$ is a valid code sequence; in this way trellis coding and TH precoding can be combined by applying to the encoded symbol sequence TH precoding as discussed in the previous section [11]. For example, Figure 13.5 illustrates

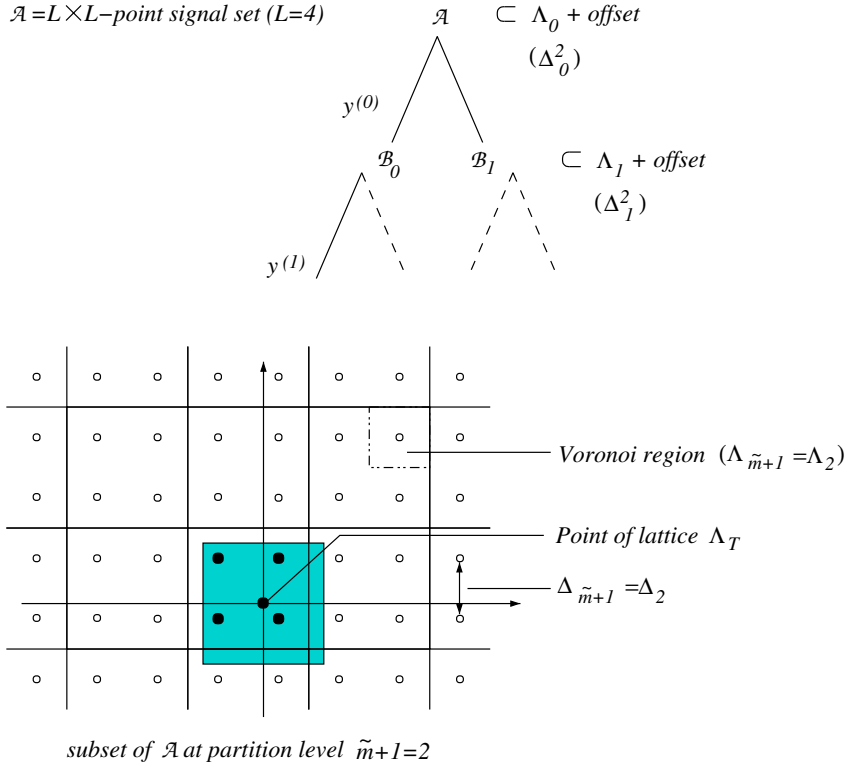


Figure 13.5. Partitioning of a 16-QAM constellation for trellis coding combined with TH precoding.

the partitioning of a 16-QAM constellation for trellis coding combined with TH precoding; note that summing a point of a lattice Λ_T (see Figure 13.4) with one of the points of the subset obtained at the partitioning level $\tilde{m}+1=2$, we still obtain a point of the subset.

In the general case of a one-dimensional constellation $\mathcal{A}$ with L points or a two-dimensional constellation with $L \times L$ points, with L even, the combination of trellis coding with TH precoding requires the application of trellis coding with feedback, or trellis augmented precoding, that we will now discuss [14]. Note that for L-PAM and $L \times L$-QAM constellations, the existence of an efficient extension is immediately verified. Moreover, for L even, we have that the subsets $\mathcal{B}_0 \subset \Lambda_1 + \lambda_1^0$ and $\mathcal{B}_1 \subset \Lambda_1 + \lambda_1^1$, obtained at the first level of partitioning, are congruent through a translation defined by a point of Λ_T, with the sets of points that are again subsets of $\Lambda_1 + \lambda_1^0$ and $\Lambda_1 + \lambda_1^1$, respectively. However, this property is not necessarily verified for all partitioning levels up to level $\tilde{m}+1$; an example is given by subsets obtained at the partitioning level $\tilde{m}+1=3$ of a 6×6–QAM constellation (see Figure 13.9).

A system using feedback to combine trellis coding with TH precoding is illustrated in Figure 13.6. The state of the trellis code $\mathbf{s}_{k-1}$ is known at the transmitter. The symbol a_k, into which the information represented by the binary vector $\mathbf{b}_k$ is mapped, is taken from the set $\mathcal{B}_{y_k^{(0)}}$, that is one of the two subsets obtained at the first level of partitioning. The set $\mathcal{B}_{y_k^{(0)}}$ is specified by the value $y_k^{(0)} = 0$ or 1 of the element with index k of the sequence $y^{(0)}(D)$, which is composed of the least significant bits of the vector sequence that describes the evolution of the state of the trellis code. The output sample $a_k^{(p)}$ is determined by (13.41).

As previously mentioned, to allow correct decoding operations the sequence $u(D)$ must represent a valid code sequence or, in other words, at the instant k the symbol u_k must represent a valid continuation of the code sequence $u(D)$ starting from the code state $\mathbf{s}_{k-1}$. The code sequence $u(D)$ is reproduced at the transmitter and presented at the input of a unit that determines, using the knowledge of the state $\mathbf{s}_{k-1}$ and the symbol u_k, the next state $\mathbf{s}_k$; this unit determines the bit sequence $y^{(0)}(D)$, such that the elements of the sequence $a(D)$ are chosen so that in turn $u(D)$ is a valid code sequence.

The code sequence $u(D)$, received in the presence of additive white Gaussian noise (see (13.45)), is input to a Viterbi decoder that yields the detected symbol sequence $\hat{u}(D)$. A

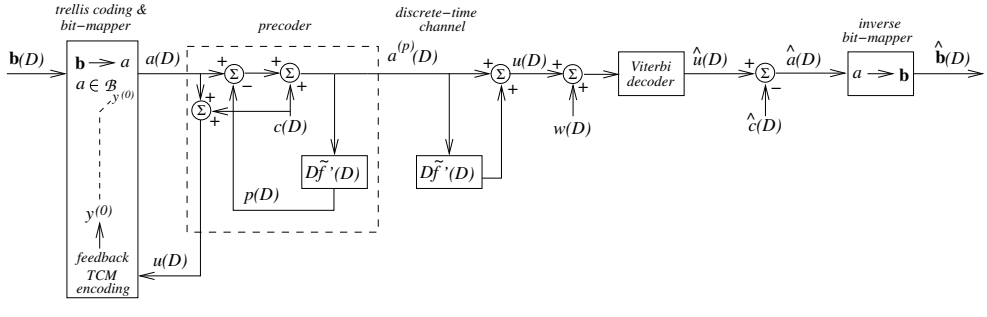


Figure 13.6. Block diagram of a system with trellis augmented precoding.

detection $\hat{a}(D)$ of the sequence $a(D)$ is obtained by the memoryless operation (13.46), thus avoiding error propagation. A detection $\hat{\mathbf{b}}(D)$ of the binary information vector sequence $\mathbf{b}(D)$ is then obtained by the inverse bit-mapping operation performed at the transmitter.

An example of application of trellis augmented precoding will be given next.

Example 13.3.1

Consider the transmission system with trellis augmented precoding illustrated in Figure 13.7. The code is an 8 state trellis code and the symbol constellation $\mathcal{A}$ is a 6×6-QAM constellation. Assume that the channel frequency response exhibits spectral nulls at $f = 0$ and $f = 1/(2T)$ Hz, and is given by

$$\tilde{f}(D) = \frac{1 - D^2}{1 - \rho D} \tag{13.48}$$

where $0 \leq \rho \leq 1$.

Figure 13.8a shows a conventional encoder for an 8-state trellis code that uses a systematic convolutional encoder for a code with rate 2/3 followed by a bit-mapper, and Figure 13.8b illustrates the code trellis diagram (see Chapter 12). The two-dimensional constellation $\mathcal{A}$ with 6×6 points and the set partitioning that yields the signal subsets assigned to the transitions on the trellis diagram are illustrated in Figure 13.9.

The mapping of the information bits $\mathbf{b} = (b_k^{(5)}, \ldots, b_k^{(1)}) \in \{(00000), (00001), \ldots, (10001)\}$, alphabet of cardinality 18, to symbols $a_k \in \mathcal{B}_{y_k^{(0)}}$, where $y_k^{(0)} \in \{0, 1\}$, is illustrated in Figure 13.9, where the lattice Λ_1, which will be used in the following, is also shown.

In particular, we show in Figure 13.10 the representation of the binary vector $\mathbf{y}_k = (y_k^{(2)}, y_k^{(1)}, y_k^{(0)})$, obtained using the set partitioning of Figure 13.9. Furthermore, we choose for the symbols u_k the representation given by

$$U_k = \left(U_{k,I},\ U_{k,Q}\right) \in 2\,\mathbf{Z}_2 + (1, 1) \tag{13.49}$$

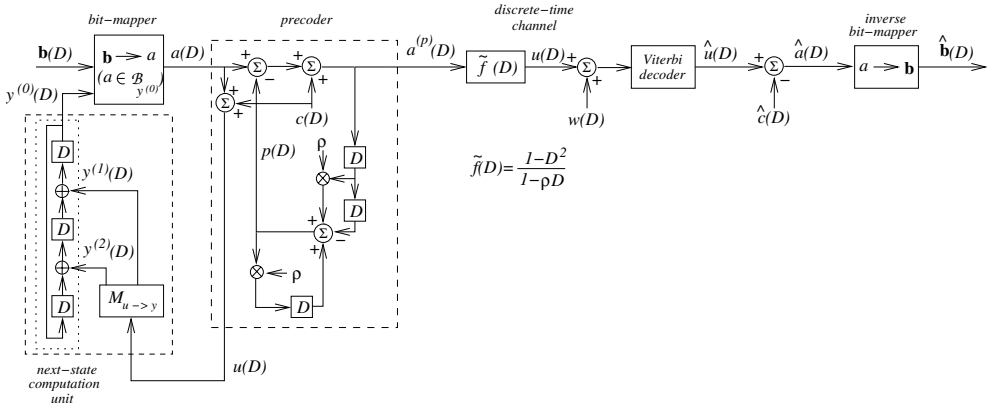
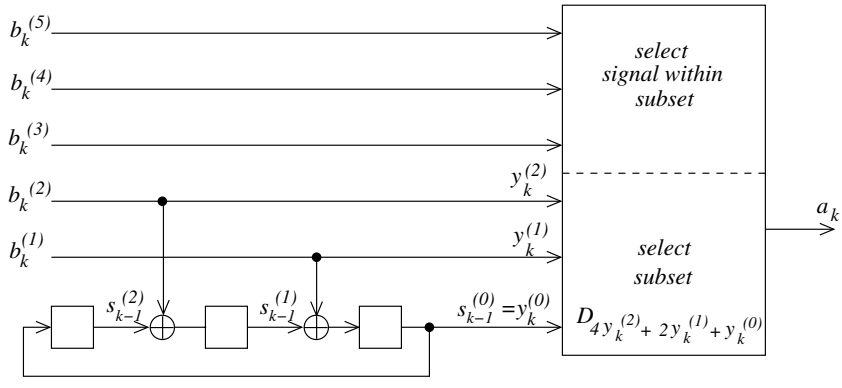
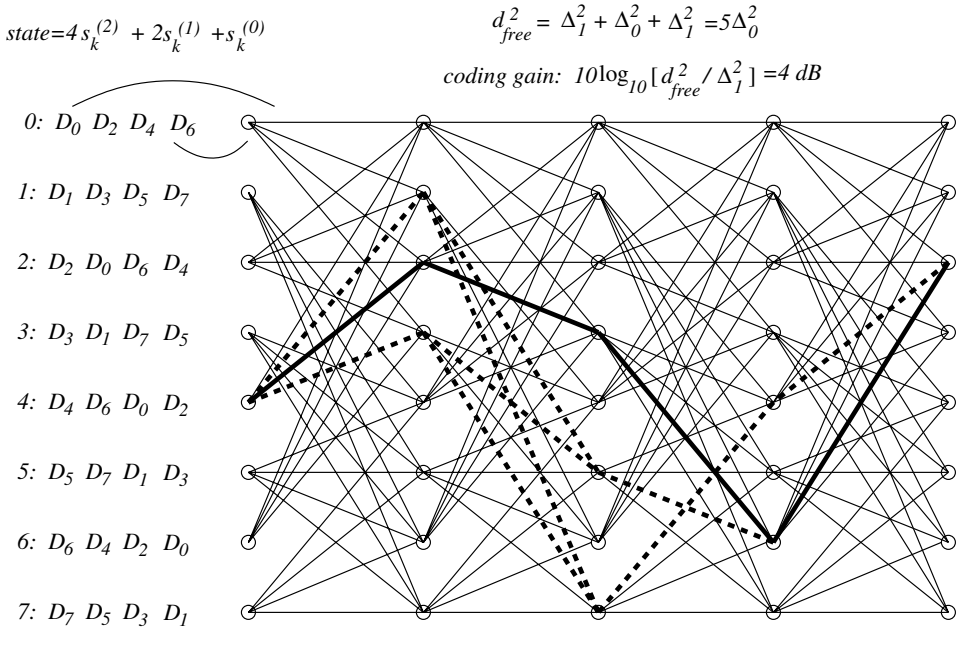


Figure 13.7. Example of a system with trellis augmented precoding.



(a)



(b)

Figure 13.8. Illustration of (a) conventional encoder for an 8-state trellis code and (b) trellis diagram.

that is

$$\begin{aligned} u_{k,I} &= -8u_{k,I}^{(3)} + 4u_{k,I}^{(2)} + 2u_{k,I}^{(1)} + 1 + c_k \\ u_{k,Q} &= -8u_{k,Q}^{(3)} + 4u_{k,Q}^{(2)} + 2u_{k,Q}^{(1)} + 1 + c_k \end{aligned} \tag{13.50}$$

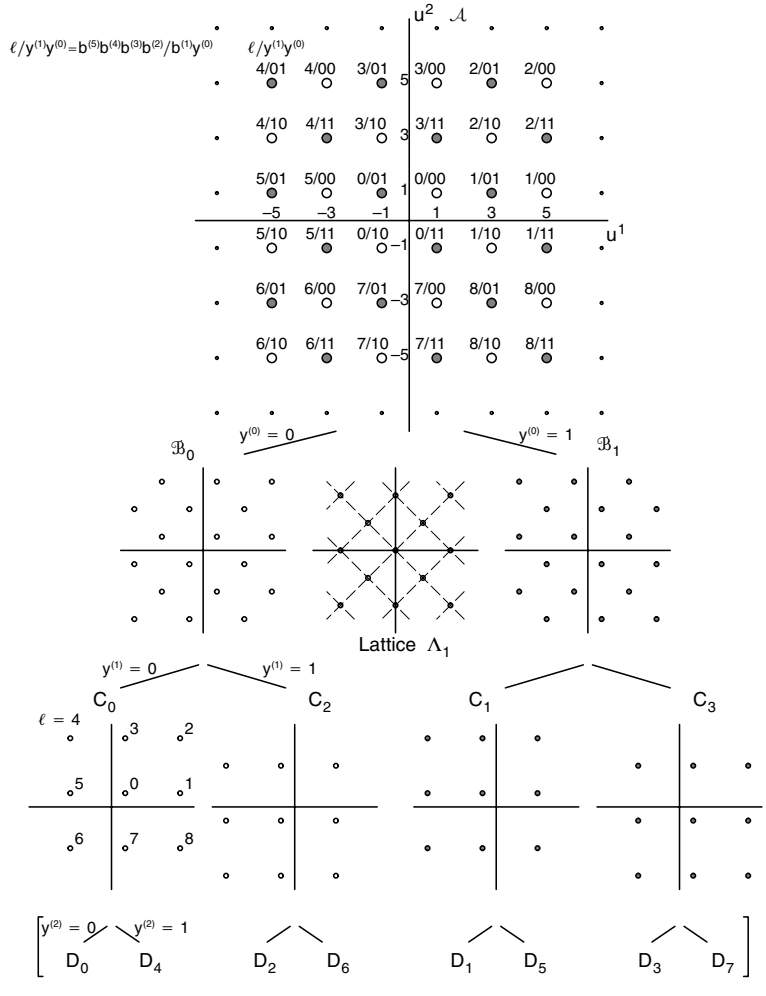


Figure 13.9. Partitioning of a 6×6-QAM constellation.

with $u_{k,I}^{(i)}, u_{k,Q}^{(i)} \in \{0, 1\}$, and $c_k \in \Lambda_T$. Note that the symbols a_k have a binary representation obtained by setting $c_k = 0$ in (13.50). It is possible to express the components of the vector $\mathbf{y}_k$ as a function of $u_{k,Q}^{(i)}, u_{k,Q}^{(i)}$, $i = 1, 2$, using for example the Karnaugh maps, as

$$\begin{aligned} y_k^{(0)} &= u_{k,I}^{(1)} \oplus u_{k,Q}^{(1)} \\ y_k^{(1)} &= u_{k,Q}^{(1)} \\ y_k^{(2)} &= u_{k,I}^{(2)} \oplus u_{k,I}^{(1)} \oplus u_{k,Q}^{(2)} \oplus u_{k,Q}^{(1)} \end{aligned} \tag{13.51}$$

Equations (13.51) define the map $M_{u \to y}$ used in Figure 13.7.

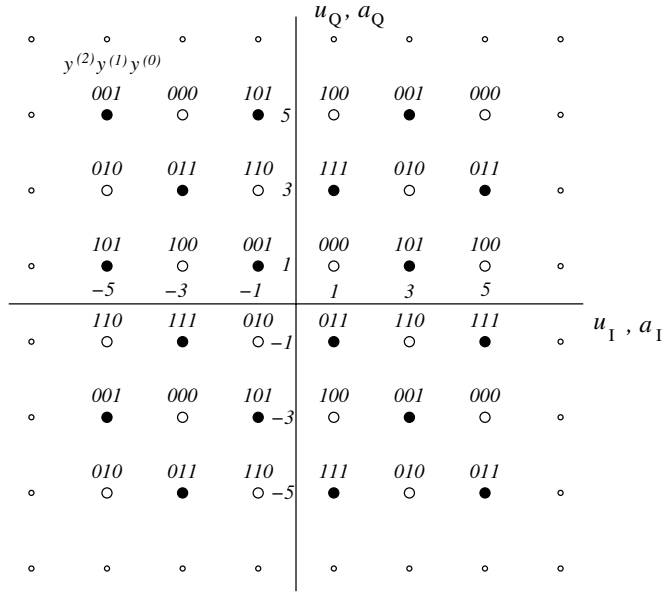


Figure 13.10. Signal constellation and bit mapping.

The sequence of channel input samples is given by (13.41), where

$$p(D) = [\tilde{f}(D) - 1]\, a^{(p)}(D) = D\frac{\rho - D}{1 - \rho D}\, a^{(p)}(D) \tag{13.52}$$

The elements of the sequence $c(D)$ are points of the lattice Λ_T that determines the efficient extension of $R(\mathcal{A})$, as illustrated in Figure 13.11. Recall that the value of the element $c_k \in \Lambda_T$ is chosen so that the statistical power of the channel input sample $a_k^{(p)}$ is minimum. The symbol sequence $u(D)$, which represents a valid trellis code sequence at the output of the channel with response $\tilde{f}(D)$ in the absence of noise, is given by (13.47).

At each instant k the element u_k of the sequence $u(D)$ is presented at the input of the unit that determines the next state of the code sequence starting from the present state $\mathbf{s}_{k-1}$. The state $\mathbf{s}_k$ is obtained by first determining the binary elements $y_k^{(1)}$ and $y_k^{(2)}$ of the sequences $y^{(1)}(D)$ and $y^{(2)}(D)$, as indicated by (13.51) and (13.50). Then the elements $y_k^{(1)}$ and $y_k^{(2)}$ are input to the systematic convolutional encoder for the code with rate 2/3; the encoder determines the next state $\mathbf{s}_k$ and it outputs the bit $y_{k+1}^{(0)}$, so that the map can generate a symbol a_{k+1} that is a valid continuation of the code sequence.

At the receiver, the Viterbi decoder outputs the sequence $\hat{u}(D)$. A detection $\hat{a}(D)$ of the sequence $a(D)$ is given by the memoryless operation

$$\hat{a}(D) = \hat{u}(D) - \hat{c}(D) \tag{13.53}$$

Then the detected sequence of information bits is obtained from the sequence $\hat{a}(D)$.

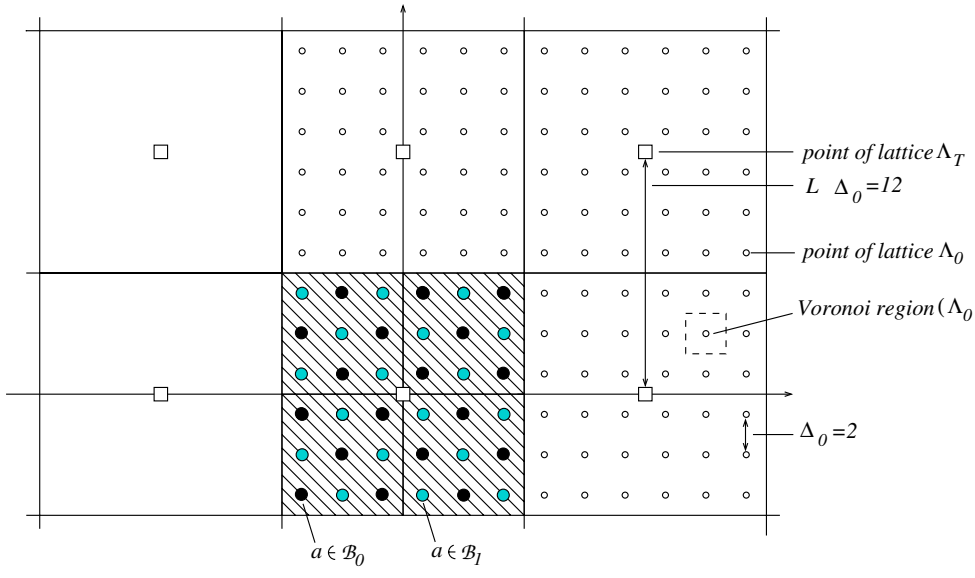


Figure 13.11. Illustration of the efficient extension of the 6 × 6-QAM constellation.

Note that the assumption of perfectly known channel characteristics only holds in an ideal situation. For example, if the considered method is applied to high speed data transmission over UTP cables, we recall from Section 4.4.2 that low-frequency disturbances and near-end cross-talk at high frequency are the main impairments. In this case it is not practical to convey to the transmitter information about the channel. The overall system must therefore be designed for worst-case channel characteristics, and deviations from the assumed characteristics must be compensated at the receiver by adaptive equalization.

13.3.3 Flexible precoding

Flexible precoding was originally introduced during the development of the V.34 modem standard for data transmission over the PSTN up to 28.8 kbit/s; a further version was applied in the V.34 bis standard for data transmission up to 33.6 kbit/s [12, 13].

First version. In the first version, illustrated in Figure 13.12, the transmitted signal $a^{(p)}(D)$ can be expressed as

$$a^{(p)}(D) = a(D) + d(D) \tag{13.54}$$

where $d(D)$ is called the dither signal and is given by

$$d(D) = c(D) - p(D) \tag{13.55}$$

where $p(D)$ is given by (13.42) and $c(D)$ is obtained from the quantization of $p(D)$ with quantizer $Q_{\Lambda_{\tilde{m}+1}}$. The quantizer $Q_{\Lambda_{\tilde{m}+1}}$ yields the k-th element of the sequence $c(D)$ by

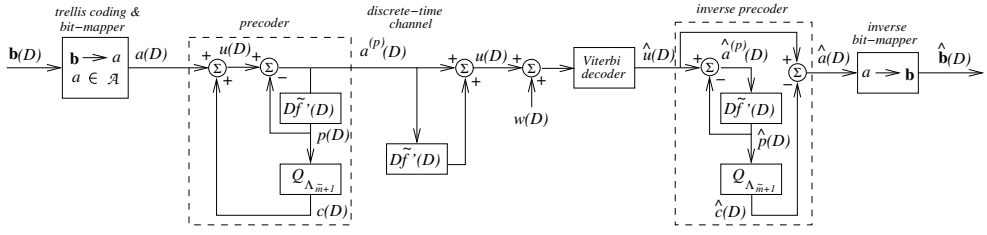


Figure 13.12. Block diagram of a system with flexible precoding.

quantizing the sample p_k to the closest point of the lattice $\Lambda_{\tilde{m}+1}$, which corresponds to the $(\tilde{m}+1)$-th level of partitioning of the signal set (see Chapter 12); in the case of an uncoded sequence, it is $\Lambda_{\tilde{m}+1} = \Lambda_0$. Note that the dither signal can be interpreted as the signal with minimum amplitude that must be added to the sequence $a(D)$ to obtain a valid code sequence at the channel output in the absence of noise. In fact, at the channel output we get the sequence $z(D) = u(D) + w(D)$, where $u(D)$ is obtained by adding a sequence of points taken from the lattice $\Lambda_{\tilde{m}+1}$ to the code sequence $a(D)$, and therefore it represents a valid code sequence.

The sequence $z(D)$ is input to a Viterbi decoder, which yields the detected sequence $\hat{u}(D)$. To obtain a detection of the sequence $a(D)$, it is first necessary to detect the sequence $\hat{c}(D)$ of the lattice points $\Lambda_{\tilde{m}+1}$ added to the sequence $a(D)$. Observing

$$\hat{a}^{(p)}(D) = \frac{\hat{u}(D)}{\tilde{f}(D)} \tag{13.56}$$

and

$$\hat{p}(D) = D\,\tilde{f}'(D)\,\hat{a}^{(p)}(D) \tag{13.57}$$

the sequence $\hat{c}(D)$ is obtained by quantizing with a quantizer $Q_{\Lambda_{\tilde{m}+1}}$ the sequence $\hat{p}(D)$, as illustrated in Figure 13.12. Then subtracting the sequence $\hat{c}(D)$ from $\hat{u}(D)$ we obtain the sequence $\hat{a}(D)$, that is used to detect the sequence of information bits.

At this point we can make the following observations with respect to the first version of flexible precoding:

1. an efficient extension of the region $R(\mathcal{A})$ is not necessary;

2. it is indispensable that the implementation of the blocks that perform similar functions in the precoder and in the inverse precoder (see Figure 13.12) is identical with regard to the binary representation of input and output signals;

3. as the dither signal can be assumed uniformly distributed in the Voronoi region $V(\Lambda_{\tilde{m}+1})$ of the point of the lattice $\Lambda_{\tilde{m}+1}$ corresponding to the origin, the transmit power penalty is equal to $\Delta^2_{\tilde{m}+1}/12$ per dimension; this can significantly reduce the coding gain if the cardinality of the constellation $\mathcal{A}$ is small;

4. to perform the inverse of the precoding operation, the inversion of the channel transfer function is required (see (13.56)); if $\tilde{f}(D)$ is minimum phase then $1/\tilde{f}(D)$ is stable and the effect of an error event at the Viterbi decoder output vanishes after a certain number of iterations; on the other hand, if $\tilde{f}(D)$ has zeros on the unit circle (spectral nulls), error events at the Viterbi decoder output can result in an unlimited propagation of errors in the detection of the sequence $a(D)$.

Second version. The second version of flexible precoding, illustrated in Figure 13.13, includes trellis coding with feedback, as previously discussed. In the precoder and in the inverse precoder the quantizer Q_{Λ_1} is now used, that yields the element c_k by quantizing the sample p_k to the closest point of the lattice Λ_1. Note that, if the symbol $a_k \in \mathcal{B}_{y_k^{(0)}}$, where we recall that $y_k^{(0)}$ represents the least significant bit of the trellis code state vector $\mathbf{s}_{k-1}$, then we have $u_k = a_k + c_k \in \mathcal{B}_{y_k^{(0)}}$. In the coding method with feedback from the knowledge of the state $\mathbf{s}_{k-1}$ and the symbol u_k, the next state $\mathbf{s}_k$ is determined, and consequently also the bit $y_{k+1}^{(0)}$; at the channel output, in the absence of noise, we therefore obtain a valid code sequence $u(D)$. Note that, as the dither signal is now uniformly distributed in the region $V(\Lambda_1)$, the transmit power penalty is reduced to $\Delta_1^2/12$ per dimension (see Figure 13.9).

The problem of unlimited error propagation that occurs in flexible precoding if the frequency response of the channel exhibits spectral nulls can be mitigated by referring to the scheme illustrated in Figure 13.14. Consider the signals and regions for a two-dimensional constellation $\mathcal{A}$ as illustrated, for example, in Figure 13.15 for a 16-QAM constellation; assume that the transfer function of the channel, that can exhibit spectral nulls, has the form

$$\tilde{f}(D) = \frac{1 + D\,\tilde{f}_N(D)}{1 + D\,\tilde{f}_D(D)} \tag{13.58}$$

To set a limit to error propagation, we exploit the knowledge that the expression of the dither signal is such that the transmitted signal sample at instant k, $a_k^{(p)}$, must be confined within the region $R_{y_k^{(0)}}$, obtained by the union of Voronoi regions of points of the subset

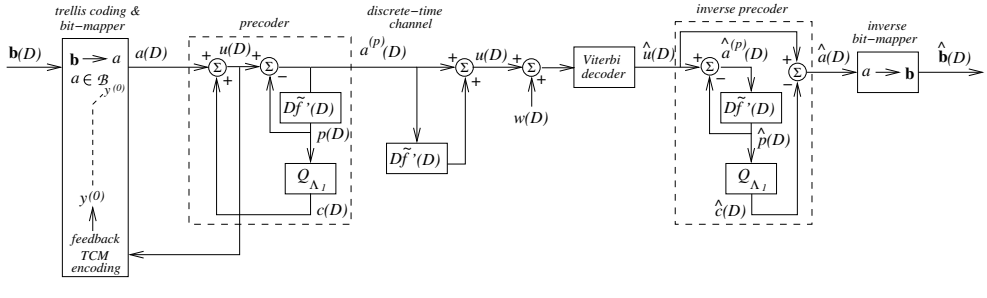


Figure 13.13. Block diagram of a system with flexible precoding and trellis coding with feedback.

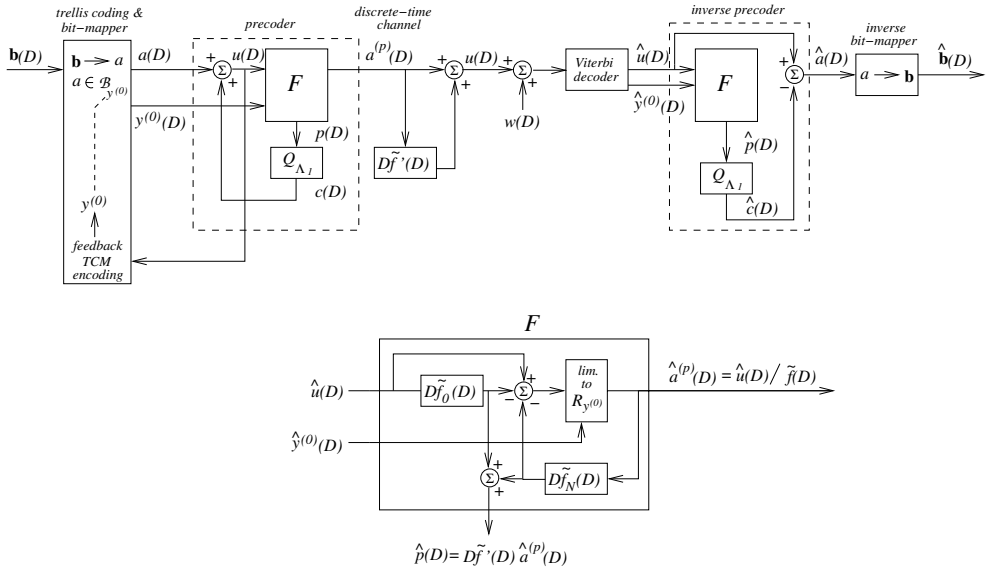


Figure 13.14. Block diagram of a system with flexible precoding and trellis coding with feedback that includes a method to mitigate error propagation in the inverse precoder.

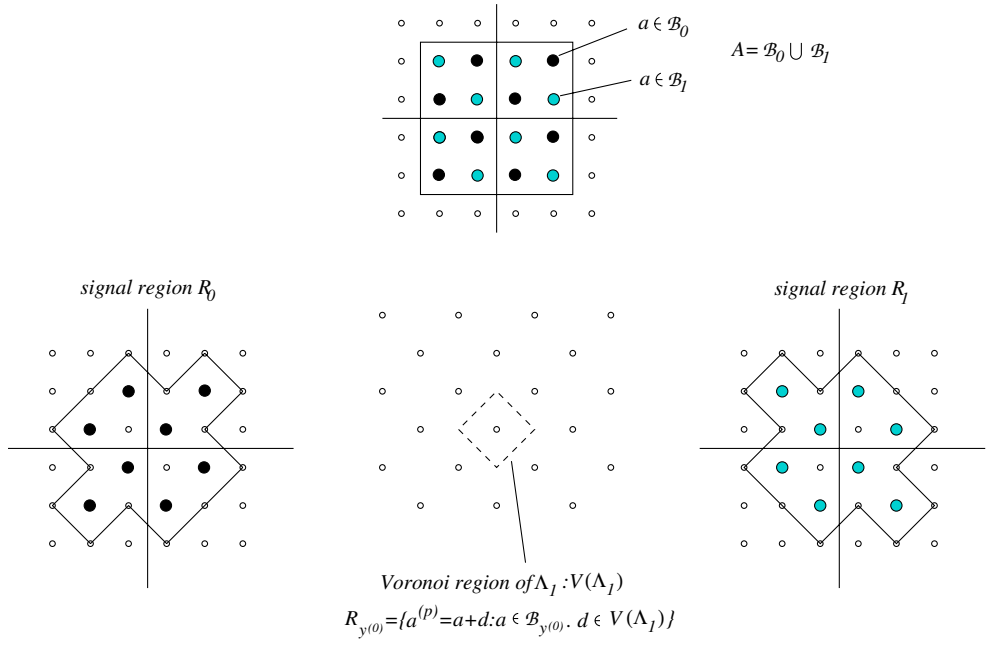


Figure 13.15. Illustration of signals and signal regions of a 16-QAM constellation for a system with flexible precoding and mitigation of the error propagation in the inverse precoder.

$\mathcal{B}_{y_k^{(0)}}$, defined as

$$R_{y_k^{(0)}} = \{a_k^{(p)} = \alpha + d : \alpha \in \mathcal{B}_{y_k^{(0)}},\ d \in V(\Lambda_1)\} \qquad y_k^{(0)} = 0, 1 \tag{13.59}$$

In the precoder and in the inverse precoder, we use identical units denoted as F (see Figures 13.14 and 13.15). Considering, for example, the transmitter we see that unit F contains a non-linear element that limits the transmitted sequence sample $a_k^{(p)}$ to the region $R_{y_k^{(0)}}$, selected as R_0 if $y_k^{(0)} = 0$, or as R_1 if $y_k^{(0)} = 1$. Note that the output signal of the non-linear element is input of the recursive section of a filter: this section is used for the inversion of the numerator of $\tilde{f}(D)$, which determines the presence of spectral nulls in the transfer function of the channel.

Example 13.3.2

Consider a dispersive AWGN channel with transfer function given by (13.48) with $\rho = 7/8$, and a QAM transmission system with a 6×6 constellation for an 8-state trellis code. The frequency response of the channel and the constellation $\mathcal{A}$ are illustrated in Figures 13.16 and 13.9, respectively. Recall that the frequency response of the channel exhibits spectral nulls at $f = 0$ and $f = 1/(2T)$ Hz.

Figure 13.17 illustrates the curves of symbol error probability as a function of the signal-to-noise ratio Γ obtained by simulating systems that employ: a) TH precoding and trellis coding with feedback; b) flexible precoding and trellis coding with feedback and limitation of error propagation; c) uncoded 18-QAM and a receiver with DFE. Note that, for the considered AWGN channel and a transmission of $\log_2(18)$ bit/s/Hz with an 8-state trellis code and 6×6-QAM, the system a) offers a margin larger than 1 dB with respect to system b), and a margin of 3 dB with respect to an uncoded system. This result can be explained by recalling that the error propagation is completely eliminated with TH precoding, while it can only be mitigated with flexible precoding.

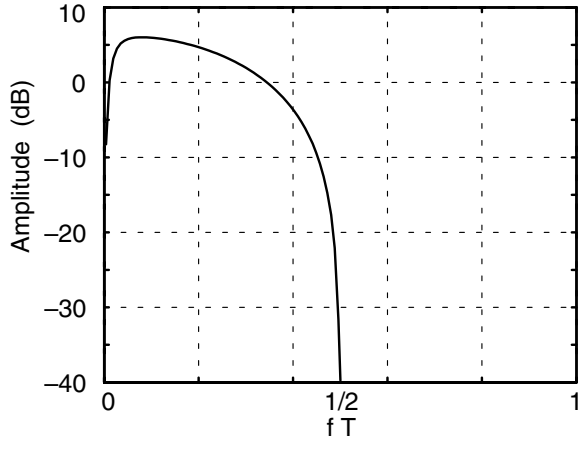


Figure 13.16. Frequency response with spectral nulls for $f = 0$ and $f = 1/(2T)$ Hz.

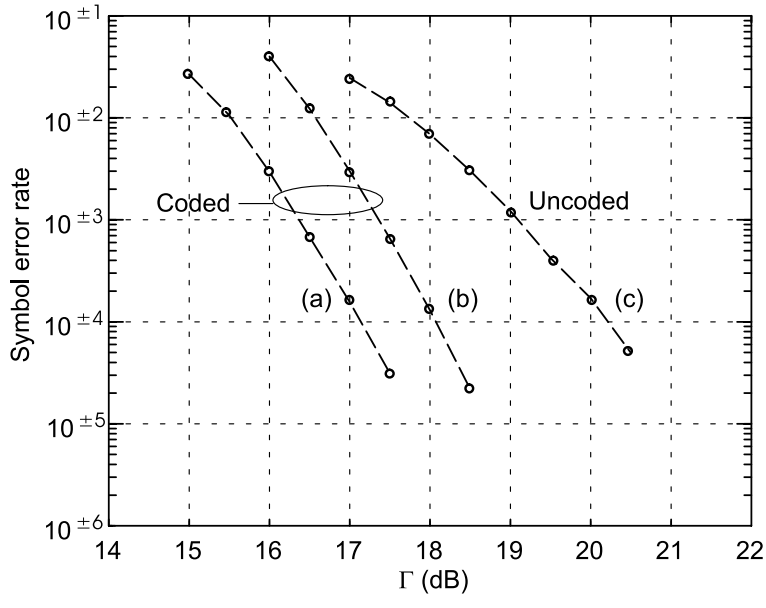


Figure 13.17. Symbol error probability as a function of the signal-to-noise ratio for systems that use: (a) TH precoding combined with trellis coding with feedback, (b) flexible precoding combined with trellis coding with feedback and limitation of error propagation, (c) uncoded 18-QAM and receiver with DFE.

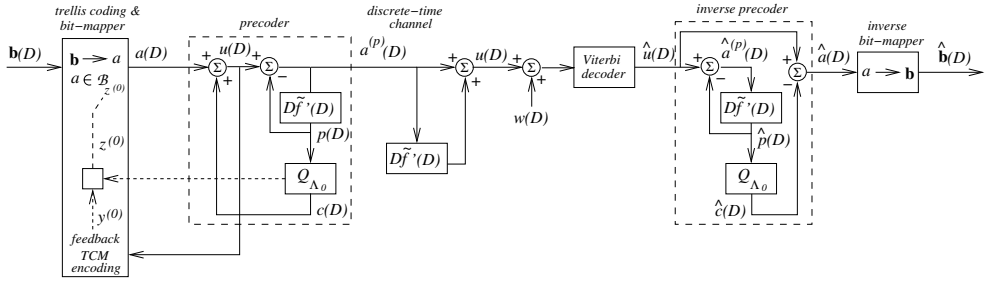


Figure 13.18. Block diagram of a system with flexible precoding and trellis coding with feedback, and a transmit power penalty equal to $\Delta_0^2/12$ per dimension.

A further reduction of the transmit power penalty in the flexible precoding and trellis coding with feedback can be obtained by using the scheme shown in Figure 13.18, where the quantizer Q_{Λ_0} yields the element c_k by quantizing the sample p_k to the closest point of the lattice Λ_0. Note that the dither signal is now uniformly distributed in $V(\Lambda_0)$ and therefore the transmit power penalty is reduced to $\Delta_0^2/12$ per dimension, that is the same value obtained with TH precoding. Observe that we have $c_k \in \Lambda_1$ or $c_k \in \Lambda_1 + (1, 0)$. To obtain a valid code sequence $u(D)$, we recall that it is necessary to obtain $u_k = a_k + c_k \in \mathcal{B}_{y_k^{(0)}}$.

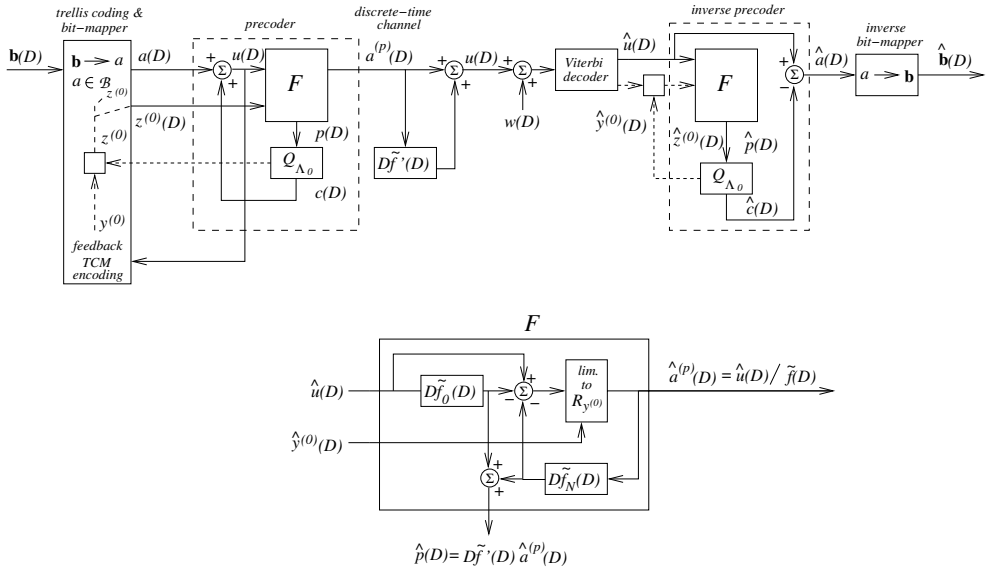


Figure 13.19. Method of mitigating error propagation for the system of Figure 13.18.

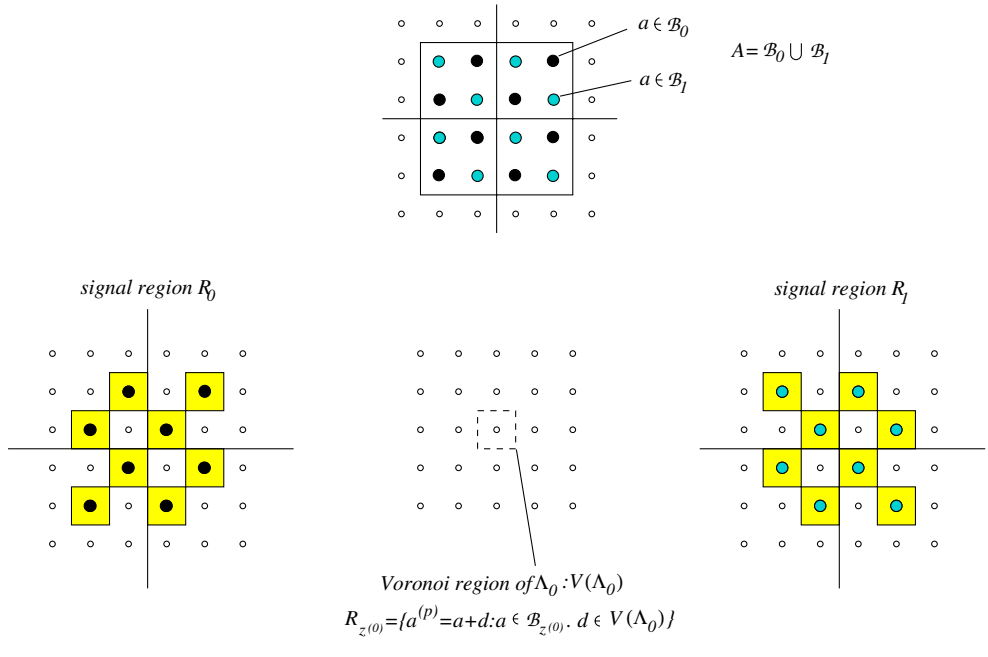


Figure 13.20. Illustration of signals and signal regions of a 16-QAM constellation for the system of Figure 13.19.

This condition is satisfied by choosing $a_k \in \mathcal{B}_{z_k^{(0)}}$, where $z_k^{(0)} = y_k^{(0)}$ if $c_k \in \Lambda_1$, and $z_k^{(0)} = \bar{y}_k^{(0)}$ if $c_k \in \Lambda_1 + (1, 0)$, where $\bar{y}_k^{(0)}$ denotes the 1's complement of $y_k^{(0)}$.

However, note that also with this version of flexible precoding channel inversion is required for the precoding inverse operation (see (13.56)). In case the channel frequency response has spectral nulls, mitigation of error propagation in detecting the sequence $a(D)$ is obtained by the scheme illustrated in Figure 13.19. Regions used to limit in the units F the sequence of transmitted samples $a^{(p)}(D)$ and the sequence of detected samples $\hat{a}^{(p)}(D)$ are shown in Figure 13.20 for a 16-QAM constellation.

Bibliography

[1] R. Price, "Nonlinearly feedback-equalized PAM versus capacity for noisy filter channels", *Proc. 1972 Int. Conf. Comm.*, pp. 22.12–17, June 1972.

[2] I. Kalet, "The multitone channel", *IEEE Trans. on Communications*, vol. 37, pp. 119–124, Feb. 1989.

[3] J. A. C. Bingham, "Multicarrier modulation for data transmission: an idea whose time has come", *IEEE Communications Magazine*, vol. 28, pp. 5–14, May 1990.

[4] P. S. Chow, J. M. Cioffi, and A. C. Bingham, "A practical discrete multitone transceiver loading algorithm for data transmission over spectrally shaped channels", *IEEE Trans. on Communications*, vol. 43, pp. 773–775, Feb./March/April 1995.

[5] A. Leke and J. M. Cioffi, "A maximum rate loading algorithm for discrete multitone systems", in *Proc. Globecom 1997*, pp. 1514–1518, Nov. 1997.

[6] J. Campello, "Practical bit loading for DMT", in *Proc. IEEE International Conference on Communications*, Vancouver, Canada, pp. 801–805, June 1999.

[7] G. Cherubini, E. Eleftheriou, and S. Ölcer, "Filtered multitone modulation for very-high-speed digital subscriber lines", *IEEE Journal on Selected Areas in Communications*, June 2002.

[8] J. M. Cioffi, G. P. Dudevoir, M. V. Eyuboglu, and G. D. Forney, Jr., "MMSE decision-feedback equalizers and coding", *IEEE Trans. on Communications*, vol. 43, pp. 2582–2604, Oct. 1995.

[9] G. Ungerboeck, "Channel coding with multilevel/phase signals", *IEEE Trans. on Information Theory*, vol. 28, pp. 55–67, Jan. 1982.

[10] G. D. Forney, Jr. and G. Ungerboeck, "Modulation and coding for linear Gaussian channels", *IEEE Trans. on Information Theory*, vol. 44, pp. 2384–2415, Oct. 1998.

[11] M. V. Eyuboglu and G. D. Forney, Jr., "Trellis precoding: combined coding, precoding and shaping for intersymbol interference channels", *IEEE Trans. on Information Theory*, vol. 38, pp. 301–314, Mar. 1992.

[12] R. Laroia, S. A. Tretter, and N. Farvardin, "A simple and effective precoding scheme for noise whitening on intersymbol interference channels", *IEEE Trans. on Communications*, vol. 41, pp. 1460–1463, Oct. 1993.

[13] R. Laroia, "Coding for intersymbol interference channels—Combined coding and precoding", *IEEE Trans. on Information Theory*, vol. 42, pp. 1053–1061, July 1996.

[14] G. Cherubini, S. Ölcer, and G. Ungerboeck, "Trellis precoding for channels with spectral nulls", in *Proc. 1997 IEEE Int. Symposium on Information Theory*, Ulm, Germany, p. 464, June 1997.

[15] M. Tomlinson, "New automatic equalizer employing modulo arithmetic", *Electronics Letters*, vol. 7, pp. 138–139, Mar. 1971.

[16] H. Harashima and H. Miyakawa, "Matched transmission technique for channels with intersymbol interference", *IEEE Trans. on Communications*, vol. 20, pp. 774–780, Aug. 1972.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 14

Synchronization

As a generalization of the receiver block diagram shown in Figure 7.5, the representation of the analog front end for a passband QAM system is illustrated in Figure 14.1. The received signal $r(t)$ is multiplied by a complex-valued carrier generated by a local oscillator, then filtered by an anti-imaging filter, g_{AI}, that extracts the image of the signal yielding the I and Q components of the signal $r_C(t)$. Often the function of the anti-imaging filter is performed by other filters, however, here g_{AI} is considered only as a model for the analysis and is assumed to be non-distorting. The receive oscillator is independent of the transmit oscillator; consequently carrier recovery must be performed using one of the following two strategies.

1. The first consists in multiplexing, usually in the frequency domain, a special signal, called a *pilot signal*, at the transmitter. This allows extracting the carrier at the receiver and therefore synchronizing the receive oscillator in phase and frequency with the transmit oscillator. If the pilot signal $v(t)$ consist of a non-modulated carrier, carrier recovery is obtained by the *phase-locked loop* (PLL) described in Section 14.2.

2. The second consists in getting the carrier directly from the modulated signal; this approach has the advantage that all the transmitted power is allocated for the transmission of the signal carrying the desired information. Some structures that implement this strategy are reported in Section 14.3.

In this chapter we will discuss methods for carrier phase and frequency recovery, as well as algorithms to estimate the timing phase. To avoid ambiguity, we refer to the latter as timing recovery algorithms, dropping the term phase. These algorithms are developed for application in the PAM and QAM transmission systems of Chapter 7, and the spread spectrum systems of Chapter 10. The problem of synchronization of OFDM systems was mentioned in Section 9.7.

14.1 The problem of synchronization for QAM systems

In Figure 14.1 the reconstructed carrier, in complex form, has the expression

$$\hat{v}_C(t) = \exp\{-j(2\pi f_1 t + \varphi_1)\} \tag{14.1}$$

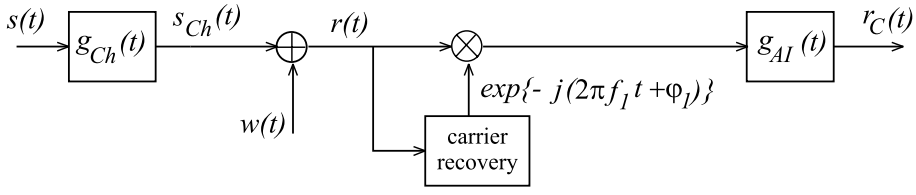


Figure 14.1. Analog front end for passband QAM systems.

Therefore, if the carrier generated at the transmitter, as observed at the receiver (*received carrier*) is given by $\exp\{j(2\pi f_0 t + \varphi_0)\}$,[1] in general we have a reconstruction error of the carrier phase $\varphi_{PA}(t)$ given by

$$\varphi_{PA}(t) = (2\pi f_1 t + \varphi_1) - (2\pi f_0 t + \varphi_0) \tag{14.2}$$

Let

$$\begin{aligned} \Omega &= 2\pi(f_0 - f_1) \\ \theta &= (\varphi_0 - \varphi_1) \end{aligned} \tag{14.3}$$

then (14.2) can be rewritten as

$$\varphi_{PA}(t) = -\Omega t - \theta \tag{14.4}$$

With reference to the notation of Figure 7.12, observing that now the phase offset is not included in the baseband equivalent channel impulse response, we have

$$g_C(t) = \frac{1}{2\sqrt{2}}\, g_{Ch}^{(bb)}(t)\, e^{-j \arg \mathcal{G}_{Ch}(f_0)} \tag{14.5}$$

The resulting baseband equivalent scheme of a QAM system is given in Figure 14.2. We assume that the anti-imaging filter frequency response is flat within the frequency interval

$$|f| \leq B + \frac{\Omega_{max}}{2\pi} \tag{14.6}$$

where B is the bandwidth of the signal s_C and Ω_{max} is the maximum value of $|\Omega|$.

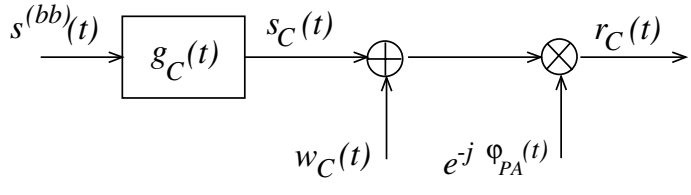


Figure 14.2. Baseband equivalent model of the channel and analog front end for a QAM system.

[1] In this chapter φ_0 is given by the sum of the phase of the transmitted carrier and the channel phase at $f = f_0$, equal to $\arg \mathcal{G}_{Ch}(f_0)$.

The received signal r_C is affected by a frequency offset Ω and a phase offset θ; moreover, the transmit filter h_{Tx} and the channel filter g_C introduce a transmission delay t_0. To simplify the analysis, this delay is assumed to be known with an error that is in the range $(-T/2, T/2)$. This coarse timing estimate can be obtained, for example, by a correlation method with known input (see (7.269)). This corresponds to assuming the overall pulse q_C non-causal, with peak at the origin.

Once set $t_0 = \varepsilon T$, with $|\varepsilon| \leq 1/2$, the signal $r_C(t)$ can be written as

$$r_C(t) = e^{j(\Omega t+\theta)} \sum_{k=-\infty}^{+\infty} a_k \, q_C(t - kT - \varepsilon T) + w_{C_\varphi}(t) \tag{14.7}$$

where

$$\begin{aligned} q_C(t) &= (h_{Tx} * g_C)(t + \varepsilon T) \\ w_{C_\varphi}(t) &= w_C(t)\, e^{-j\varphi_{PA}(t)} \end{aligned} \tag{14.8}$$

Furthermore, the receiver clock is independent of the transmitter clock, consequently the receiver clock period, that we denote as T_c, is different from the symbol period T at the transmitter: we assume that the ratio $F_0 = T/T_c$ is in general a real number.

The synchronization process consists in recovering the carrier, in the presence of phase offset θ and frequency offset Ω, and the timing phase or time shift εT.

14.2 The phase-locked loop

We assume the transmitted sinusoidal component $v(t)$ has been isolated from the signal at the receiver by a suitable narrowband filter. Now the problem consists in generating by a local oscillator, for example, using a PLL, a signal $v_{VCO}(t)$ with the same frequency and phase of $v(t)$, apart from a known offset.

A PLL, whose block diagram is shown in Figure 14.3, is a control system used to automatically regulate the phase of a locally generated signal $v_{VCO}(t)$ so that it coincides with that of an input signal $v(t)$. We assume that the two signals are given by

$$v(t) = A_1 \sin[\omega_0 t + \varphi_0(t)] \qquad v_{VCO}(t) = A_2 \cos[\omega_1 t + \varphi_1(t)] \tag{14.9}$$

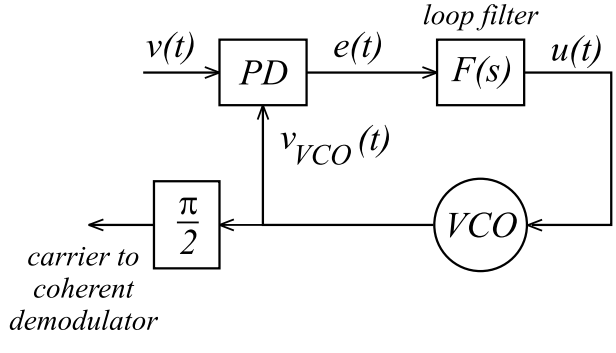


Figure 14.3. Block diagram of a PLL.

where the phase $\varphi_0(t)$ is a slowly time-varying function with respect to ω_0, or

$$\left|\frac{d\varphi_0(t)}{dt}\right| \ll \omega_0 \tag{14.10}$$

We write the instantaneous phase of $v_{VCO}(t)$ as follows:

$$\omega_1 t + \varphi_1(t) = \omega_0 t + (\omega_1 - \omega_0)\, t + \varphi_1(t) \tag{14.11}$$

and we let

$$\hat{\varphi}_0(t) = (\omega_1 - \omega_0)\, t + \varphi_1(t) \tag{14.12}$$

thus

$$v_{VCO}(t) = A_2 \cos[\omega_0 t + \hat{\varphi}_0(t)] \tag{14.13}$$

and $\hat{\varphi}_0(t)$ then represents the estimate of the phase $\varphi_0(t)$ obtained by the PLL.

We define the *phase error* as the difference between the instantaneous phases of the signals $v(t)$ and $v_{VCO}(t)$, that is

$$\phi(t) = (\omega_0 t + \varphi_0(t)) - (\omega_1 t + \varphi_1(t)) = \varphi_0(t) - \hat{\varphi}_0(t) \tag{14.14}$$

As illustrated in Figure 14.3, a PLL comprises:

- a *phase detector* (PD), that yields an output signal $e(t)$ given by the sine of the difference between the instantaneous phases of the two input signals, that is

$$e(t) = K_D \sin[\phi(t)] \tag{14.15}$$

  where K_D denotes the phase detector gain; we observe that $e(t)$ is an odd function of $\phi(t)$; therefore the PD produces a signal having the same sign as the *phase error*, at least for values of ϕ between $-\pi$ and $+\pi$;

- a lowpass filter $F(s)$, called *loop filter*, whose output $u(t)$ is equal to

$$u(t) = f * e(t) \tag{14.16}$$

- a *voltage-controlled oscillator* (VCO), which provides a periodic output signal $v_{VCO}(t)$ whose phase $\hat{\varphi}_0(t)$ satisfies the relation

$$\frac{d\hat{\varphi}_0(t)}{dt} = K_0\, u(t) \tag{14.17}$$

  called VCO *control law*, where K_0 denotes the VCO gain.

In practice, the PD is often implemented by a simple multiplier; then the signal $e(t)$ is proportional to the product of $v(t)$ and $v_{VCO}(t)$. If K_m denotes the multiplier gain and we define

$$K_D = \tfrac{1}{2}\, A_1\, A_2\, K_m \tag{14.18}$$

then we obtain

$$\begin{aligned} e(t) &= K_m\, v(t)\, v_{VCO}(t) \\ &= K_m\, A_1 \sin[\omega_0 t + \varphi_0(t)]\, A_2 \cos[\omega_1 t + \varphi_1(t)] \\ &= K_D \sin[\phi(t)] + K_D \sin[2\omega_0 t + \varphi_0(t) + \hat{\varphi}_0(t)] \end{aligned} \tag{14.19}$$

Note that, with respect to the signal $e(t)$ defined in (14.15), there is now an additional term with radian frequency $2\omega_0$. However, as from (14.10) $\varphi_0(t)$ is slowly varying in comparison with the term at frequency $2\omega_0$, the high frequency components are eliminated by the lowpass filter $F(s)$ or, in case the lowpass filter is not implemented, by the VCO that has a lowpass frequency response. Therefore the two schemes, with a PD or with a multiplier, may be viewed as equivalent; because of its simplicity, the latter will be considered in the following analysis.

14.2.1 PLL baseband model

We now derive a baseband equivalent model of the PLL. From (14.16) we have

$$u(t) = \int_0^t f(t-\xi)\, e(\xi)\, d\xi \tag{14.20}$$

substitution of (14.20) in (14.17) yields

$$\frac{d\hat{\varphi}_0(t)}{dt} = K_0 \int_0^t f(t-\xi)\, e(\xi)\, d\xi \tag{14.21}$$

By this relation we derive the baseband scheme of Figure 14.4. Subtraction of the phase estimate $\hat{\varphi}_0(t)$ from the phase $\varphi_0(t)$ yields the phase error $\phi(t)$ that, transformed by the non-linear block $K_D \sin(\cdot)$, in turn gives the signal $e(t)$. The signal $e(t)$ is input to the loop filter $F(s)$, which outputs the control signal $u(t)$. The integration block, with gain K_0, integrates the signal $u(t)$ and yields the estimate $\hat{\varphi}_0(t)$, thus closing the loop.

Substitution in (14.21) of the quantity $\varphi_0(t) - \phi(t)$ for $\hat{\varphi}_0(t)$, and of the expression (14.15) for $e(t)$, yields

$$\frac{d\phi(t)}{dt} = \frac{d\varphi_0(t)}{dt} - K_D\, K_0 \int_0^t f(t-\xi) \sin[\phi(\xi)]\, d\xi \tag{14.22}$$

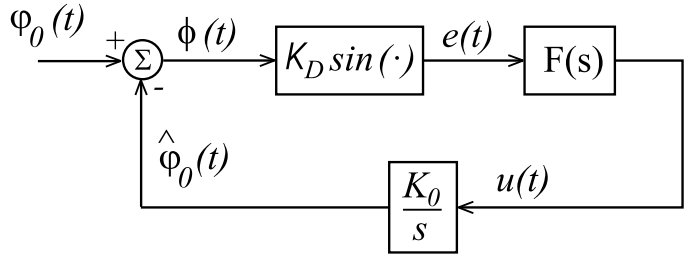


Figure 14.4. Baseband model of a PLL.

The (14.22) represents the integro-differential equation that governs the dynamics of the PLL. Later we will study this equation for particular expressions of the phase $\varphi_0(t)$, and only for the case $\phi(t) \simeq 0$, i.e. assuming the PLL is in the steady state or in the so-called *lock condition*; the transient behavior, that is for the case $\phi(t) \neq 0$, is difficult to analyze and we refer to [1] for further study.

Linear approximation

Assume that the phase error $\phi(t)$ is small, or $\phi(t) \simeq 0$; then the following approximation holds

$$\sin[\phi(t)] \simeq \phi(t) \tag{14.23}$$

and (14.22) simplifies into

$$\frac{d\phi(t)}{dt} = \frac{d\varphi_0(t)}{dt} - K_D\, K_0 \int_0^t f(t-\xi)\, \phi(\xi)\, d\xi \tag{14.24}$$

In this way the non-linear block $K_D \sin(\cdot)$ of Figure 14.4 becomes a multiplier by the constant K_D, and the whole structure is linear, as illustrated in the simplified block diagram of Figure 14.5.

We denote by $P_\phi(s)$ the Laplace transform of $\phi(t)$; by taking the Laplace transform of (14.24) and assuming $\hat{\varphi}_0(0) = 0$ we obtain

$$s\,P_\phi(s) = s\,\Phi_0(s) - K_D\, K_0\, F(s)\, P_\phi(s) \tag{14.25}$$

Substituting $P_\phi(s)$ with $\Phi_0(s) - \hat{\Phi}_0(s)$, we derive the *loop transfer function* as

$$H(s) = \frac{\hat{\Phi}_0(s)}{\Phi_0(s)} = \frac{K\,F(s)}{s + K\,F(s)} \qquad K = K_D\, K_0 \tag{14.26}$$

Then from (14.26) we get the following two relations:

$$P_\phi(s) = \Phi_0(s) - \hat{\Phi}_0(s) = [1 - H(s)]\, \Phi_0(s) \tag{14.27}$$

$$\frac{P_\phi(s)}{\Phi_0(s)} = \frac{1}{1 + [K\,F(s)/s]} \tag{14.28}$$

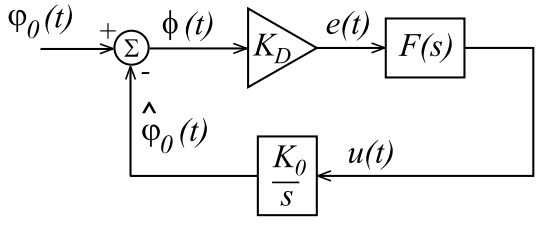


Figure 14.5. Linearized baseband model of the PLL.

Table 14.1 Three expressions of $\varphi_0(t)$ and corresponding Laplace transforms.

$\varphi_0(t)$	$\Phi_0(s)$
$\varphi_s\, 1(t)$	$\dfrac{\varphi_s}{s}$
$\omega_s t\, 1(t)$	$\dfrac{\omega_s}{s^2}$
$\omega_r \dfrac{t^2}{2} 1(t)$	$\dfrac{\omega_r}{s^3}$

We define as *steady state error* ϕ_∞ the limit for $t \to \infty$ of $\phi(t)$; recalling the final value theorem, and using (14.28), ϕ_∞ can be computed as follows:

$$\phi_\infty = \lim_{t\to\infty} \phi(t) = \lim_{s\to 0} s P_\phi(s) = \lim_{s\to 0} s\Phi_0(s) \frac{1}{1 + KF(s)/s} \tag{14.29}$$

We compute now the value of ϕ_∞ for the three expressions of $\varphi_0(t)$ given in Table 14.1 along with the corresponding Laplace transforms.

- *phase step*: $\varphi_0(t) = \varphi_s\, 1(t)$;

$$\begin{aligned} \phi_\infty &= \lim_{s\to 0} \left[s \frac{\varphi_s}{s} \frac{1}{1 + [KF(s)/s]} \right] \\ &= \lim_{s\to 0} \left[\frac{\varphi_s s}{s + KF(s)} \right] \end{aligned} \tag{14.30}$$

  thus we obtain

$$\phi_\infty = 0 \Longleftrightarrow F(0) \neq 0 \tag{14.31}$$

  Observe that (14.31) holds even if $F(s) = 1$, i.e. in case the loop filter is absent.

- *frequency step*: $\varphi_0(t) = \omega_s t\, 1(t)$;

$$\phi_\infty = \lim_{s\to 0} \left[s \frac{\omega_s}{s^2} \frac{1}{1 + [KF(s)/s]} \right] = \lim_{s\to 0} \left[\frac{\omega_s}{s + KF(s)} \right] \tag{14.32}$$

  If we choose

$$F(s) = s^{-k} F_1(s) \qquad \text{with } k \geq 1 \text{ and } 0 < |F_1(0)| < \infty \tag{14.33}$$

  then $\phi_\infty = 0$.

- *frequency ramp*: $\varphi_0(t) = (\omega_r t^2/2)\, 1(t)$

$$\phi_\infty = \lim_{s\to 0}\left[s\,\frac{\omega_r}{s^3}\,\frac{1}{1+[KF(s)/s]}\right] = \lim_{s\to 0}\left[\frac{\omega_r}{s^2+KF(s)s}\right] \tag{14.34}$$

If we use a loop filter of the type (14.33) with $k = 1$, i.e. with one pole at the origin, then we obtain a steady state error ϕ_∞ given by

$$\phi_\infty = \frac{\omega_r}{KF_1(0)} \neq 0 \tag{14.35}$$

As a general rule we can state that, in the presence of an input signal having Laplace transform of the type s^{-k} with $k \geq 1$, to get a steady state error $\phi_\infty = 0$, a filter with at least $(k-1)$ poles at the origin is needed.

The choice of the above elementary expressions of the phase $\varphi_0(t)$ for the analysis is justified by the fact that an arbitrary phase $\varphi_0(t)$ can always be approximated by a Taylor series expansion truncated to the second order, and therefore as a linear combination of the considered functions.

14.2.2 Analysis of the PLL in the presence of additive noise

We now extend the PLL baseband model and relative analysis to the case in which white noise $w(t)$ with spectral density $N_0/2$ is added to the signal $v(t)$. Introducing the in-phase and quadrature components of $w(t)$, from (1.162) we get the relation

$$w(t) = w_I(t)\cos(\omega_0 t) - w_Q(t)\sin(\omega_0 t) \tag{14.36}$$

where $w_I(t)$ and $w_Q(t)$ are two uncorrelated random processes having spectral density in the desired signal band given by

$$\mathcal{P}_{w_I}(f) = \mathcal{P}_{w_Q}(f) = N_0 \tag{14.37}$$

Letting

$$K_w = \tfrac{1}{2}\, A_2\, K_m \tag{14.38}$$

the multiplier output signal $e(t)$ assumes the expression

$$\begin{aligned} e(t) &= K_m[v(t)+w(t)]\, v_{VCO}(t) \\ &= K_D \sin[\phi(t)] + K_w\, w_Q(t)\sin[\hat{\varphi}_0(t)] + K_w\, w_I(t)\cos[\hat{\varphi}_0(t)] \\ &\quad + K_D \sin[2\omega_0 t + \varphi_0(t) + \hat{\varphi}_0(t)] - K_w\, w_Q(t)\sin[2\omega_0 t + \hat{\varphi}_0(t)] \\ &\quad + K_w\, w_I(t)\cos[2\omega_0 t + \hat{\varphi}_0(t)] \end{aligned} \tag{14.39}$$

Ignoring the high-frequency components in (14.39), (14.20) becomes

$$u(t) = \int_0^t f(t-\xi)\{K_D \sin[\phi(\xi)] + K_w\, w_Q(\xi)\sin[\hat{\varphi}_0(\xi)] + K_w\, w_I(\xi)\cos[\hat{\varphi}_0(\xi)]\}\, d\xi \tag{14.40}$$

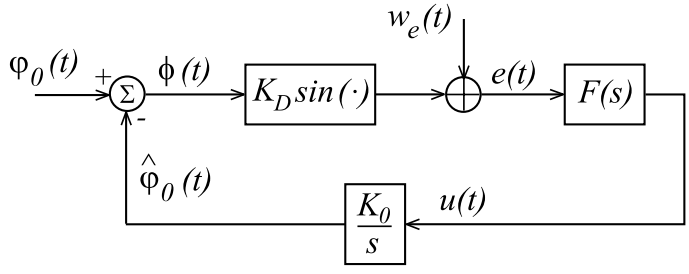


Figure 14.6. PLL baseband model in the presence of noise.

Defining the noise signal

$$w_e(t) = K_w[w_I(t)\sin\hat{\varphi}_0(t) + w_Q(t)\cos\hat{\varphi}_0(t)] \tag{14.41}$$

from (14.21) we get the integro-differential equation that describes the dynamics of the PLL in the presence of noise, expressed as

$$\frac{d\phi(t)}{dt} = \frac{d\varphi_0(t)}{dt} - K_0\int_0^t f(t-\xi)\{K_D\sin[\phi(\xi)] + w_e(\xi)\}\,d\xi \tag{14.42}$$

From (14.42) we obtain the PLL baseband model illustrated in Figure 14.6.

Noise analysis using the linearity assumption

In the case $\phi(t) \simeq 0$, we obtain the linearized PLL baseband model shown in Figure 14.7.

We now determine the contribution of the noise $w(t)$ to the phase error $\phi(t)$ in terms of variance of the phase error, σ_ϕ^2, assuming that the phase of the desired input signal is zero, or $\varphi_0(t) = 0$. From (14.14) we obtain

$$\hat{\varphi}_0(t) = -\phi(t) \tag{14.43}$$

Recalling that the transfer function of a filter that has $w_e(t)$ as input and $\hat{\varphi}_0(t)$ as output is given by $(1/K_D)\,H(s)$ (see (14.26)), the spectral density of $\phi(t)$ is given by

$$\mathcal{P}_\phi(f) = \mathcal{P}_{\hat{\varphi}_0}(f) = \frac{1}{K_D^2}\,|\mathcal{H}(f)|^2\,\mathcal{P}_{w_e}(f) \tag{14.44}$$

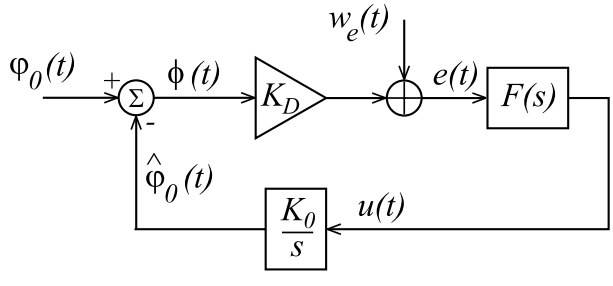


Figure 14.7. Linearized PLL baseband model in the presence of additive noise.

where

$$\mathcal{H}(f) = H(j2\pi f) \tag{14.45}$$

To obtain $\mathcal{P}_{w_e}(f)$ we use (14.41). Assuming $w_I(t)$ and $w_Q(t)$ are uncorrelated white random processes with autocorrelation

$$\mathtt{r}_{w_I}(\tau) = \mathtt{r}_{w_Q}(\tau) = N_0\,\delta(\tau) \tag{14.46}$$

and using the property of the Dirac function $\delta(\tau)\,f(t+\tau) = \delta(\tau)\,f(t)$, the autocorrelation of $w_e(t)$ turns out to be

$$\begin{aligned}\mathtt{r}_{w_e}(t, t-\tau) &= K_w^2\,\mathtt{r}_{w_I}(\tau)\{\sin[\hat{\varphi}_0(t)]\sin[\hat{\varphi}_0(t-\tau)] + \cos[\hat{\varphi}_0(t)]\cos[\hat{\varphi}_0(t-\tau)]\} \\ &= K_w^2\,N_0\,\delta(\tau)\{\sin^2[\hat{\varphi}_0(t)] + \cos^2[\hat{\varphi}_0(t)]\} \\ &= K_w^2\,N_0\,\delta(\tau)\end{aligned} \tag{14.47}$$

Taking the Fourier transform of (14.47) we obtain

$$\mathcal{P}_{w_e}(f) = K_w^2\,N_0 \tag{14.48}$$

Therefore using (14.18) and (14.38), from (14.44) we get the variance of the phase error, given by

$$\sigma_\phi^2 = \int_{-\infty}^{+\infty} \frac{1}{K_D^2}\,|\mathcal{H}(f)|^2\,\mathcal{P}_{w_e}(f)\,df = \frac{N_0}{A_1^2}\int_{-\infty}^{+\infty} |\mathcal{H}(f)|^2\,df \tag{14.49}$$

From (1.139) we now define the *equivalent noise bandwidth of the loop filter* as

$$B_L = \frac{\displaystyle\int_0^{+\infty} |\mathcal{H}(f)|^2\,df}{|\mathcal{H}(0)|^2} \tag{14.50}$$

Then (14.49) can be written as

$$\sigma_\phi^2 = \frac{2N_0B_L}{A_1^2} \tag{14.51}$$

where $A_1^2/2$ is the statistical power of the desired input signal, and $N_0B_L = (N_0/2)2B_L$ is the input noise power evaluated over a bandwidth B_L.

In Table 14.2 the expressions of B_L for different choices of the loop filter $F(s)$ are given.

14.2.3 Analysis of a second-order PLL

In this section we analyze the behavior of a second-order PLL, using the linearity assumption. In particular, we find the expression of the phase error $\phi(t)$ for the input signals given in Table 14.1, and we evaluate the variance of the phase error σ_ϕ^2.

Table 14.2 Expressions of B_L for different choices of the loop filter $F(s)$.

Loop order	$F(s)$	$H(s)$	B_L
First	1	$\dfrac{K}{s+K}$	$\dfrac{K}{4}$
Second	$\dfrac{s+a}{s}$	$\dfrac{K(s+a)}{s^2+Ks+Ka}$	$\dfrac{K+a}{4}$
Sec.-imperfect	$\dfrac{s+a}{s+b}$	$\dfrac{K(s+a)}{s^2+(K+b)s+Ka}$	$\dfrac{K(K+a)}{4(K+b)}$
Third	$\dfrac{s^2+as+b}{s^2}$	$\dfrac{K(s^2+as+b)}{s^3+Ks^2+aKs+bK}$	$\dfrac{K(aK+a^2-b)}{4(aK-b)}$

From Table 14.2 the transfer function of a second-order loop is given by

$$H(s) = \frac{K(s+a)}{s^2+Ks+Ka} \tag{14.52}$$

We define the *natural radian frequency*, ω_n, and the *damping factor of the loop*, ζ, as

$$\omega_n = \sqrt{Ka} \qquad \zeta = \frac{1}{2a}\sqrt{Ka} \tag{14.53}$$

As $K = 2\zeta\omega_n$ and $a = \omega_n/(2\zeta)$, (14.52) can be expressed as

$$H(s) = \frac{2\zeta\,\omega_n s + \omega_n^2}{s^2 + 2\zeta\,\omega_n s + \omega_n^2} \tag{14.54}$$

Once the expression of $\varphi_0(t)$ is known, $\phi(t)$ can be obtained by (14.27) and finding the inverse transform of $P_\phi(s)$. The relation is simplified if in place of s we introduce the normalized variable $\tilde{s} = s/\omega_n$; in this case we obtain

$$P_\phi(\tilde{s}) = \frac{\tilde{s}^2}{\tilde{s}^2 + 2\zeta\tilde{s} + 1}\,\Phi_0(\tilde{s}) \tag{14.55}$$

which depends only on the parameter ζ.

In Figures 14.8, 14.9, and 14.10 we show the plots of $\phi(t)$, with ζ as a parameter, for the three inputs of Table 14.1, respectively. Note that for the first two inputs $\phi(t)$ converges to zero, while for the third input it converges to a non-zero value (see (14.35)), because $F(s)$ has only one pole at the origin, as can be seen from Table 14.2. We note that if $\phi(t)$ is a phase step the speed of convergence increases with increasing ζ, whereas if $\phi(t)$ is a frequency step the speed of convergence is maximum for $\zeta = 1$.

In Figure 14.11 the plot of B_L is shown as a function of ζ. As

$$B_L = \omega_n\left(\frac{\zeta}{2} + \frac{1}{8\zeta}\right) \tag{14.56}$$

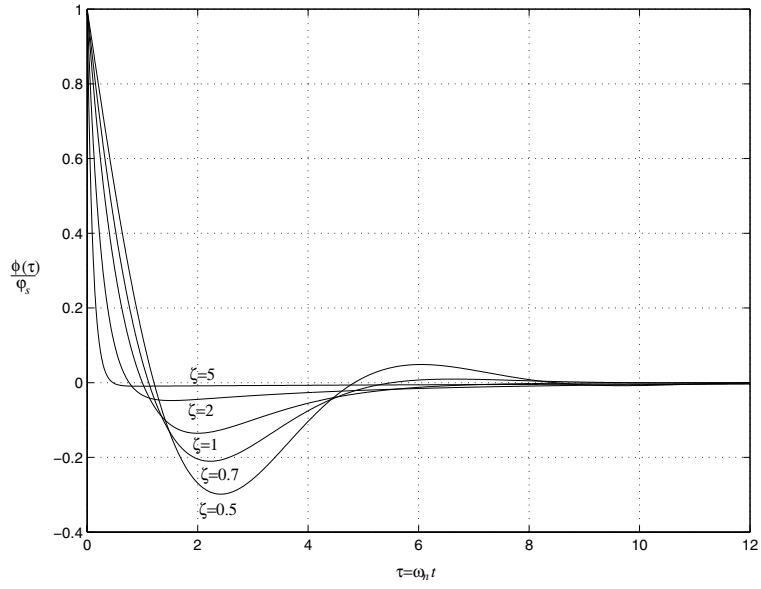


Figure 14.8. Plots of $\phi(\tau)$ as a function of $\tau = \omega_n t$, for a second-order loop filter with a *phase step input signal*: $\varphi_0(t) = \varphi_s 1(t)$.

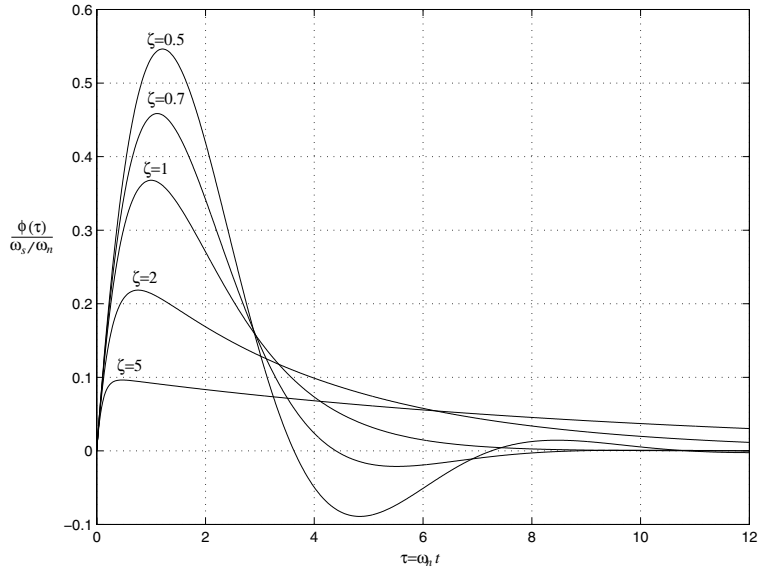


Figure 14.9. Plots of $\phi(\tau)$ as a function of $\tau = \omega_n t$, for a second-order loop filter with a *frequency step input signal*: $\varphi_0(t) = \omega_s t\, 1(t)$.

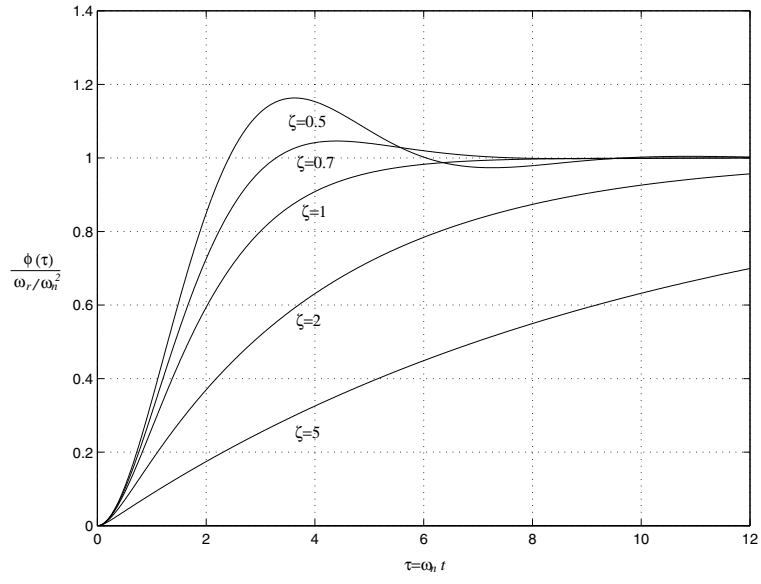


Figure 14.10. Plots of $\phi(\tau)$ as a function of $\tau = \omega_n t$, for a second-order loop filter with a *frequency ramp input signal*: $\varphi_0(t) = \omega_r(t^2/2)\,1(t)$.

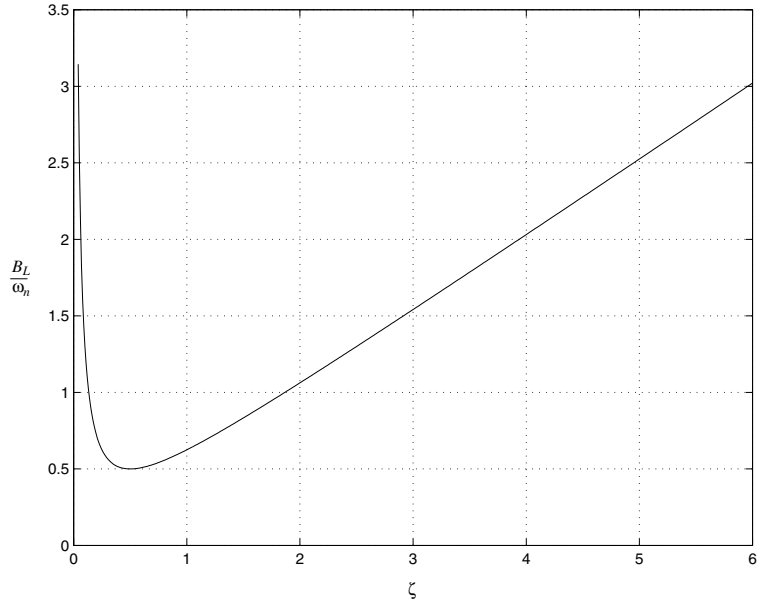


Figure 14.11. Plot of B_L as function of ζ for a second-order loop.

we note that B_L has a minimum for $\zeta = 0.5$, and that for $\zeta > 0.5$ it increases as $(1/2)\,\zeta$; the choice of ζ is therefore critical and represents a trade-off between the variance of the phase error and the speed of convergence.

For a detailed analysis of the second and third-order loops we refer to [1].

14.3 Costas loop

In the previous section, the PLL was presented as a structure capable of performing carrier recovery for signals of the type

$$s_{Ch}(t) = \sin(2\pi f_0 t + \varphi_0) \tag{14.57}$$

and, in general, for signals that contain periodic components of period n/f_0, with n positive integer.

We now discuss carrier recovery schemes for both PAM-DSB (see Appendix 7.C) and QAM (see Section 7.3) signals; these signals do not contain periodic components, but are *cyclostationary* and hence have periodic statistical moments.

We express the generic received signal $s_{Ch}(t)$, of PAM-DSB or QAM type, in terms of the complex envelope $s_{Ch}^{(bb)}(t)$, that for simplicity we denote by $a(t)$, as

$$s_{Ch}(t) = Re[a(t)\, e^{j(2\pi f_0 t + \varphi_0)}] \tag{14.58}$$

If in the reference carrier of the complex envelope we include also the phase φ_0, (7.39) becomes equal to (14.5), and

$$s_C(t) = \frac{1}{\sqrt{2}}\, s_{Ch}^{(bb)}(t) = \frac{a(t)}{\sqrt{2}} \tag{14.59}$$

The expression of $s_C(t)$, apart from the delay εT and the phase offset $e^{-j \arg \mathcal{G}_{Ch}(f_0)}$, is given by (7.42).

The autocorrelation of $s_{Ch}(t)$ is given by (see also (1.304))

$$\begin{aligned} \mathtt{r}_{s_{Ch}}(t, t-\tau) = {} & \tfrac{1}{2}\, Re[\mathtt{r}_a(t, t-\tau)\, e^{j2\pi f_0 \tau}] \\ & + \tfrac{1}{2}\, Re[\mathtt{r}_{aa^*}(t, t-\tau)\, e^{j[2\pi f_0 (2t-\tau) + 2\varphi_0]}] \end{aligned} \tag{14.60}$$

from which the statistical power is obtained as

$$\mathtt{M}_{s_{Ch}}(t) = \mathtt{r}_{s_{Ch}}(t, t) = \tfrac{1}{2}\, E[\,|a(t)|^2] + \tfrac{1}{2}\, Re\{E[a^2(t)]\, e^{j(4\pi f_0 t + 2\varphi_0)}\} \tag{14.61}$$

14.3.1 PAM signals

We assume that the channel frequency response $\mathcal{G}_C(f)$, obtained from (14.5), is Hermitian;[2] then $g_C(t)$ and $a(t)$ are real valued, hence

$$|a(t)|^2 = [a(t)]^2 \tag{14.62}$$

[2] In practice it is sufficient that $\mathcal{G}_C(f)$ is Hermitian in a small interval around $f = 0$ (see page 32).

and (14.61) becomes

$$\mathtt{M}_{s_{Ch}}(t) = \tfrac{1}{2}\,\mathtt{M}_a(t)[1 + \cos(4\pi f_0 t + 2\varphi_0)] \tag{14.63}$$

As $a(t)$ is a cyclostationary random process with period T (see Example 1.9.9 on page 69, $\mathtt{M}_a(t)$ is periodic of period T; therefore $\mathtt{M}_{s_{Ch}}(t)$ is also periodic and, assuming $1/T \ll f_0$, which is often verified in practice, its period is equal to $1/(2f_0)$.

Suppose the signal $s_{Ch}(t)$ is input to a device (*squarer*) that computes the square of the signal. The output of the squarer has a mean value (deterministic component) equal to the statistical power of $s_{Ch}(t)$, given by (14.63); if the squarer is cascaded with a narrow passband filter $\mathcal{H}_N(f)$ (see Figure 14.12), with $\mathcal{H}_N(2f_0) = 1$, then the mean value of the filter output is a sinusoidal signal with frequency $2f_0$, phase $2\varphi_0$ (in practice we need to sum also the phase introduced by the filter), and amplitude $(1/2)\,\mathtt{M}_a(t)$.

Assuming that $\mathcal{H}_N(f)$ completely suppresses the components at low frequencies, the output filter signal is given by

$$y(t) = \tfrac{1}{2}\,Re[(h_N^{(bb)} * a^2)(t)\, e^{\,j[4\pi f_0 t + 2\varphi_0]}] \tag{14.64}$$

This expression is obtained from the scheme of Figure 14.13 in which the product of two generic passband signals, $y_1(t)$ and $y_2(t)$, is expressed as a function of their complex envelopes, and decomposed into a baseband component $z_1(t)$ and a passband component $z_2(t)$ centered at $\pm 2f_0$.

If $\mathcal{H}_N(f)$ is Hermitian around the frequency $2f_0$, then $(h_N^{(bb)} * a^2)(t)$ is real valued,[3] and $y(t)$ can be written as

$$y(t) = \frac{(h_N^{(bb)} * a^2)(t)}{2}\,\cos(4\pi f_0 t + 2\varphi_0) \tag{14.65}$$

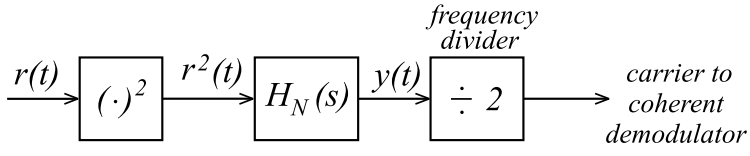


Figure 14.12. Carrier recovery in PAM-DSB systems.

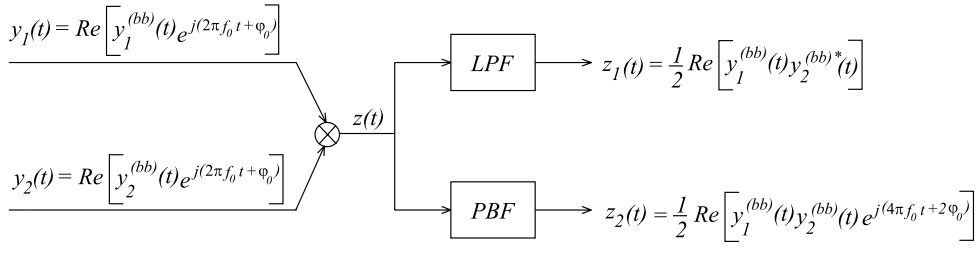


Figure 14.13. Baseband and passband components of the product of two generic passband signals, $y_1(t)$ and $y_2(t)$, as a function of their complex envelopes.

[3] In PAM-SSB and PAM-VSB systems, $(h_N^{(bb)} * a^2)(t)$ will also contain a quadrature component.

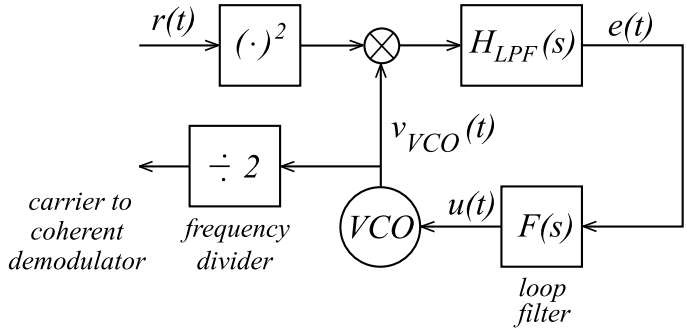


Figure 14.14. Squarer/PLL for carrier recovery in PAM-DSB systems.

Thus we have obtained a sinusoidal signal with frequency $2f_0$, phase $2\varphi_0$, and slowly varying amplitude, function of the bandwidth of $\mathcal{H}_N(f)$.

The carrier can be reconstructed by passing the signal $y(t)$ through a limiter, which eliminates the dependence on the amplitude, and then to a *frequency divider* that returns a sinusoidal signal with frequency and phase equal to half those of the square wave.

In the case of a time-varying phase $\varphi_0(t)$, the signal $y(t)$ can be sent to a PLL with a VCO that operates at frequency $2f_0$, and generates a reference signal equal to

$$v_{VCO}(t) = -A\sin(4\pi f_1 t + 2\varphi_1(t)) = -A\sin(4\pi f_0 t + 2\hat{\varphi}_0(t)) \tag{14.66}$$

The signal v_{VCO} must then be sent to a frequency divider to obtain the desired carrier. The block diagram of this structure is illustrated in Figure 14.14. Observe that the passband filter $\mathcal{H}_N(f)$ is substituted by the lowpass filter $\mathcal{H}_{LPF}(f)$, with $\mathcal{H}_{LPF}(0) = 1$, inserted in the feedback loop of the PLL; this structure is called the *squarer/PLL*.

An alternative tracking structure to the squarer/PLL, called a *Costas loop*, is shown in Figure 14.15. In a Costas loop the signal $e(t)$ is obtained by multiplying the I and Q components of the signal $r(t)$; the VCO directly operates at frequency f_0, thus eliminating the frequency divider, and generates the reconstructed carrier

$$v_{VCO}(t) = \sqrt{2A}\cos(2\pi f_0 t + \hat{\varphi}_0(t)) \tag{14.67}$$

By the equivalences of Figure 14.13 we find that the input of the loop filter is identical to that of the squarer/PLL, and is given by

$$e(t) = \frac{A}{4}a^2(t)\sin[2\phi(t)] \tag{14.68}$$

where ϕ is the phase error (14.14).

14.3.2 QAM signals

The schemes of Figures 14.12 and 14.14 cannot be directly applied to QAM systems. Indeed, the symmetry of the constellation of a QAM system usually leads to $E[a^2(t)] = 0$ (see (1.403)); therefore the periodic component in (14.61) is suppressed.

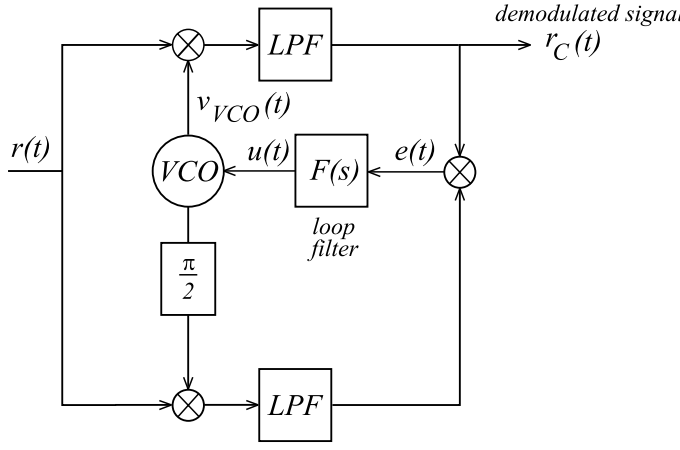


Figure 14.15. Costas loop for PAM-DSB systems.

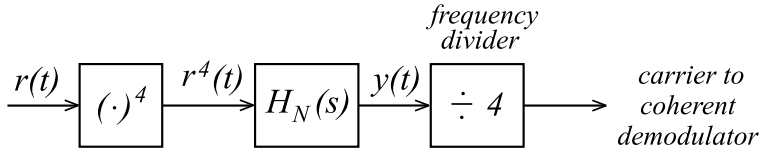


Figure 14.16. Carrier recovery in QAM systems.

We compute now the fourth power of $s_{Ch}(t)$; after a few steps we obtain

$$\begin{aligned} s_{Ch}^4(t) = & \tfrac{1}{8} Re[a^4(t) \exp(j8\pi f_0 t + j4\varphi_0)] \\ & + Re[|a(t)|^2 a^2(t) \exp(j4\pi f_0 t + j2\varphi_0)] \\ & + \tfrac{3}{8} |a(t)|^4 \end{aligned} \tag{14.69}$$

Filtering s_{Ch}^4 by a passband filter centered at $\pm 4 f_0$ (see Figure 14.16), eventually followed by a PLL in the case of a time-varying phase, we obtain a signal $y(t)$ having a mean value given by

$$E[y(t)] = \tfrac{1}{4} \{E[a_I^4(t)] - \mathtt{M}_{a_I}^2\} \cos(8\pi f_0 t + 4\varphi_0) \qquad a_I(t) = Re\,[a(t)] = s_{Ch,I}^{(bb)} \tag{14.70}$$

which is a periodic signal with period $1/(4 f_0)$ and phase $4\varphi_0$.

In the case of M-PSK signals, there exists a variant of the Costas loop called *extended Costas loop*; the scheme for a QPSK system is illustrated in Figure 14.17.

In the presence of additive noise, a passband filter centered at $\pm f_0$ is placed in front of all schemes to limit the noise without distorting the desired signal. For the performance analysis we refer to [2, 3], in which similar conclusions to those described in Section 14.2 for the PLL are reached.

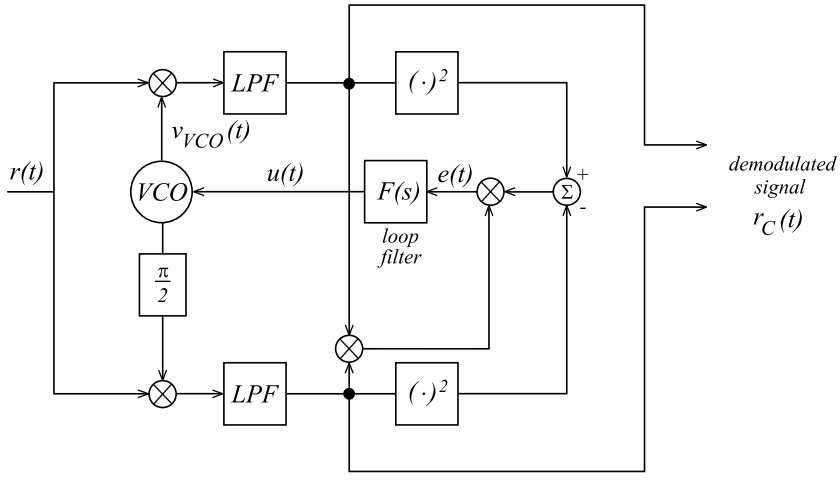


Figure 14.17. Extended Costas loop for QPSK systems.

14.4 The optimum receiver

With reference to the signal model (14.7), once the carrier has been recovered by one of the two methods described in the previous sections, after demodulation the three parameters $(\theta, \varepsilon, \Omega)$ need to be estimated. In this section the optimum receiver is obtained using the ML criterion discussed in Chapter 6 (see also (2.199)). The next two sections synthetically describe various estimation methods given in [4, 5].

From (14.7) the received signal $r_C(t)$ can be expressed as

$$\begin{aligned} r_C(t) &= e^{j(\Omega t+\theta)}\, s_C(t) + w_{C_\varphi}(t) \\ &= s_C(t;\theta,\varepsilon,\Omega) + w_{C_\varphi}(t) \end{aligned} \tag{14.71}$$

where $s_C(t)$ is given by[4]

$$s_C(t) = \sum_{k=-\infty}^{+\infty} a_k\, q_C(t - kT - \varepsilon T) \tag{14.72}$$

We express (14.71) using the vector notation, that is

$$\mathbf{r} = \mathbf{s} + \mathbf{w} \tag{14.73}$$

with the following assumptions:

1. the phase offset θ is equal to z,
2. the time shift εT is equal to eT,

[4] The phasor $e^{-j \arg \mathcal{G}_{Ch}(f_0)}$ is included in q_C.

3. the frequency offset Ω is equal to o,

4. the transmitted data sequence $\mathbf{a}$ is equal to $\boldsymbol{\alpha}$;

then the probability density function of $\mathbf{r}$ is given by (6.16), that is

$$p_{\mathbf{r}|\theta,\varepsilon,\Omega,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, o, \boldsymbol{\alpha}) = \mathcal{K} \exp\left(-\frac{1}{N_0} \|\boldsymbol{\rho} - \mathbf{s}\|^2\right) \tag{14.74}$$

The quantities $\|\boldsymbol{\rho}\|^2$ and $\|s\|^2$ are constants[5] [3]; therefore (14.74) is proportional to the likelihood

$$\mathrm{L}_{\theta,\varepsilon,\Omega,\mathbf{a}}(z, e, o, \boldsymbol{\alpha}) = \exp\left\{\frac{2}{N_0} Re[\langle \boldsymbol{\rho}, \mathbf{s} \rangle]\right\} \tag{14.75}$$

Referring to the transmission of K symbols or to a sufficiently large observation interval $T_K = KT$, (14.75) can be written as

$$\mathrm{L}_{\theta,\varepsilon,\Omega,\mathbf{a}}(z, e, o, \boldsymbol{\alpha}) = \exp\left\{Re\left[\frac{2}{N_0} \int_{T_K} \rho(t)\, s_C^*(t)\, e^{-j(ot+z)}\, dt\right]\right\} \tag{14.76}$$

Inserting the expression (14.72) of $s_C(t)$, limited to the transmission of K symbols, in (14.76) and interchanging the operations of summation and integration we obtain

$$\mathrm{L}_{\theta,\varepsilon,\Omega,\mathbf{a}}(z, e, o, \boldsymbol{\alpha}) = \exp\left\{\frac{2}{N_0} \sum_{k=0}^{K-1} Re\left[\alpha_k^*\, e^{-jz} \int_{T_K} \rho(t)\, e^{-jot}\, q_C^*(t - kT - eT)\, dt\right]\right\} \tag{14.77}$$

We introduce the matched filter[6]

$$g_M(t) = q_C^*(-t) \tag{14.78}$$

and assume that the pulse $(q_C * g_M)(t)$ is a Nyquist pulse; therefore there exists a suitable sampling phase for which ISI is avoided.

Finally, if we denote the integral in (14.77) by $x(kT + eT, o)$, that is

$$x(kT + eT, o) = (r_C(\tau)\, e^{-jo\tau} * g_M(\tau))(t)|_{t=kT+eT} \tag{14.79}$$

(14.77) becomes

$$\mathrm{L}_{\theta,\varepsilon,\Omega,\mathbf{a}}(z, e, o, \boldsymbol{\alpha}) = \exp\left\{\frac{2}{N_0} \sum_{k=0}^{K-1} Re[\alpha_k^*\, x(kT + eT, o)\, e^{-jz}]\right\} \tag{14.80}$$

Let us now suppose that the optimum values of z, e, o that maximize (14.80), i.e. the *estimates* of z, e, o have been determined in some manner.

---

[5] Here we are not interested in the detection of $\mathbf{a}$, as in the formulation of Section 8.10, but rather in the estimate of the parameters θ, ε, and Ω; in this case, if the observation is sufficiently long we can assume that $\|s\|^2$ is invariant with respect to the different parameters.

[6] In this formulation, the filter g_M is anticausal; in practice a delay equal to the duration of q_C must be taken into account.

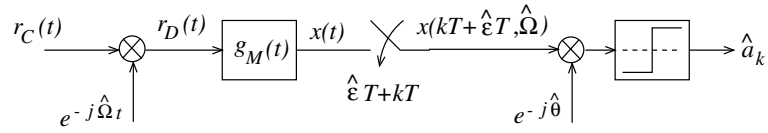


Figure 14.18. Analog receiver for QAM systems.

The structure of the optimum receiver derived from (14.80) is illustrated in Figure 14.18. The signal $r_C(t)$ is multiplied by $\exp\{-j\hat{\Omega}t\}$, where $\hat{\Omega}$ is an estimate of Ω, to remove the frequency offset, then filtered by the matched filter $g_M(t)$ and sampled at the sampling instants $kT + \hat{\varepsilon}T$, where $\hat{\varepsilon}$ is an estimate of ε. The samples $x(kT + \hat{\varepsilon}T, \hat{\Omega})$ are then multiplied by $\exp\{-j\hat{\theta}\}$ to remove the phase offset. Finally, the data detector decides on the symbol $\hat{a}_k$ that, in the absence of ISI, maximizes the k-th term of the summation in (14.80) evaluated for $(z, e, o) = (\hat{\theta}, \hat{\varepsilon}, \hat{\Omega})$:

$$\hat{a}_k = \arg\max_{\alpha_k} Re[\alpha_k^* \, x(kT + \hat{\varepsilon}T, \hat{\Omega}) \, e^{-j\hat{\theta}}] \tag{14.81}$$

The digital version of the scheme of Figure 14.18 is illustrated in Figure 14.19; it uses an anti-aliasing filter and a sampler with period T_c such that (recall the sampling theorem on page 30)

$$\frac{1}{2T_c} \geq B + \frac{\Omega_{max}}{2\pi} \tag{14.82}$$

Observation 14.1
To simplify the implementation of the digital receiver, the ratio $F_0 = T/T_c$ is chosen as an integer; in this case, for $F_0 = 4$ or 8, the interpolator filter may be omitted and the timing after the matched filter has a precision of $T_c = T/F_0$. This approach is usually adopted in radio systems for the transmission of packet data.

To conclude this section, we briefly discuss the algorithms for timing and carrier phase recovery.

Timing recovery

Ideally, at the output of the anti-aliasing filter the received signal should be sampled at the instants $t = kT + \varepsilon T$; however, there are two problems:

1. the value of ε is not known;
2. the clock at the receiver allows sampling at multiples of T_c, not at multiples of T, and the ratio T/T_c is not necessarily a rational number.

Therefore time synchronization methods are usually composed of two basic functions.

1) *Timing estimate.* The first function gives an estimate $\hat{\varepsilon}$ of ε.

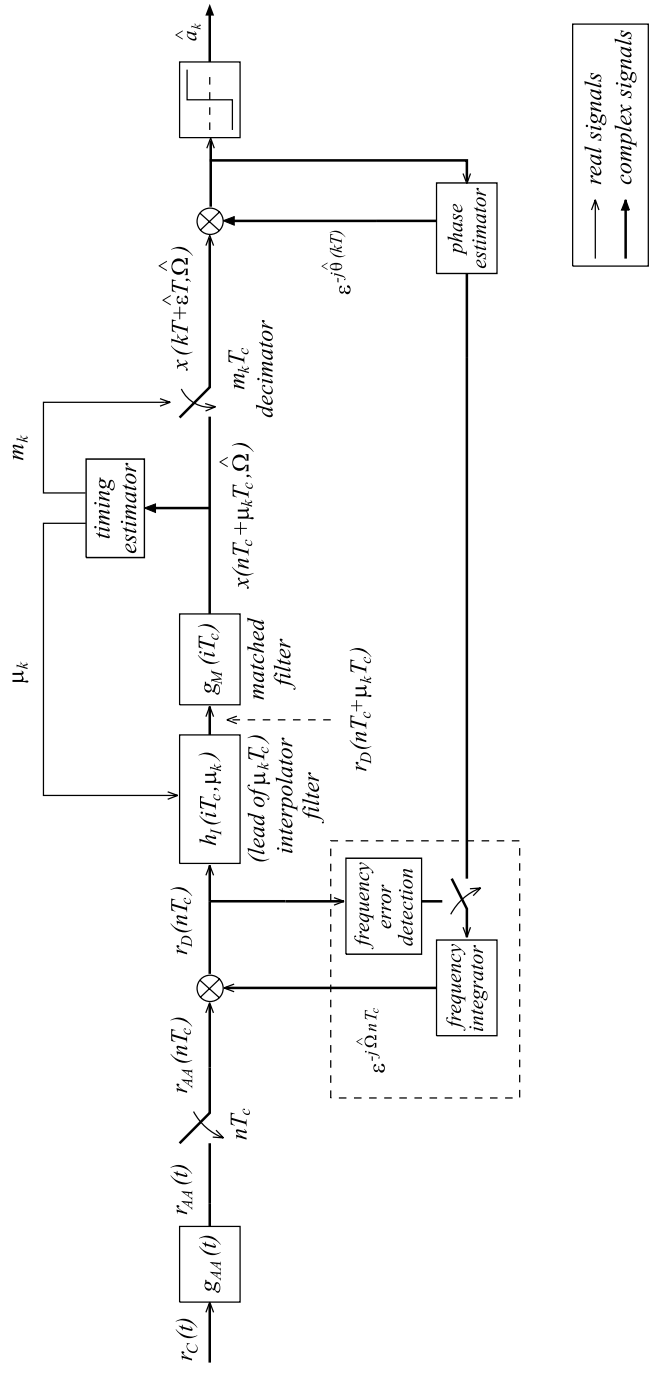


Figure 14.19. Digital receiver for QAM systems. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

2) *Interpolation and decimation.* The sampling instants $t = kT + \hat{\varepsilon}T$ can be written as

$$kT + \hat{\varepsilon}T = \left[k\,\frac{T}{T_c} + \hat{\varepsilon}\,\frac{T}{T_c} \right] T_c \tag{14.83}$$

The expression within brackets admits the following decomposition:

$$\begin{aligned} \left[k\,\frac{T}{T_c} + \hat{\varepsilon}\,\frac{T}{T_c} \right] &= \left\lfloor \left(k\,\frac{T}{T_c} + \hat{\varepsilon}\,\frac{T}{T_c} \right) \right\rfloor + \mu_k \\ &= \mathrm{m}_k + \mu_k \end{aligned} \tag{14.84}$$

Given a real number a, $\lfloor a \rfloor$ denotes the largest integer smaller than or equal to a (floor), and $\mu = a - \lfloor a \rfloor$ is the fractionary part that we denote as $[a]_F$. Suppose now that the estimate of ε is time varying; we denote it by $\hat{\varepsilon}_k$. Consider the $(k+1)$-th sampling instant, expressed as

$$(k+1)\,T + \hat{\varepsilon}_{k+1}T \tag{14.85}$$

By adding and subtracting $\hat{\varepsilon}_k T$, (14.85) can be rewritten as follows:

$$(k+1)\,T + \hat{\varepsilon}_{k+1}T = kT + \hat{\varepsilon}_k T + T + (\hat{\varepsilon}_{k+1} - \hat{\varepsilon}_k)\,T \tag{14.86}$$

Substituting $kT + \hat{\varepsilon}_k T$ with $\mathrm{m}_k T_c + \mu_k T_c$, we obtain

$$\begin{aligned} (k+1)\,T + \hat{\varepsilon}_{k+1}T &= \mathrm{m}_k T_c + \mu_k T_c + T + (\hat{\varepsilon}_{k+1} - \hat{\varepsilon}_k)\,T \\ &= \left[\mathrm{m}_k + \mu_k + \frac{T}{T_c}(1 + (\hat{\varepsilon}_{k+1} - \hat{\varepsilon}_k)) \right] T_c \end{aligned} \tag{14.87}$$

Recalling that m_k is a positive integer, and μ_k is real valued and belongs to the interval [0,1), from (14.87) the following recursive expressions for m_k and μ_k are obtained:

$$\begin{aligned} \mathrm{m}_{k+1} &= \mathrm{m}_k + \left\lfloor \mu_k + \frac{T}{T_c}(1 + (\hat{\varepsilon}_{k+1} - \hat{\varepsilon}_k)) \right\rfloor \\ \mu_{k+1} &= \left[\mu_k + \frac{T}{T_c}(1 + (\hat{\varepsilon}_{k+1} - \hat{\varepsilon}_k)) \right]_F \end{aligned} \tag{14.88}$$

The quantity $\mathrm{saw}(\hat{\varepsilon}_{k+1} - \hat{\varepsilon}_k)$ is often substituted for $(\hat{\varepsilon}_{k+1} - \hat{\varepsilon}_k)$, where $\mathrm{saw}(x)$ is the saw-tooth function illustrated in Figure 14.20. Thus, the difference between two successive estimates belongs to the interval $[-1/2, 1/2]$; this choice reduces the effects that a wrong estimate of ε would have on the value of the pair (m_k, μ_k).

Figure 14.21 illustrates the graphic representation of (14.84) in the ideal case $\hat{\varepsilon} = \varepsilon$. The transmitter time scale, defined by multiples of T, is shifted by a constant quantity equal to εT. The receiver time scale is defined by multiples of T_c. The fact that the ratio T/T_c may be a non-rational number has two consequences: first, the time shift $\mu_k T_c$ is time varying even if εT is a constant; second, the instants $\mathrm{m}_k T_c$ form a non-uniform subset of the receiver time axis, such that on average the considered samples are separated by an interval T.

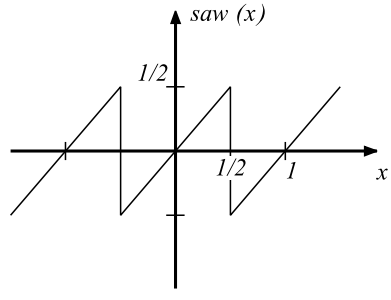


Figure 14.20. Plot of the saw-tooth function saw(x).

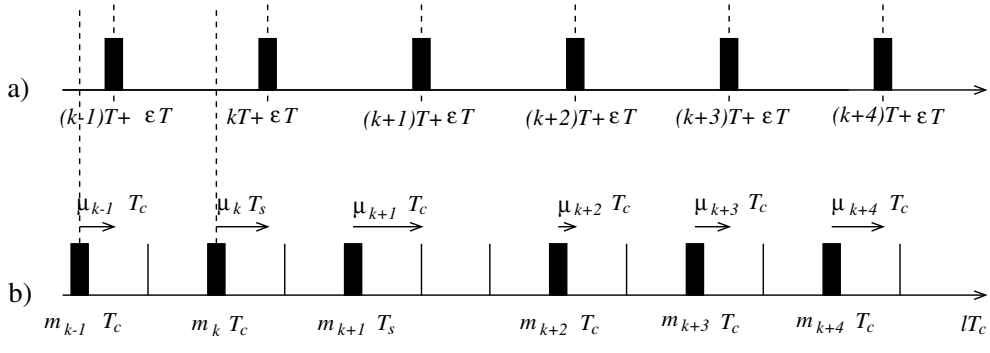


Figure 14.21. (a) Transmitter time scale; (b) receiver time scale with $T_c < T$, for the ideal case $\hat{\varepsilon} = \varepsilon$. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

With reference to Figure 14.19, to obtain the samples of the signal $x(t)$ at the instants $(\mathrm{m}_k + \mu_k)\, T_c$ we can proceed as follows:

a) implement a digital interpolator filter that provides samples of the received signal at the instants $(n+\mu_k)\, T_c$, starting from samples at the instants nT_c (see Section 1.A.5);

b) implement a downsampler that yields samples at the instants $(\mathrm{m}_k+\mu_k)\, T_c = kT+\hat{\varepsilon}T$.

With regard to the digital interpolator filter, consider a signal $r_D(t)$ with bandwidth $B_{r_D} \leq 1/(2T_c)$. From the sampling theorem, the signal $r_D(t)$ can be reconstructed from its samples $r_D(iT_c)$ using the relation

$$r_D(t) = \sum_{i=-\infty}^{+\infty} r_D(iT_c)\,\mathrm{sinc}\left(\frac{t-iT_c}{T_c}\right) \tag{14.89}$$

This expression is valid for all t; in particular it is valid for $t = t_1 + \mu_k T_c$, thus yielding the signal $r_D(t_1 + \mu_k T_c)$. Sampling this signal at $t_1 = nT_c$ we obtain

$$r_D(nT_c + \mu_k T_c) = \sum_{i=-\infty}^{+\infty} r_D(iT_c)\,\mathrm{sinc}(n + \mu_k - i) \tag{14.90}$$

Observe that the second member of (14.90) is a discrete-time convolution; in fact, introducing the interpolator filter with impulse response h_I and parameter μ_k,

$$h_I(iT_c;\mu_k) = \text{sinc}(i+\mu_k) \qquad i = -\infty, \ldots, +\infty \tag{14.91}$$

(14.90) can be rewritten as

$$r_D(nT_c + \mu_k T_c) = [r_D(iT_c) * h_I(iT_c;\mu_k)]\,(nT_c) \tag{14.92}$$

In other words, to obtain from samples of $r_D(t)$ at instants nT_c the samples at $nT_c + \mu_k T_c$ we can use a filter with impulse response $h_I(iT_c;\mu_k)$.[7]

With regard to the cascade of the matched filter $g_M(iT_c)$ and the decimator at instants $\mathrm{m}_k T_c$, we point out that a more efficient solution is to implement a filter with input at instants nT_c that generates output samples only at instants $\mathrm{m}_k T_c$.

We conclude this section by recalling that, if after the matched filter g_M, or directly in place of g_M, there is an equalizer filter c with input signal having sampling period equal to T_c, the function of the filter h_I is performed by the filter c itself (see Section 8.4).

Carrier phase recovery

An offset of the carrier phase equal to θ has the effect of rotating the complex symbols by $\exp(j\theta)$; this error can be corrected by multiplying the matched filter output by $\exp(-j\hat{\theta})$, where $\hat{\theta}$ is an estimate of θ.

Carrier phase recovery consists of three basic functions:

1) *Phase estimate.* In the scheme of Figure 14.19, phase estimation is performed after the matched filter, using samples with sampling period equal to the symbol period T. In this scheme, timing recovery is implemented before phase recovery, and must operate in one of the following modes:

a) with an arbitrary phase offset;

b) with a phase estimate anticipating the multiplication by $e^{-j\hat{\theta}}$ after the decimator;

c) jointly recovering phase and timing.

---

[7] In practice, the filter impulse response $h_I(\cdot;\mu_k)$ must have a finite number N of coefficients. The choice of N depends on the ratio T/T_c and on the desired precision; for example, for $T/T_c = 2$ and a normalized MSE, given by

$$J = 2T \int_0^1 \int_{-1/(2T)}^{1/(2T)} |e^{j2\pi f\mu} - \mathcal{H}_I(f;\mu)|^2 \, df \, d\mu \tag{14.93}$$

where

$$\mathcal{H}_I(f;\mu) = \sum_{i=-(N/2)+1}^{N/2} h_I(iT_c;\mu)\, e^{-j2\pi f i T_c} \tag{14.94}$$

equal to -50 dB, it turns out $N \simeq 5$. Of course, more efficient interpolator filters than that defined by (14.91) can be utilized.

2) *Phase rotation.*

a) The samples $x(kT + \hat{\varepsilon}T, \hat{\Omega})$ are multiplied by the complex signal $\exp(-j\hat{\theta}(kT))$ (see Figure 14.19); a possible residual frequency offset $\Delta\Omega$ can be corrected by a time-varying phase given by $\hat{\theta}(kT) = \hat{\theta} + kT\widehat{\Delta\Omega}$.

b) The samples $x(kT + \hat{\varepsilon}T, \hat{\Omega})\, e^{-j\hat{\theta}}$ are input to the data detector, assuming $(\hat{\theta}, \hat{\varepsilon})$ are the true values of (θ, ε).

3) *Frequency synchronization.* A first coarse estimate of the frequency offset needs to be performed in the analog domain. In fact, algorithms for timing and phase recovery only work in the presence of a small residual frequency offset. A second block provides a fine estimate of this residual offset that is used for frequency offset compensation.

14.5 Algorithms for timing and carrier phase recovery

In this section we discuss digital algorithms to estimate the time shift and the carrier phase offset under the assumption of absence of frequency offset, or $\Omega = 0$. Thus, the output samples of the decimator of Figure 14.19 are expressed as $x(kT + \hat{\varepsilon}T, 0)$; they will be simply denoted as $x(kT + \hat{\varepsilon}T)$, or in compact notation as $x_k(\hat{\varepsilon})$.

14.5.1 ML criterion

The expression of the likelihood is obtained from (14.80) assuming $o = 0$, that is

$$\begin{aligned} \mathrm{L}_{\theta,\varepsilon,\mathbf{a}}(z, e, \boldsymbol{\alpha}) &= \exp\left\{\frac{2}{N_0}\sum_{k=0}^{K-1} Re[\alpha_k^*\, x_k(e)\, e^{-jz}]\right\} \\ &= \prod_{k=0}^{K-1} \exp\left\{\frac{2}{N_0} Re[\alpha_k^*\, x_k(e)\, e^{-jz}]\right\} \end{aligned} \tag{14.95}$$

Assumption of slow time varying channel

In general both the time shift and the phase offset are time varying; from now on we assume that the rate at which these parameters vary is much lower than the symbol rate $1/T$. Thus, it is useful to consider two time scales: one that refers to the symbol period T, for symbol detection and estimation of synchronization parameters, and the other that refers to a period much larger than T, for the variation of the synchronization parameters.

14.5.2 Taxonomy of algorithms using the ML criterion

Synchronization algorithms are obtained by the ML criterion (see (14.77) or equivalently (14.74)) averaging the probability density function $p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \boldsymbol{\alpha})$ with respect to parameters that do not need to be estimated. Then we obtain the following likelihood functions:

- for the joint estimate of (θ, ε),

$$p_{\mathbf{r}|\theta,\varepsilon}(\boldsymbol{\rho} \mid z, e) = \sum_{\boldsymbol{\alpha}} p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \boldsymbol{\alpha})\, P[\mathbf{a} = \boldsymbol{\alpha}] \tag{14.96}$$

- for the estimate of the phase,

$$p_{\mathbf{r}|\theta}(\boldsymbol{\rho} \mid z) = \int \left[\sum_{\boldsymbol{\alpha}} p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \boldsymbol{\alpha})\, P[\mathbf{a} = \boldsymbol{\alpha}]\, p_{\varepsilon}(e) \right] de \tag{14.97}$$

- for the estimate of timing,

$$p_{\mathbf{r}|\varepsilon}(\boldsymbol{\rho} \mid e) = \int \left[\sum_{\boldsymbol{\alpha}} p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \boldsymbol{\alpha})\, P[\mathbf{a} = \boldsymbol{\alpha}]\, p_{\theta}(z) \right] dz \tag{14.98}$$

With the exception of some special cases, the above functions cannot be computed in close form. Consequently we need to develop appropriate approximation techniques.

A first classification of synchronization algorithms is based on whether knowledge of the data sequence is available or not; in this case we distinguish two classes:

1. *decision-directed* (DD) or *data-aided* (DA);
2. *non-data aided* (NDA).

If the data sequence is known, for example, by sending a training sequence $\mathbf{a} = \boldsymbol{\alpha}_0$ during the acquisition phase, we speak of *data-aided* algorithms. As the sequence $\mathbf{a}$ is known, in the sum in the expression of the likelihood function only the term for $\boldsymbol{\alpha} = \boldsymbol{\alpha}_0$ remains. For example, the joint estimate of (θ, ε) reduces to the maximization of the likelihood $p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \boldsymbol{\alpha}_0)$, and we get

$$(\hat{\theta}, \hat{\varepsilon})_{DA} = \arg\max_{z,e}\ p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \boldsymbol{\alpha}_0) \tag{14.99}$$

On the other hand, whenever we use the detected sequence $\hat{\mathbf{a}}$ as if it were the true sequence $\mathbf{a}$, we speak of *data-directed* algorithms. If there is a high probability that $\hat{\mathbf{a}} = \mathbf{a}$, again in the sum in the expression of the likelihood function only one term remains. Taking again the joint estimate of (θ, ε) as an example, in (14.96) the sum reduces to

$$\sum_{\boldsymbol{\alpha}} p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \boldsymbol{\alpha})\, P[\mathbf{a} = \boldsymbol{\alpha}] \simeq p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \hat{\mathbf{a}}) \tag{14.100}$$

as $P[\mathbf{a} = \hat{\mathbf{a}}] \simeq 1$. The joint estimate (θ, ε) is thus given by

$$(\hat{\theta}, \hat{\varepsilon})_{DD} = \arg\max_{z,e}\ p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \hat{\mathbf{a}}) \tag{14.101}$$

Non-data aided algorithms apply instead, in an exact or approximate fashion, the averaging operation with respect to the data sequence.

A second classification of synchronization algorithms is made based on the synchronization parameters that must be eliminated; then we have four cases:

1. DD & Dε: *data and timing directed*,

$$p_{\mathbf{r}|\theta}(\boldsymbol{\rho} \mid z) = p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, \hat{\varepsilon}, \hat{\mathbf{a}}) \tag{14.102}$$

2. DD, *timing independent*,

$$p_{\mathbf{r}|\theta}(\boldsymbol{\rho} \mid z) = \int p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \hat{\mathbf{a}})\, p_{\varepsilon}(e)\, de \tag{14.103}$$

3. DD & Dθ: *data and phase directed*,

$$p_{\mathbf{r}|\varepsilon}(\boldsymbol{\rho} \mid e) = p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid \hat{\theta}, e, \hat{\mathbf{a}}) \tag{14.104}$$

4. DD, *phase independent or non-coherent* (NC),

$$p_{\mathbf{r}|\varepsilon}(\boldsymbol{\rho} \mid e) = \int p_{\mathbf{r}|\theta,\varepsilon,\mathbf{a}}(\boldsymbol{\rho} \mid z, e, \hat{\mathbf{a}})\, p_{\theta}(z)\, dz \tag{14.105}$$

A further classification is based on the method for obtaining the timing and phase estimates:

1. *feedforward* (FF). The FF algorithms directly estimate the parameters (θ, ε) without using signals that are modified by the estimates; this implies using signals before the interpolator filter for timing recovery and before the phase rotator (see Figure 14.19) for carrier phase recovery.

2. *feedback* (FB). The FB algorithms estimate the parameters (θ, ε) using also signals that are modified by the estimates; in particular they yield an estimate of the errors $e_\theta = \hat{\theta} - \theta$ and $e_\varepsilon = \hat{\varepsilon} - \varepsilon$, which are then used to control the interpolator filter and the phase rotator, respectively. In general, feedback structures are able to track slow changes of parameters.

Next, we give a brief description of FB estimators, with emphasis on the fundamental blocks and on input–output relations.

Feedback estimators

In Figure 14.22 the block diagrams of a FB phase (FBθ) estimator and of a FB timing (FBε) estimator are illustrated. These schemes can be easily extended to the case of a FB frequency offset estimator. The two schemes only differ in the first block. In the case of the FBθ estimator, the first block is a phase rotator that, given the input signal $s(kT)$ ($x(kT + \hat{\varepsilon}T)$ in Figure 14.19), yields $s(kT)\exp(-j\hat{\theta}_k)$, where $\hat{\theta}_k$ is the estimate of θ at instant kT; in the case of the FBε estimator, it is an interpolator filter that, given the input signal $s(kT)$ ($r_D(nT_c)$ in Figure 14.19), returns $s(kT+\hat{\varepsilon}_k T)$ ($r_D(kT+\hat{\varepsilon}T)$ in Figure 14.19), where $\hat{\varepsilon}_k$ is the estimate of ε at instant kT.

We analyze only the FBθ estimator, as the FBε estimator is similar. The error detector block is the fundamental block and has the function of generating a signal $e(kT)$, called *error signal*, whose *mean value is written as*

$$E[e(kT)] = g_\theta(\theta - \hat{\theta}_k) \tag{14.106}$$

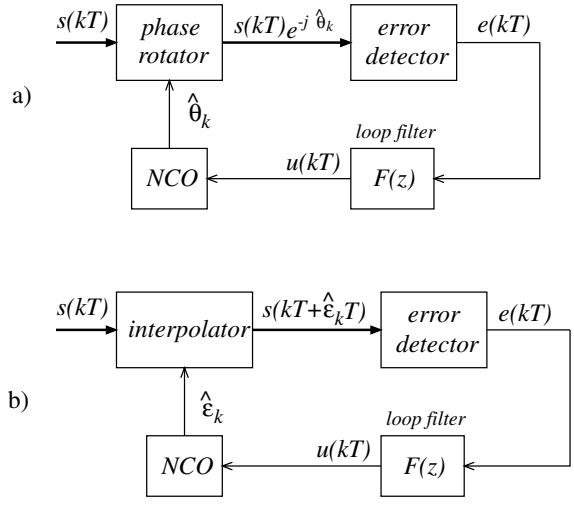


Figure 14.22. (a) Feedback estimator of θ; (b) feedback estimator of ε.

where $g_\theta(\cdot)$ is an odd function. Then the error signal $e(kT)$ admits the following decomposition

$$e(kT) = g_\theta(\theta - \hat{\theta}_k) + \eta_\theta(kT; \theta; \hat{\theta}_k) \tag{14.107}$$

where $\eta_\theta(\cdot)$ is a disturbance term called *loop noise*; the loop filter $F(z)$ is a lowpass filter that has two tasks: it regulates the speed of convergence of the estimator and mitigates the effects of the loop noise. The loop filter output, $u(kT)$, is input to the *numerically controlled oscillator* (NCO), that updates the phase estimate θ according to the following recursive relation:

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \mu_\theta \, u(kT) \tag{14.108}$$

where μ_θ denotes the NCO gain.

In most FB estimators the error signal $e(kT)$ is obtained as the derivative of the likelihood, or of its logarithm, with respect to the parameter to be estimated, evaluated using the most recent estimates $\hat{\theta}_k$, $\hat{\varepsilon}_k$. In particular, we have the two cases

$$\begin{aligned} e(kT) &\propto \frac{\partial}{\partial e} \mathrm{L}_{\theta,\varepsilon,\mathbf{a}}(\hat{\theta}_k, e, \boldsymbol{\alpha})\Big|_{e=\hat{\varepsilon}_k} \\ e(kT) &\propto \frac{\partial}{\partial z} \mathrm{L}_{\theta,\varepsilon,\mathbf{a}}(z, \hat{\varepsilon}_k, \boldsymbol{\alpha})\Big|_{z=\hat{\theta}_k} \end{aligned} \tag{14.109}$$

We note that in FB estimators the vector $\mathbf{a}$ represents the transmitted symbols from instant 0 up to the instant corresponding to the estimates $\hat{\theta}_k$, $\hat{\varepsilon}_k$.

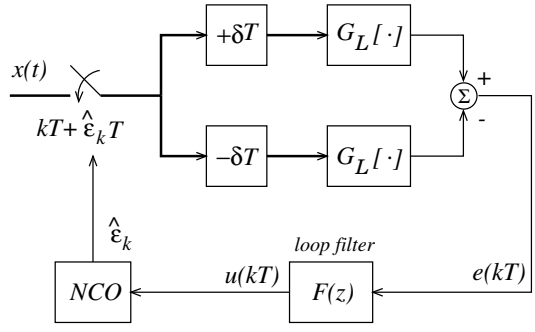


Figure 14.23. FB early-late timing estimator.

Early-late estimators

Early-late estimators constitute a subclass of FB estimators, where the error signal is computed according to (14.109), and the derivative operation is approximated by a finite difference [3], i.e., given a signal $p(t)$, its derivative is computed as follows:

$$\frac{dp(t)}{dt} \simeq \frac{p(t+\delta) - p(t-\delta)}{2\delta} \tag{14.110}$$

where δ is a positive real number.

Consider, for example, a DD & Dθ timing estimator and denote by $G_{\mathrm{L}}[\cdot]$ the function that, given the input signal $x_k(\hat{\varepsilon}_k)$, yields the likelihood $\mathtt{L}_{\varepsilon}(e)$ evaluated at $e = \hat{\varepsilon}_k$. From (14.110) the error signal is given by

$$e(kT) \propto G_{\mathrm{L}}[x_k(\hat{\varepsilon}_k + \delta)] - G_{\mathrm{L}}[x_k(\hat{\varepsilon}_k - \delta)] \tag{14.111}$$

The block diagram of Figure 14.22b is modified into that of Figure 14.23. Observe that in the upper branch the signal $x_k(\hat{\varepsilon}_k)$ is anticipated of δT, while in the lower branch it is delayed by $-\delta T$: hence the name of *early-late estimator*.

14.5.3 Timing estimators

Non-data aided

The estimate of ε can be obtained from (14.95) by eliminating the dependence on the parameters $\boldsymbol{\alpha}$ and z that are not estimated. To remove this dependence, we take the expectation with respect to $\mathbf{a}$; assuming i.i.d. symbols we obtain the likelihood

$$\mathtt{L}_{\theta,\varepsilon}(z, e) = \prod_{k=0}^{K-1} E_{a_k}\left[\exp\left\{\frac{2}{N_0} Re[a_k^* \, x_k(e)\, e^{-jz}]\right\}\right] \tag{14.112}$$

where E_{a_k} denotes the expectation with respect to a_k.

For an M-PSK signal, with $M > 2$, we approximate a_k as $e^{j\varphi_k}$, where φ_k is a uniform r.v. on $(-\pi;\pi]$; then (14.112) becomes

$$\mathtt{L}_{\theta,\varepsilon}(z,e) = \prod_{k=0}^{K-1} \int_{-\pi}^{+\pi} \exp\left\{ \frac{2}{N_0} Re[e^{-j v_k}\, x_k(e)\, e^{-jz}] \frac{dv_k}{2\pi} \right\} \tag{14.113}$$

If we use the definition of the Bessel function (4.216), (14.113) is independent of the phase θ and we obtain

$$\mathtt{L}_{\varepsilon}(e) = \prod_{k=0}^{K-1} I_0\left(\frac{|x_k(e)|}{N_0/2} \right) \tag{14.114}$$

On the other hand, if we take the expectation of (14.95) only with respect to the phase θ we obtain

$$\mathtt{L}_{\varepsilon,\mathbf{a}}(e,\boldsymbol{\alpha}) = \prod_{k=0}^{K-1} I_0\left(\frac{|x_k(e)\,\alpha_k^*|}{N_0/2} \right) \tag{14.115}$$

We observe that, for M-PSK, $\mathtt{L}_{\varepsilon,\mathbf{a}}(e,\boldsymbol{\alpha}) = \mathtt{L}_{\varepsilon}(e)$, as $|\alpha_k|$ is a constant, while this does not occur for M-QAM.

To obtain estimates from the two likelihood functions just obtained, if the signal-to-noise ratio Γ is sufficiently high, we utilize the fact that $I_0(\cdot)$ can be approximated as

$$I_0(\zeta) \simeq 1 + \frac{\zeta^2}{2} \qquad \text{for } |\zeta| \ll 1 \tag{14.116}$$

Taking the logarithm of the likelihood and eliminating non-relevant terms, we obtain the following NDA estimator and DA estimator.

$$\begin{aligned} \text{NDA:} \qquad \hat{\varepsilon} &= \arg\max_e \ln\{\mathtt{L}_{\varepsilon}(e)\} \\ &\simeq \arg\max_e \sum_{k=0}^{K-1} |x_k(e)|^2 \end{aligned} \tag{14.117}$$

$$\begin{aligned} \text{DA:} \qquad \hat{\varepsilon} &= \arg\max_e \ln\{\mathtt{L}_{\varepsilon,\mathbf{a}}(e,\boldsymbol{\alpha})\} \\ &\simeq \arg\max_e \sum_{k=0}^{K-1} |x_k(e)|^2\, |\alpha_k|^2 \end{aligned} \tag{14.118}$$

On the other hand, if $\Gamma \ll 1$, (14.112) can be approximated using a power series expansion of the exponential function. Taking the logarithm of (14.112), using the hypothesis of i.i.d. symbols, with $E[a_n] = 0$, and eliminating non-relevant terms, we obtain the following log-likelihood:

$$\ell_{\theta,\varepsilon}(z,e) = E[|a_n|^2] \sum_{k=0}^{K-1} |x_k(e)|^2 + Re\left[E[a_n^2] \sum_{k=0}^{K-1} (x_k^*(e))^2\, e^{j2z} \right] \tag{14.119}$$

Averaging with respect to θ we obtain the following phase independent log-likelihood:

$$\ell_\varepsilon(e) = \sum_{k=0}^{K-1} |x_k(e)|^2 \tag{14.120}$$

which yields the same NDA estimator as (14.117).

For a modulation technique characterized by $E[a_n^2] \neq 0$, (14.119) may be used to obtain an NDA joint estimate of phase and timing. In fact, for a phase estimate given by

$$\hat{\theta} = -\frac{1}{2} \arg \left\{ E[a_n^2] \sum_{k=0}^{K-1} (x_k^*(e))^2 \right\} \tag{14.121}$$

the second term of (14.119) is maximized. Substitution of (14.121) in (14.119) yields a new estimate $\hat{\varepsilon}$ given by

$$\hat{\varepsilon} = \arg \max_e \; E[|a_n|^2] \sum_{k=0}^{K-1} |x_k(e)|^2 + \left| E[a_n^2] \sum_{k=0}^{K-1} x_k^2(e) \right| \tag{14.122}$$

The block diagram of the joint estimator is shown in Figure 14.24, where P values of the time shift ε, $\varepsilon^{(m)}$, $m = 1, \ldots, P$, equally spaced in $[-1/2, 1/2]$ are considered; usually the resolution obtained with $P = 8$ or 10 is sufficient. For each time shift $\varepsilon^{(m)}$, the log-likelihood (14.122) is computed and the value of $\varepsilon^{(m)}$ associated with the largest value of the log-likelihood is selected as the timing estimate. Furthermore, we observe that in the generic branch m, filtering by the matched filter $g_M(iT_c + \varepsilon^{(m)}T)$ and sampling at the instants kT can be implemented by the cascade of an interpolator filter $h_I(iT_c; \mu^{(m)})$ (where $\mu^{(m)}$ depends on $\varepsilon^{(m)}$) and a filter $g_M(iT_c)$, followed by a decimator that provides samples at the instants $\mathtt{m}_k T_c$, as illustrated in Figure 14.19 and described in Section 14.4.

Non-data aided via spectral estimation

Let us consider the log-likelihood (14.120) limited to a symmetric observation time interval $[-LT, LT]$; thus we obtain

$$\ell_\varepsilon(e) = \sum_{k=-L}^{L} |x(kT + eT)|^2 \tag{14.123}$$

Now, as x is a QAM signal, the process $|x(kT + eT)|^2$ is approximately cyclostationary in e of period 1 (see Section 7.2). We introduce the following Fourier series representation

$$|x(kT + eT)|^2 = \sum_{i=-\infty}^{+\infty} c_i^{(k)} \, e^{j2\pi ie} \tag{14.124}$$

where the coefficients $\{c_i^{(k)}\}$ are random variables given by

$$c_i^{(k)} = \int_0^1 |x(kT + eT)|^2 \, e^{-j2\pi ie} \, de \tag{14.125}$$

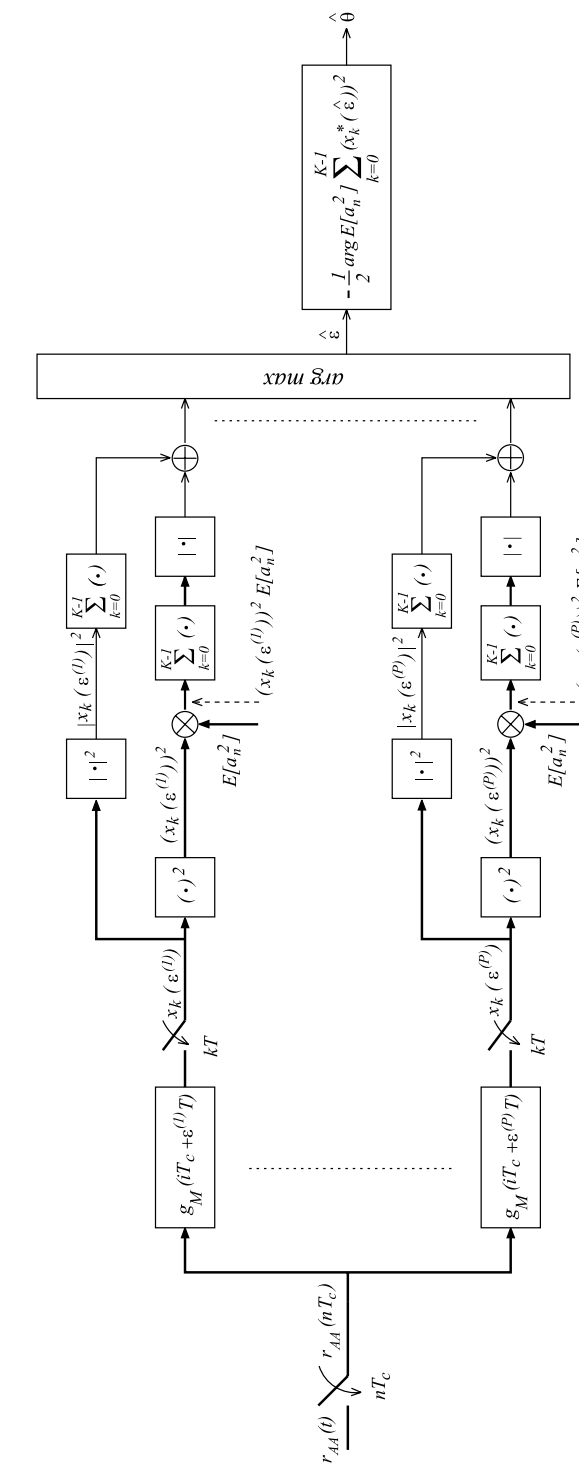


Figure 14.24. NDA joint timing and phase (for $E[a_n^2] \neq 0$) estimator. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

Now (14.123) is equal to the average of the cyclostationary process $|x(kT+eT)|^2$ in the interval $[-L;L]$; defining

$$c_i = \sum_{k=-L}^{L} c_i^{(k)} \tag{14.126}$$

it results [4] that only c_0 and c_1 have non-zero mean, and (14.123) can be written as

$$\ell_\varepsilon(e) = c_0 + 2Re[c_1\, e^{j2\pi e}] + \underbrace{\sum_{|i|\geq 2} 2Re[c_i\, e^{j2\pi ie}]}_{\substack{\text{disturbance with zero mean} \\ \text{for each value of } e}} \tag{14.127}$$

As c_0 and $|c_1|$ are independent of e, the maximum of $\ell_\varepsilon(e)$ yields

$$\hat{\varepsilon} = -\frac{1}{2\pi} \arg c_1 \tag{14.128}$$

However, the coefficient c_1 is obtained by integration, which in general is hard to implement in the digital domain; on the other hand, if the bandwidth of $|x(lT)|^2$ satisfies the relation

$$B_{|x|^2} = \frac{1}{T}(1+\rho) < \frac{1}{2T_c} \tag{14.129}$$

where ρ is the roll-off factor of the matched filter, then c_1 can be computed by DFT. Let $F_0 = T/T_c$, then we obtain

$$c_1 = \sum_{k=-L}^{L} \left[\frac{1}{F_0} \sum_{l=0}^{F_0-1} |x([kF_0+l]\,T_c)|^2\, e^{-j(2\pi/F_0)l} \right] \tag{14.130}$$

A simple implementation of the estimator is possible for $F_0 = 4$; in fact, in this case no multiplications are needed; as $e^{-j(2\pi/4)l} = (-j)^l$, (14.130) simplifies into

$$c_1 = \sum_{k=-L}^{L} \left[\frac{1}{4} \sum_{l=0}^{3} |x([4k+l]\,T_c)|^2\, (-j)^l \right] \tag{14.131}$$

Figure 14.25 illustrates the implementation of the estimator for $F_0 = 4$.

Data-aided and data-directed

If in (14.95) we substitute the parameters α_k and z with their estimates we obtain the phase independent DA (DD) log-likelihood

$$\mathtt{L}_\varepsilon(e) = \exp\left\{ \frac{2}{N_0} Re\left[\sum_{k=0}^{K-1} \hat{a}_k^*\, x_k(e)\, e^{-j\hat{\theta}} \right] \right\} \tag{14.132}$$

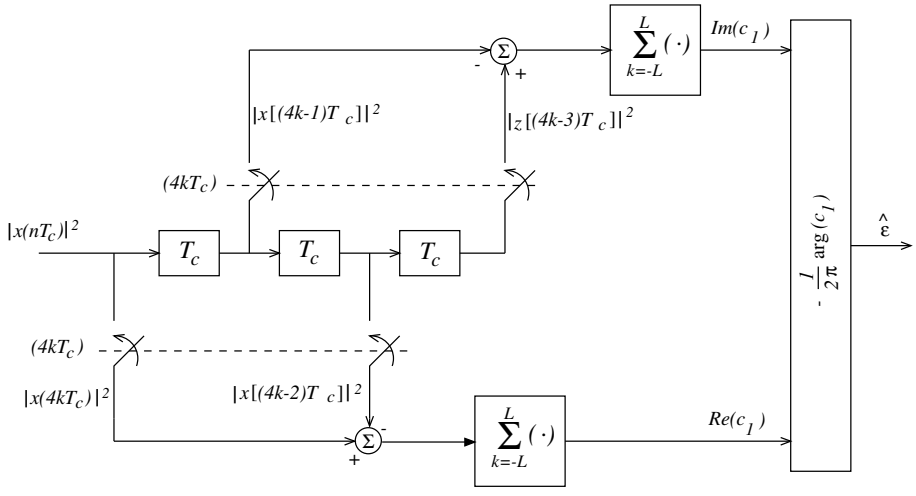


Figure 14.25. NDA timing estimator via spectral estimation for the case $F_0 = 4$. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

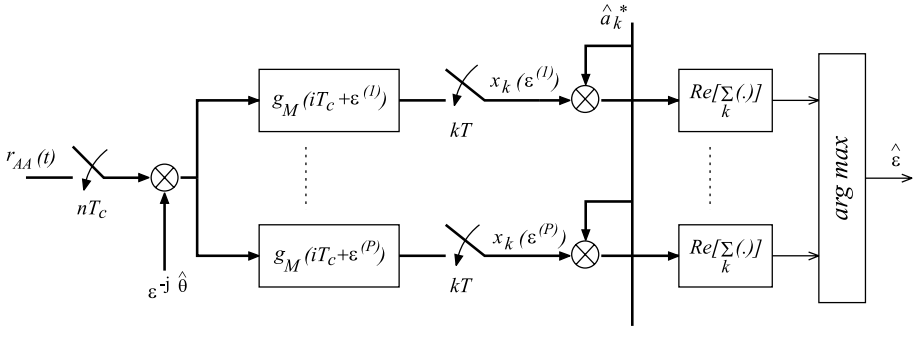


Figure 14.26. Phase independent DA (DD) timing estimator. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

from which we immediately derive the estimate

$$\hat{\varepsilon} = \arg\max_{e} \mathtt{L}_{\varepsilon}(e) \tag{14.133}$$

The block diagram of the estimator is shown in Figure 14.26; note that this algorithm can only be used in the case phase recovery is carried out before timing recovery.

For a joint phase and timing estimator, from (14.95) we get

$$\mathtt{L}_{\theta,\varepsilon}(z, e) = \exp\left\{\frac{2}{N_0} Re\left[\sum_{k=0}^{K-1} \hat{a}_k^* \, x_k(e)\, e^{-jz}\right]\right\} \tag{14.134}$$

Defining

$$\mathtt{r}(e) = \sum_{k=0}^{K-1} \hat{a}_k^* \, x_k(e) \tag{14.135}$$

the estimation algorithm becomes

$$\begin{aligned}(\hat{\theta}, \hat{\varepsilon}) &= \arg\max_{z,e} \; Re[\mathtt{r}(e)\, e^{-jz}] \\ &= \arg\max_{z,e} \; |\mathtt{r}(e)|\, Re[e^{-j(z-\arg(\mathtt{r}(e)))}]\end{aligned} \tag{14.136}$$

The two-variable search of the maximum reduces to a single-variable search; as a matter of fact, once the value of e that maximizes $|\mathtt{r}(e)|$ is obtained, which is independent of z, the second term

$$Re[e^{-j(z-\arg(\mathtt{r}(e)))}] \tag{14.137}$$

is maximized by $z = \arg(\mathtt{r}(e))$. Therefore the joint estimation algorithm is given by

$$\begin{aligned}\hat{\varepsilon} &= \arg\max_{e} \; |\mathtt{r}(e)| = \arg\max_{e} \left| \sum_{k=0}^{K-1} \hat{a}_k^* \, x_k(e) \right| \\ \hat{\theta} &= \arg \mathtt{r}(\hat{\varepsilon}) = \arg \sum_{k=0}^{K-1} \hat{a}_k^* \, x_k(\hat{\varepsilon})\end{aligned} \tag{14.138}$$

Figure 14.27 illustrates the implementation of this second estimator; note that this scheme is a particular case of (7.269).

For both estimators, estimation of the synchronization parameters is carried out every K samples, according to the assumption of slow parameter variations made at the beginning of the section.

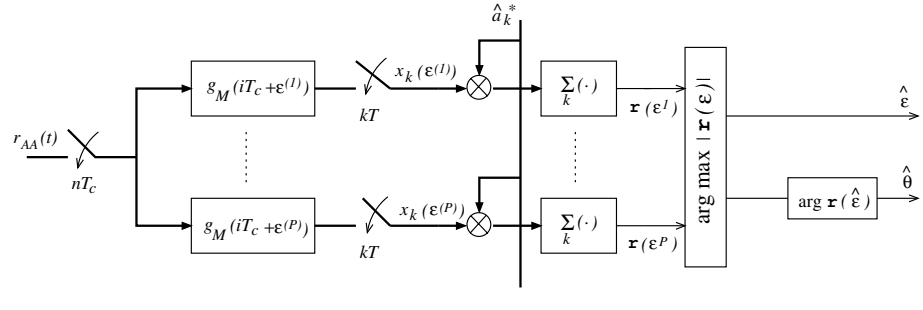


Figure 14.27. DA (DD) joint phase and timing estimator. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

Observation 14.2
In the case the channel is not known, to implement the matched filter g_M we need to estimate the overall impulse response q_C; then the estimation of q_C, for example, by one of the methods presented in Appendix 3.A, and of timing can be performed jointly.

Let $F_0 = T/T_c$ and $Q_0 = T/T_Q$ be integers, with $Q_0 \geq F_0$. From the signal $\{r_{AA}(qT_Q)\}$, obtained by oversampling $r_{AA}(t)$ or by interpolation of $\{r_{AA}(nT_c)\}$, and the knowledge of the training sequence $\{a_k\}$, $k = 0, \ldots, L_{TS} - 1$, the estimate of q_C with sampling period T_Q, or equivalently the estimate of its Q_0/F_0 polyphase components with sampling period T_c (see Observation 8.5), is made. Limiting the estimate to the more significant consecutive samples around the peak, the determination of the timing phase with precision T_Q coincides with the selection of the polyphase component with the largest energy among the Q_0/F_0 polyphase components. This determines the optimum filter g_M with sampling period T_c. Typically for radio systems $F_0 = 2$, and $Q_0 = 4$ or 8.

Data- and phase-directed with feedback: differentiator scheme

Differentiating the log-likelihood (14.95) with respect to e, neglecting non-relevant terms, and evaluating the result at $(\hat{\theta}, e, \hat{\mathbf{a}})$, we obtain

$$\frac{\partial}{\partial e} \ln \{\mathtt{L}_\varepsilon(e)\} \propto Re\left[\sum_{k=0}^{K-1} \hat{a}_k^* \frac{\partial}{\partial e} x(kT + eT)\, e^{-j\hat{\theta}}\right] \tag{14.139}$$

With reference to the scheme of Figure 14.22, if we suppose that the sum in (14.139) is approximated by the filtering operation by the loop filter $F(z)$, the error signal $e(kT)$ results

$$e(kT) = Re\left[\hat{a}_k^* \frac{\partial}{\partial e} x(kT + eT)|_{e=\hat{\varepsilon}_k}\, e^{-j\hat{\theta}}\right] \tag{14.140}$$

The partial derivative of $x(kT + eT)$ with respect to e can be carried out in the digital domain by a differentiator filter with an ideal frequency response given by

$$\mathcal{H}_d(f) = j2\pi f \qquad |f| \leq \frac{1}{2T_c} \tag{14.141}$$

In practice, if $T/T_c \geq 2$ it is simpler to implement a differentiator filter by a finite difference filter having a symmetric impulse response given by

$$h_d(iT_c) = \frac{1}{2T_c}(\delta_{i+1} - \delta_{i-1}) \tag{14.142}$$

Figure 14.28 illustrates the block diagram of the estimator, where the compact notation $\dot{x}(t)$ is used in place of $(dx(t)/dt)$; moreover, based on the analysis of Section 14.5.2, if $u(kT)$ is the loop filter output, the estimate of ε is given by

$$\hat{\varepsilon}_{k+1} = \hat{\varepsilon}_k + \mu_\varepsilon u(kT) \tag{14.143}$$

where μ_ε is a suitable constant. Applying (14.88) to the value of $\hat{\varepsilon}_{k+1}$, we obtain the values of μ_{k+1} and $\mathtt{m}_{k+1}$.

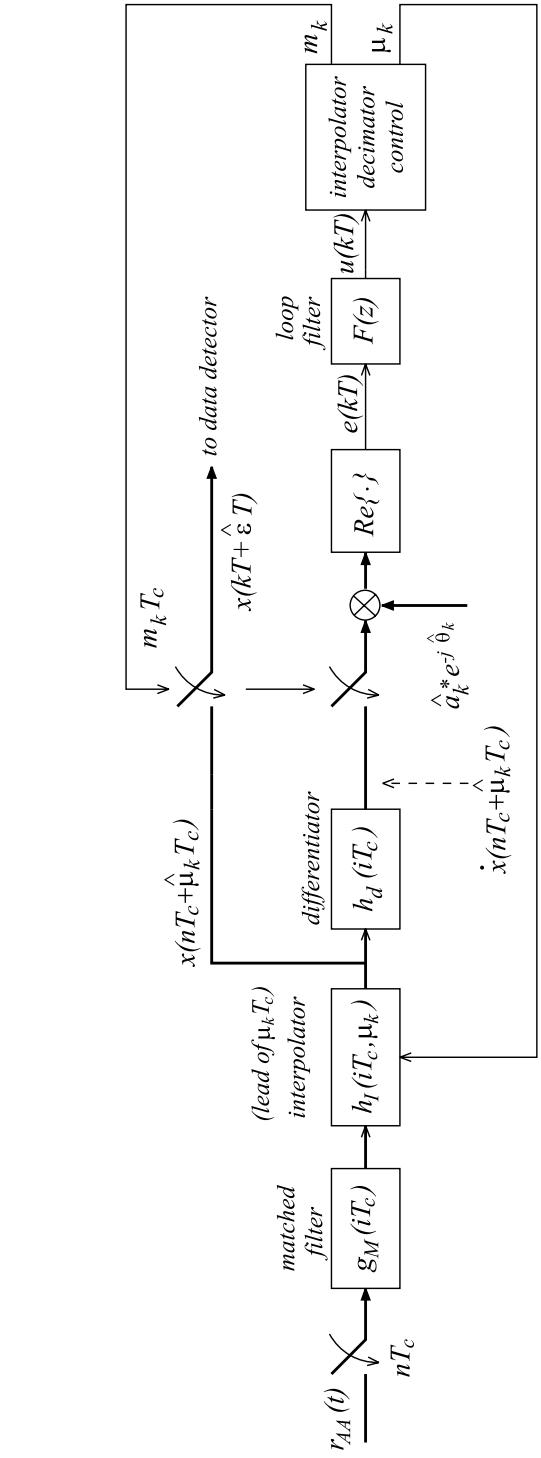


Figure 14.28. DD & Dθ-FB timing estimator. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

Data- and phase-directed with feedback: Mueller & Muller scheme

The present algorithm gets its name from Mueller and Muller, who first proposed it in 1976 [6]. Consider the estimation error $e_\varepsilon = \hat{\varepsilon} - \varepsilon$ and the pulse $q_R(t) = q_C * g_M(t)$; the basic idea consists in generating an *error signal* whose mean value assumes one of the following two expressions:

$$\text{Type A:} \qquad E[e(kT)] = Re\left\{\tfrac{1}{2}[q_R(e_\varepsilon T + T) - q_R(e_\varepsilon T - T)]\right\} \tag{14.144}$$

$$\text{Type B:} \qquad E[e(kT)] = Re\{q_R(e_\varepsilon T + T)\} \tag{14.145}$$

Observe that, under the assumptions of Section 14.4, $q_R(t)$ is a Nyquist pulse; moreover, we assume that in the absence of channel distortion, $q_R(t)$ is an even function.

Note that the signal (14.144) is an odd function of the estimation error e_ε for $e_\varepsilon \in (-\infty, \infty)$, whereas the signal (14.145) is an odd function of e_ε only around $e_\varepsilon = 0$. Under lock conditions, i.e. for $e_\varepsilon \to 0$, the two versions of the algorithm exhibit a similar behavior. However, the type A algorithm gives better results than the type B algorithm in transient conditions because the mean value of the error signal for the type B algorithm is not symmetric. Moreover, the type A algorithm results effective also in the presence of signal distortion.

The error signal for the type A algorithm is chosen equal to

$$e(kT) = \kappa\, Re[[\hat{a}^*_{k-1}\, x_k(\hat{\varepsilon}) - \hat{a}^*_k\, x_{k-1}(\hat{\varepsilon})]\, e^{-j\hat{\theta}}] \tag{14.146}$$

where κ is a suitable constant whose value is discussed below. Assuming that $\hat{a}_{k-1} = a_{k-1}$, $\hat{a}_k = a_k$, and $\hat{\theta} = \theta$, from (14.71) and (14.79) for $o = \Omega = 0$ (14.146) can be written as

$$\begin{aligned} e(kT) = \kappa\, Re\Bigg\{\Bigg[a^*_{k-1} \sum_{i=-\infty}^{+\infty} a_i\, q_R(kT + e_\varepsilon T - iT) \\ - a^*_k \sum_{i=-\infty}^{+\infty} a_i\, q_R((k-1)T + e_\varepsilon T - iT)\Bigg] \\ + a^*_{k-1}\, \tilde{w}_k - a^*_k\, \tilde{w}_{k-1}\Bigg\} \end{aligned} \tag{14.147}$$

where $\tilde{w}_k$ is the decimated noise signal at the matched filter output. We define

$$q_m(e_\varepsilon) = q_R(mT + e_\varepsilon T) \tag{14.148}$$

then with a suitable change of variables (14.147) becomes

$$e(kT) = \kappa Re\left\{\left[a^*_{k-1} \sum_{m=-\infty}^{+\infty} a_{k-m} q_m(e_\varepsilon) - a^*_k \sum_{m=-\infty}^{+\infty} a_{k-1-m} q_m(e_\varepsilon)\right] + a^*_k \tilde{w}_k - a^*_k \tilde{w}_{k-1}\right\} \tag{14.149}$$

Taking the mean value of $e(kT)$ we obtain

$$\begin{aligned} E[e(kT)] &= \kappa\, Re\{(E[|a_k|^2] - |\mathrm{m}_a|^2)[q_1(e_\varepsilon) - q_{-1}(e_\varepsilon)]\} \\ &= \kappa\, Re\{(E[|a_k|^2] - |\mathrm{m}_a|^2)[q_R(e_\varepsilon T + T) - q_R(e_\varepsilon T - T)]\} \end{aligned} \tag{14.150}$$

For

$$\kappa = \frac{1}{2(E[|a_k|^2] - |\mathrm{m}_a|^2)} \tag{14.151}$$

we obtain (14.144). Similarly, in the case of the type B algorithm the error signal assumes the expression

$$e(kT) = \frac{1}{(E[|a_k|^2] - |\mathrm{m}_a|^2)}\, [\hat{a}_{k-1} - \mathrm{m}_a]^*\, x_k(\hat{\varepsilon})\, e^{-j\hat{\theta}} \tag{14.152}$$

Figure 14.29 illustrates the block diagram of the direct section of the type A estimator. The constant κ is included in the loop filter and is not explicitly shown.

Non-data aided with feedback

We consider the log-likelihood (14.120), obtained for a NDA estimator, and differentiate it with respect to e to get

$$\begin{aligned} \frac{\partial \ell_\varepsilon(e)}{\partial e} &= \frac{\partial}{\partial e} \sum_{k=0}^{K-1} |x(kT + eT)|^2 \\ &= \sum_{k=0}^{K-1} 2Re[x(kT + eT)\, \dot{x}^*(kT + eT)] \end{aligned} \tag{14.153}$$

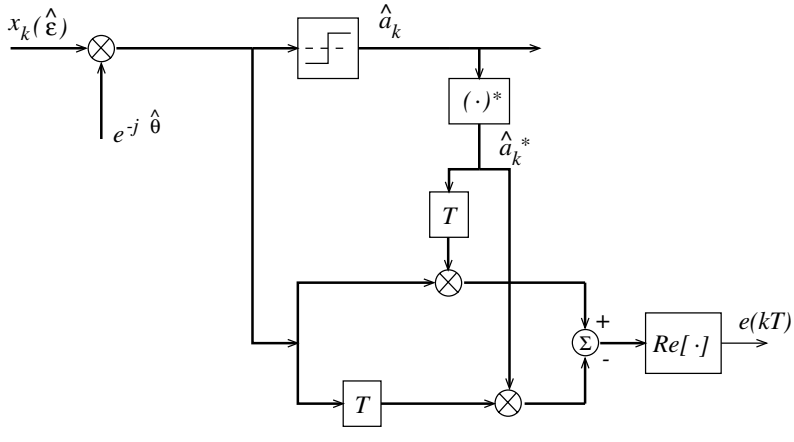


Figure 14.29. Mueller & Muller type A timing estimator.

If we assume that the sum is carried out by the loop filter, the error signal is given by

$$e(kT) = Re[x(kT + \hat{\varepsilon}_k T)\, \dot{x}^*(kT + \hat{\varepsilon}_k T)] \tag{14.154}$$

14.5.4 Phasor estimators

Data- and timing-directed

We discuss an algorithm that directly yields the phasor $\exp(j\hat{\theta})$ in place of the phase $\hat{\theta}$. Assuming that an estimate of $\hat{\mathbf{a}}$ and $\hat{\varepsilon}$ is available, the likelihood (14.95) becomes

$$\mathtt{L}_\theta(z) = \exp\left\{\frac{2}{N_0} Re\left[\sum_{k=0}^{K-1} \hat{a}_k^*\, x_k(\hat{\varepsilon})\, e^{-jz}\right]\right\} \tag{14.155}$$

and is maximized by

$$e^{j\hat{\theta}} = e^{j \arg \sum_{k=0}^{K-1} \hat{a}_k^*\, x_k(\hat{\varepsilon})} \tag{14.156}$$

Figure 14.30 illustrates the implementation of the estimator (14.156).

Non-data aided for M-PSK signals

In an M-PSK system, to remove the data dependence from the decimator output signal in the scheme of Figure 14.19, we raise the samples $x_k(\hat{\varepsilon})$ to the M-th power. Assuming absence of ISI, we get

$$x_k^M(\hat{\varepsilon}) = [a_k e^{j\theta} + \tilde{w}_k]^M = a_k^M e^{jM\theta} + w_{M,k} \tag{14.157}$$

where $\tilde{w}_k$ represents the decimator output noise, and $w_{M,k}$ denotes the overall disturbance.

As $a_k^M = (e^{j2\pi l/M})^M = 1$, (14.157) becomes

$$x_k^M(\hat{\varepsilon}) = e^{jM\theta} + w_{M,k} \tag{14.158}$$

From (14.95), we substitute $(x_k(\hat{\varepsilon}))^M$ for $\alpha_k^* x_k(\hat{\varepsilon})$ obtaining the likelihood

$$\mathtt{L}_\theta(z) = \exp\left\{\frac{2}{N_0} Re\left[\sum_{k=0}^{K-1} (x_k(\hat{\varepsilon}))^M e^{-jzM}\right]\right\} \tag{14.159}$$

which is maximized by the phasor

$$\exp(j\hat{\theta}M) = \exp\left[j \arg \sum_{k=0}^{K-1} (x_k(\hat{\varepsilon}))^M\right] \tag{14.160}$$

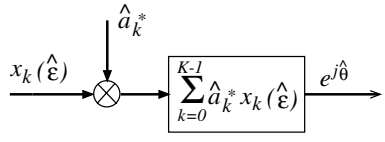


Figure 14.30. DD & Dε estimator of the phasor $e^{j\theta}$.

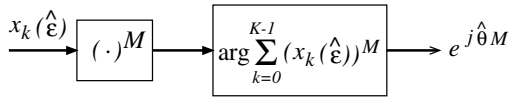


Figure 14.31. NDA estimator of the phasor $e^{j\theta}$ for M-PSK.

We note that raising $x_k(\hat{\varepsilon})$ to the M-th power causes a phase ambiguity equal to a multiple of $(2\pi)/M$; in fact, if $\hat{\theta}$ is a solution to (14.160), also $(\hat{\theta} + 2\pi l/M)$ for $l = 0, \ldots, M-1$, are solutions. This ambiguity can be removed, for example, by differential encoding (see Section 6.5.2). The estimator block diagram is illustrated in Figure 14.31.

Data- and timing-directed with feedback

Consider the likelihood (14.155) obtained for the DD & Dε estimator of the phasor $e^{j\theta}$; taking the logarithm, differentiating it with respect to z, and ignoring non-relevant terms, we obtain the error signal

$$e(kT) = Im[\hat{a}_k^* \, x_k(\hat{\varepsilon}) \, e^{-j\hat{\theta}_k}] \tag{14.161}$$

Observe that, in the absence of noise, $x_k(\hat{\varepsilon}) = a_k \, e^{j\theta}$, and (14.161) becomes

$$e(kT) = |\hat{a}_k|^2 \sin(\theta - \hat{\theta}_k) \tag{14.162}$$

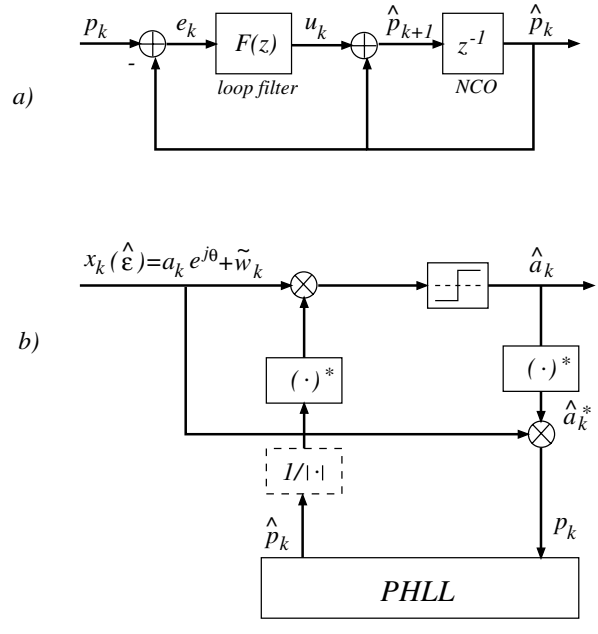


Figure 14.32. (a) PHLL; (b) DD & Dε-FB phasor estimator. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

Hence, we can use a digital version of the PLL to implement the estimator. However, the error signal (14.161) introduces a phase ambiguity; in fact it assumes the same value if we substitute $(\hat{\theta}_k - \pi)$ for $\hat{\theta}_k$. An alternative to the digital PLL is given by the *phasor-locked loop* (PHLL), that provides an estimate of the phasor $e^{j\theta}$, rather than the estimate of θ, thus eliminating the ambiguity.

The block diagram of the PHLL is illustrated in Figure 14.32a; it is a feedback structure with the phasor $p_k = e^{j\theta_k}$ as input and the estimate $\hat{p}_k = e^{j\hat{\theta}_k}$ as output. The error signal e_k is obtained by subtracting the estimate $\hat{p}_k$ from p_k; then e_k is input to the loop filter $F(z)$ that yields the signal u_k, which is used to update the phasor estimate according to the recursive relation

$$\hat{p}_{k+1} = \hat{p}_k + (f * u)(k) \tag{14.163}$$

Figure 14.32b illustrates the block diagram of a DD & Dε phasor estimator that implements the PHLL. Observe that the input phasor p_k is obtained by multiplying $x_k(\hat{\varepsilon})$ by $\hat{a}_k^*$ to remove the dependence on the data; the dashed block normalizes the estimate $\hat{p}_k$ in the QAM case.

14.6 Algorithms for carrier frequency recovery

As mentioned in Section 14.4, phase and timing estimation algorithms work correctly only if the frequency offset is small. Therefore the frequency offset must be compensated before the estimate of the other two synchronization parameters takes place. Hence the algorithms that we will present are mainly NDA and *non-clock-aided* (NCA); timing-directed algorithms are possible only in the case the frequency offset has a magnitude much smaller than $1/T$.

In Figure 14.33 we redraw part of the digital receiver scheme of Figure 14.19; observe that the position of the matched filter is interchanged with that of the interpolator filter. In this scheme, the samples $r_{AA}(nT_c)$ are multiplied by $\exp(-j\hat{\Omega}nT_c)$ to remove the frequency offset.

In [4] it is shown that, whenever Ω satisfies the condition

$$\left| \frac{\Omega T}{2\pi} \right| \leq 0.15 \tag{14.164}$$

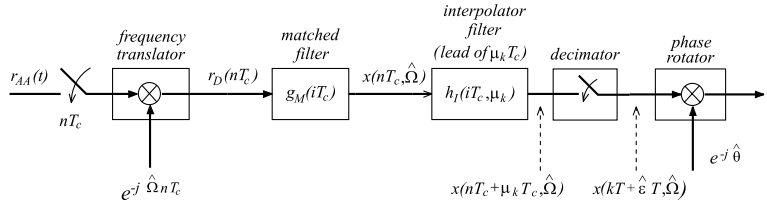


Figure 14.33. Receiver of Figure 14.19 with interpolator and matched filter interchanged. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

the following approximation holds:

$$x(kT + eT, o) \simeq e^{-jokT}\, x_k(e) \tag{14.165}$$

Then the likelihood (14.80) can be written as

$$\mathrm{L}_{\theta,\varepsilon,\Omega,\mathbf{a}}(z, e, o, \boldsymbol{\alpha}) = \exp\left\{\frac{2}{N_0}\sum_{k=0}^{K-1} Re[\alpha_k^*\, x_k(e)\, e^{-jokT}\, e^{-jz}]\right\} \tag{14.166}$$

Therefore in the schemes of Figure 14.19 and Figure 14.33 the frequency translator may be moved after the decimator, together with the phase rotator.

14.6.1 Frequency offset estimators

Non-data aided

Suppose the receiver operates with a low signal-to-noise ratio Γ; similarly to (14.123) the log-likelihood for the joint estimate of (ε, Ω) in the observation interval $[-LT, LT]$ is given by

$$\ell_{\varepsilon,\Omega}(e, o) = \sum_{k=-L}^{L} |x(kT + eT, o)|^2 \tag{14.167}$$

By expanding $\ell_{\varepsilon,\Omega}(e, o)$ in Fourier series and using the notation introduced in the previous section we obtain

$$\ell_{\varepsilon,\Omega}(e, o) = c_0 + 2Re[c_1\, e^{j2\pi e}] + \underbrace{\sum_{|i|\geq 2} 2Re[c_i\, e^{j2\pi ie}]}_{\text{disturbance}} \tag{14.168}$$

Now the mean value of c_0, $E[c_0]$, depends on o, but is independent of e and furthermore is maximized for $o = \hat{\Omega}$; hence

$$\hat{\Omega} = \arg\max_o c_0 \tag{14.169}$$

As we did for the derivation of (14.131), starting with (14.169) and assuming the ratio $F_0 = T/T_c$ is an integer, we obtain the following joint estimate of (Ω, ε) [4]:

$$\begin{aligned} \hat{\Omega} &= \arg\max_o \sum_{n=-LF_0}^{LF_0-1} |x(nT_c, o)|^2 \\ \hat{\varepsilon} &= \arg \sum_{n=-LF_0}^{LF_0-1} |x(nT_c, o)|^2\, e^{-j2\pi n/F_0} \end{aligned} \tag{14.170}$$

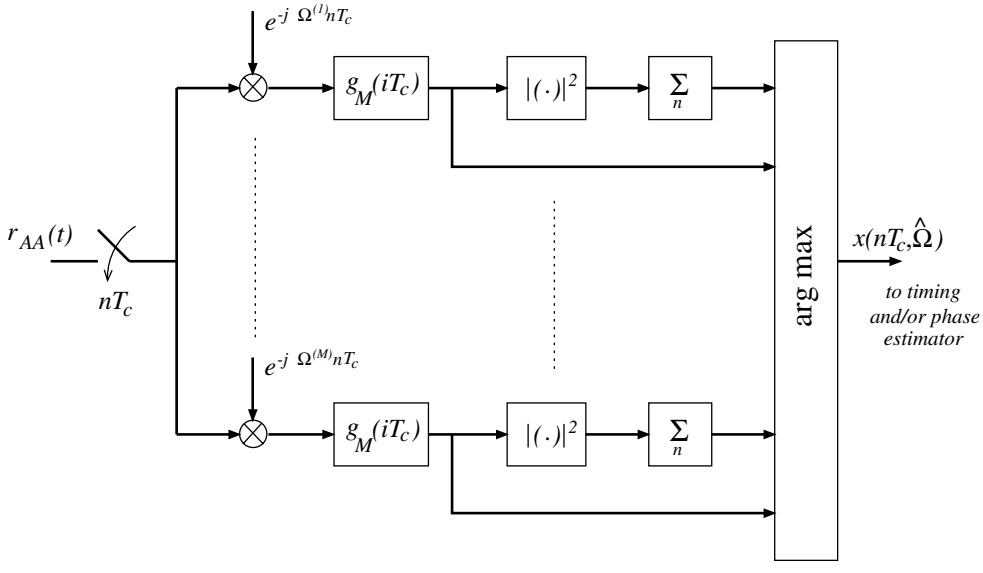


Figure 14.34. NDA frequency offset estimator. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

The implementation of the estimator is illustrated in Figure 14.34; observe that the signal $x(nT_c, o)$ can be rewritten as

$$\begin{aligned} x(nT_c, o) &= \sum_i r_{AA}(iT_c)\, e^{-joiT_c}\, g_M(nT_c - iT_c) \\ &= e^{-jonT_c} \sum_i r_{AA}(iT_c)\, e^{-jo(i-n)T_c}\, g_M(nT_c - iT_c) \\ &= e^{-jonT_c} \sum_i r_{AA}(iT_c)\, g_M^{(pb)}(nT_c - iT_c; o) \end{aligned} \tag{14.171}$$

where the expression of the filter

$$g_M^{(pb)}(iT_c; o) = g_M(iT_c)\, e^{joiT_c} \tag{14.172}$$

depends on the offset o. Defining

$$x_o(nT_c) = \sum_i r_{AA}(iT_c)\, g_M^{(pb)}(nT_c - iT_c; o) \tag{14.173}$$

we note that $|x(nT_c, o)| = |x_o(nT_c)|$, and hence in the m-th branch of Figure 14.34 the cascade of the frequency translator and the filter can be substituted with a simple filter with impulse response $g_M^{(pb)}(iT_c; \Omega^{(m)})$.

Non-data aided and timing-independent with feedback

Differentiating the log-likelihood defined by (14.170), equal to $c_0(\Omega)$, with respect to Ω we obtain the error signal

$$e(nT_c) = 2Re\left[x(nT_c, o)\,\frac{\partial}{\partial o}\,x^*(nT_c, o)\right]\Bigg|_{o=\hat{\Omega}_n} \tag{14.174}$$

Observe that, as $c_0(\Omega)$ is independent of e, then also $e(nT_c)$ is independent of e. From the first of (14.171), the partial derivative of $x(nT_c, o)$ with respect to o is given by

$$\frac{\partial}{\partial o}\,x(nT_c, o) = \sum_{i=-\infty}^{+\infty} (-jiT_c)\,r_{AA}(iT_c)\,e^{-joiT_c}\,g_M(nT_c - iT_c) \tag{14.175}$$

We define the *frequency matched filter* as

$$g_{FM}(iT_c) = (jiT_c)\cdot g_M(iT_c) \tag{14.176}$$

Observe now that, if the signal $r_D(nT_c) = r_{AA}(nT_c)\,e^{-jonT_c}$ is input to the filter $g_{FM}(iT_c)$, then from (14.175) the output is given by

$$x_{FM}(nT_c) = g_{FM} * r_D(nT_c) = \frac{\partial}{\partial o}\,x(nT_c, o) + jnT_c\,x(nT_c, o) \tag{14.177}$$

from which we obtain

$$\frac{\partial}{\partial o}\,x(nT_c, o) = x_{FM}(nT_c) - jnT_c\,x(nT_c, o) \tag{14.178}$$

Therefore the expression of the error signal (14.174) becomes

$$e(nT_c) = 2Re[x(nT_c, \hat{\Omega}_n)\,x^*_{FM}(nT_c, \hat{\Omega}_n)] \tag{14.179}$$

The block diagram of the resultant estimator is shown in Figure 14.35. The loop filter output $u(kT_c)$ is sent to the NCO that yields the frequency offset estimate according to the recursive equation

$$\hat{\Omega}_{n+1}\,T_c = \hat{\Omega}_n\,T_c + \mu_\Omega\,u(nT_c) \tag{14.180}$$

where μ_Ω is the NCO gain.

Non-data aided and timing-directed with feedback

Consider the log-likelihood (14.167); to get the Dε estimator, we substitute e with the estimate $\hat{\varepsilon}$ obtaining the log-likelihood

$$\ell_\Omega(o) = \sum_{k=0}^{K-1} |x(kT + \hat{\varepsilon}T, o)|^2 \tag{14.181}$$

Proceeding as in the previous section, we get the block diagram illustrated in Figure 14.36.

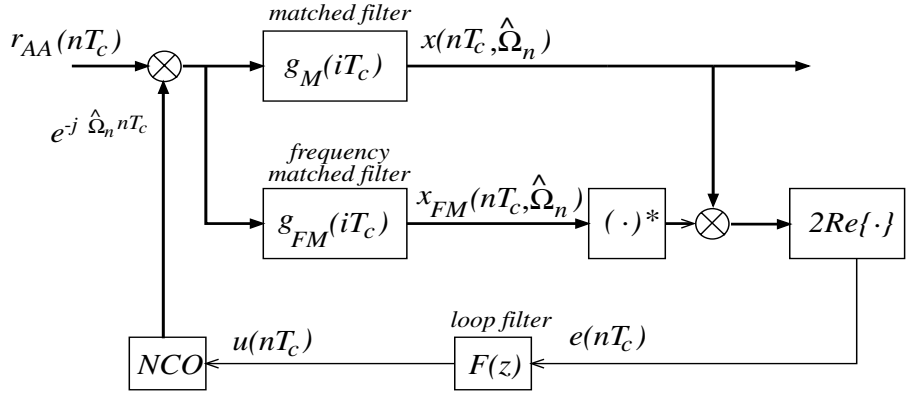


Figure 14.35. NDA-NDε-FB frequency offset estimator. [*From Meyr, Moeneclaey, and Fechtel (1998). Reproduced by permission of Wiley.*]

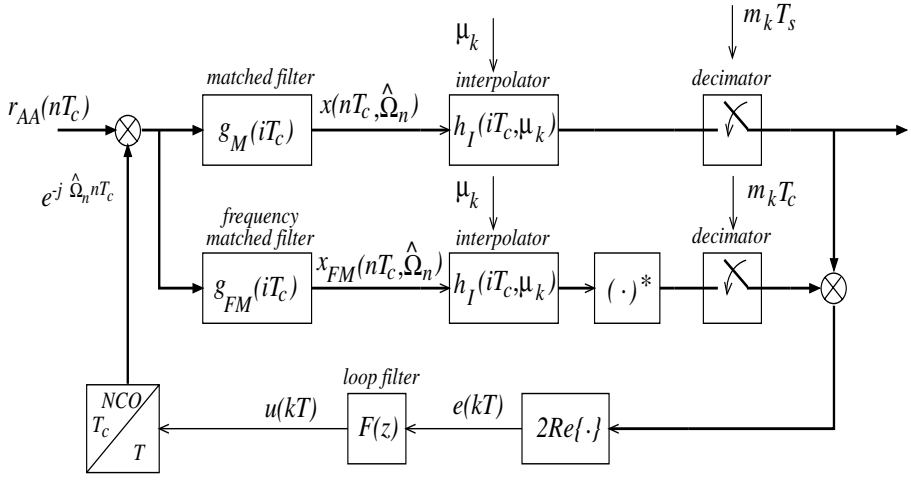


Figure 14.36. NDA-Dε-FB frequency offset estimator.

14.6.2 Estimators operating at the modulation rate

As mentioned at the beginning of Section 14.6, whenever the condition (14.164) is satisfied, the frequency translator can be moved after the decimator and consequently frequency offset estimation may take place after timing estimation, thus obtaining Dε algorithms. The likelihood (14.166) for an observation interval $[-(K-1)/2, (K-1)/2]$, with K odd, evaluated at $e = \hat{\varepsilon}$ becomes

$$\Lambda_{\theta,\Omega,\mathbf{a}}(z, o, \boldsymbol{\alpha}) = \exp\left\{\frac{2}{N_0}\sum_{k=-(K-1)/2}^{(K-1)/2} Re[\alpha_k^* x_k(\hat{\varepsilon})\, e^{-jokT}\, e^{-jz}]\right\} \tag{14.182}$$

Data-aided and data-directed

We assume a known training sequence $\mathbf{a} = \boldsymbol{\alpha}_0$ is transmitted during the acquisition phase, and denote as $\alpha_{0,k}$ the k-th symbol of the sequence. The log-likelihood yields the joint estimate of (Ω, θ) as

$$(\hat{\Omega}, \hat{\theta}) = \arg\max_{z,o} \sum_{k=-(K-1)/2}^{(K-1)/2} Re[\alpha_{0,k}^* \, x_k(\hat{\varepsilon}) \, e^{-jokT} \, e^{-jz}] \tag{14.183}$$

Note that the joint estimate is computed by finding the maximum of a function of one variable; in fact, defining

$$\mathtt{r}(o) = \sum_{k=-(K-1)/2}^{(K-1)/2} \alpha_{0,k}^* \, x_k(\hat{\varepsilon}) \, e^{-jokT} \tag{14.184}$$

(14.183) can be rewritten as

$$(\hat{\Omega}, \hat{\theta}) = \arg\max_{z,o} \, |\mathtt{r}(o)| \, Re[e^{-j(z - \arg\{\mathtt{r}(o)\})}] \tag{14.185}$$

The maximum (14.183) is obtained by first finding the value of o that maximizes $|\mathtt{r}(o)|$,

$$\hat{\Omega} = \arg\max_{o} \, |\mathtt{r}(o)| \tag{14.186}$$

and then finding the value of z for which the term within brackets in (14.185) becomes real valued,

$$\hat{\theta} = \arg\{\mathtt{r}(\hat{\Omega})\} \tag{14.187}$$

We now want to solve (14.186) in close form; a necessary condition to get a maximum is that the derivative of $|\mathtt{r}(o)|^2 = \mathtt{r}(o)\,\mathtt{r}^*(o)$ with respect to o is equal to zero for $o = \hat{\Omega}$. Defining

$$b_k = \left[\frac{K^2 - 1}{4} - k(k+1) \right] \tag{14.188}$$

we obtain

$$\hat{\Omega} T = \arg \left\{ \sum_{k=-(K-1)/2}^{(K-1)/2} b_k \, \frac{\alpha_{0,k+1}^*}{\alpha_{0,k}^*} \, [x_{k+1}(\hat{\varepsilon}) \, x_k^*(\hat{\varepsilon})] \right\} \tag{14.189}$$

The DD estimator is obtained by substituting $\boldsymbol{\alpha}_0$ with the estimate $\hat{\mathbf{a}}$.

Non-data aided for M-PSK

By raising $(\alpha_{0,k+1}^*/\alpha_{0,k}^*)(x_{k+1}(\hat{\varepsilon})\, x_k^*(\hat{\varepsilon}))$ to the M-th power we obtain the NDA version of the DA (DD) algorithm for M-PSK signals as

$$\hat{\Omega} T = \arg \left\{ \sum_{k=-(K-1)/2}^{(K-1)/2} b_k \, [x_{k+1}(\hat{\varepsilon}) \, x_k^*(\hat{\varepsilon})]^M \right\} \tag{14.190}$$

14.7 Second-order digital PLL

In feedback systems, the recovery of the phase θ and the frequency Ω can be jointly performed by a second-order *digital* PLL (DPLL), given by

$$\hat{\theta}_{k+1} = \hat{\theta}_k + \mu_\theta \, e_\theta(kT) + \hat{\Omega}_k \tag{14.191}$$

$$\hat{\Omega}_{k+1} = \hat{\Omega}_k + \mu_{\Omega,1} \, e_\theta(kT) + \mu_{\Omega,2} \, e_\Omega(kT) \tag{14.192}$$

where e_θ and e_Ω are estimates of the phase error and of the frequency error, respectively, and μ_θ, $\mu_{\Omega,1}$ and $\mu_{\Omega,2}$ are suitable constants. Typically (see Example 15.6.4 on page 1110) $\mu_{\Omega,1} \simeq \mu_{\Omega,2} \simeq \sqrt{\mu_\theta}$.

Observe that (14.191) and (14.192) form a digital version of the second-order analog PLL illustrated in Figure 14.7.

14.8 Synchronization in spread spectrum systems

In this section we discuss early-late FB schemes, named *delay-locked loops* (DLLs) [7], for the timing recovery in spread spectrum systems (see Chapter 10).

14.8.1 The transmission system

Transmitter

In Figure 14.37 (see Figures 10.2 and 10.3) the (baseband equivalent) scheme of a spread spectrum transmitter is illustrated. The symbols $\{a_k\}$, generated at the modulation rate $1/T$, are input to a holder, which outputs the symbols $\bar{a}_m$ at the *chip* rate $1/T_{chip}$. The *chip* period T_{chip} is given by

$$T_{chip} = \frac{T}{N_{SF}} \tag{14.193}$$

where N_{SF} denotes the spreading factor. Then the symbols $\{a_k\}$ are multiplied by the spreading code $\{c_m\}$ and input to a filter with impulse response $h_{Tx}(t)$ that includes the DAC. The (baseband equivalent) transmitted signal $s(t)$ is expressed as

$$s(t) = \sum_{m=-\infty}^{+\infty} \bar{a}_m \, c_m \, h_{Tx}(t - mT_{chip}) \tag{14.194}$$

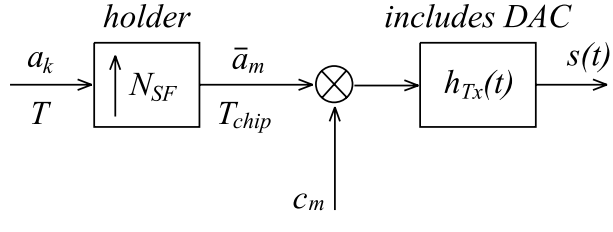


Figure 14.37. Baseband transmitter for spread spectrum systems.

In terms of the symbols $\{a_k\}$, we obtain the following alternative representation that will be used next:

$$s(t) = \sum_{k=-\infty}^{+\infty} a_k \sum_{m=kN_{SF}}^{kN_{SF}+N_{SF}-1} c_m \, h_{Tx}(t - mT_{chip}) \tag{14.195}$$

Optimum receiver

With the same assumptions of Section 14.1 and for the transmission of K symbols, the received signal $r_C(t)$ is expressed as

$$r_C(t) = \sum_{k=0}^{K-1} a_k \sum_{m=kN_{SF}}^{kN_{SF}+N_{SF}-1} c_m \, q_C(t - mT_{chip} - \varepsilon T_{chip}) \, e^{j(\Omega t+\theta)} + w_{C_\varphi}(t) \tag{14.196}$$

The likelihood $\mathtt{L}_{ss} = \mathtt{L}_{\theta,\varepsilon,\Omega,\mathbf{a}}(z, e, o, \boldsymbol{\alpha})$ can be computed as in (14.77); after a few steps we obtain

$$\begin{aligned} \mathtt{L}_{ss} = \exp\left\{ \frac{2}{N_0} \sum_{k=0}^{K-1} Re\left[\alpha_k^* \, e^{-jz} \sum_{m=kN_{SF}}^{kN_{SF}+N_{SF}-1} c_m^* \right.\right. \\ \left.\left. \int_{T_K} \rho(t) \, e^{-jot} \, g_M(mT_{chip} + eT_{chip} - t) \, dt \right] \right\} \end{aligned} \tag{14.197}$$

Defining the two signals

$$\begin{aligned} x(t, o) &= \int_{T_K} \rho(\tau) \, e^{-jo\tau} \, g_M(t - \tau) \, d\tau \\ y(kT, e, o) &= \sum_{m=kN_{SF}}^{kN_{SF}+N_{SF}-1} c_m^* \, x(mT_{chip} + eT_{chip}, o) \end{aligned} \tag{14.198}$$

the likelihood becomes

$$\mathtt{L}_{\theta,\varepsilon,\Omega,\mathbf{a}}(z, e, o, \boldsymbol{\alpha}) = \exp\left\{ \frac{2}{N_0} \sum_{k=0}^{K-1} Re[\alpha_k^* \, y(kT, e, o) \, e^{-jz}] \right\} \tag{14.199}$$

To obtain the samples $y(kT, e, o)$ we can proceed as follows:

1. obtain the samples

$$y(lT_{chip}, e, o) = \sum_{m=l-N_{SF}+1}^{l} c_m^* \, x(mT_{chip} + eT_{chip}, o) \tag{14.200}$$

2. decimate $y(lT_{chip}, e, o)$ at $l = (k + 1)N_{SF} - 1$, i.e. evaluate $y(lT_{chip}, e, o)$ for $l = N_{SF} - 1, 2N_{SF} - 1, \ldots, K \cdot N_{SF} - 1$.

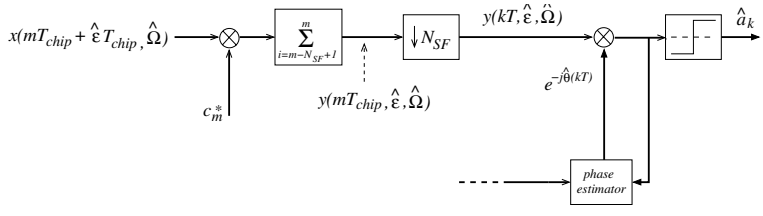


Figure 14.38. Digital receiver for spread spectrum systems.

By (14.199) it is possible to derive the optimum digital receiver. In particular, up to the decimator that outputs samples at instants $\{\mathtt{m}_m T_c\}$ the receiver is identical to that of Figure 14.19;[8] then in Figure 14.38 only part of the receiver is shown.

14.8.2 Timing estimators with feedback

Assume there is no frequency offset, i.e. $\Omega = 0$; the likelihood is obtained by letting $o = 0$ in (14.199) to yield

$$\mathtt{L}_{\theta,\varepsilon,\mathbf{a}}(z, e, \boldsymbol{\alpha}) = \exp\left\{\frac{2}{N_0}\sum_{k=0}^{K-1} Re[\alpha_k^*\, y_k(e)\, e^{-jz}]\right\} \tag{14.201}$$

where we use the compact notation

$$y_k(e) = y(kT, e, 0) \tag{14.202}$$

The early-late FB estimators are obtained by approximating the derivative of (14.201) with respect to e with a finite difference, and evaluating it for $e = \hat{\varepsilon}_k$.

Non-data aided: non-coherent DLL

In the NDA case, the log-likelihood is obtained from (14.120); using $y_k(e)$ instead of $x_k(e)$ yields

$$\ell_\varepsilon(e) = \sum_{k=0}^{K-1} |y_k(e)|^2 \tag{14.203}$$

From (14.110) the derivative of $\ell_\varepsilon(e)$ is approximated as

$$\frac{\partial \ell_\varepsilon(e)}{\partial e} \simeq \frac{1}{2\delta}\sum_{k=0}^{K-1}[|y_k(e+\delta)|^2 - |y_k(e-\delta)|^2] \tag{14.204}$$

By including the constant $1/(2\delta)$ in the loop filter, and also assuming that the sum is performed by the loop filter we obtain the error signal

$$e(kT) = |y_k(\hat{\varepsilon}_k + \delta)|^2 - |y_k(\hat{\varepsilon}_k - \delta)|^2 \tag{14.205}$$

---

[8] Note that now $T_c \simeq T_{chip}/2$, and the sampling instants at the decimator are such that $mT_{chip} + \hat{\varepsilon}\, T_{chip} = \mathtt{m}_m T_c + \mu_m T_c$. The estimate $\hat{\varepsilon}$ is updated at every symbol period $T = T_{chip} \cdot N_{SF}$.

The block diagram of the estimator is shown in Figure 14.39; note that the lag and the lead equal to δT_{chip} are implemented by interpolator filters operating at the sampling period T_c (see (14.91)) with parameter μ equal, respectively, to $-\tilde{\delta}$ and $+\tilde{\delta}$, where $\tilde{\delta} = \delta(T_{chip}/T_c)$. The estimator is called a non-coherent digital DLL [8, 9], as the dependence of the error signal on the pair $(\theta, \mathbf{a})$ is eliminated without computing the estimates.

Non-data aided MCTL

We compute now the derivative of (14.203) in exact form,

$$\frac{\partial \ell_\varepsilon(e)}{\partial e} = 2Re\left[\sum_{k=0}^{K-1} y_k(e)\,\frac{d}{de}\,y_k^*(e)\right] \tag{14.206}$$

Approximating the derivative of $y_k^*(e)$ as in (14.110) we obtain

$$\frac{\partial \ell_\varepsilon(e)}{\partial e} \simeq \frac{1}{\delta}\sum_{k=0}^{K-1} Re[y_k(e)(y_k(e+\delta) - y_k(e-\delta))^*] \tag{14.207}$$

Assuming the loop filter performs the multiplication by $1/\delta$ and the sum, the error signal is given by

$$e(kT) = Re[y_k(\hat{\varepsilon}_k)(y_k(\hat{\varepsilon}_k + \delta) - y(\hat{\varepsilon}_k - \delta))^*] \tag{14.208}$$

The block diagram of the estimator is shown in Figure 14.40 and is called *modified code tracking loop* (MCTL) [10]; also in this case the estimator is non-coherent.

Data- and phase-directed: coherent DLL

In case the estimates $\hat{\theta}$ and $\hat{\mathbf{a}}$ are given, the log-likelihood is expressed as

$$\ell_\varepsilon(e) = \sum_{k=0}^{K-1} Re[\hat{a}_k^*\, y_k(e)\, e^{-j\hat{\theta}}] \tag{14.209}$$

Approximating the derivative as in (14.110) and including both the multiplicative constant and the summation in the loop filter, the error signal is given by

$$e(kT) = Re[\hat{a}_k^*\, e^{-j\hat{\theta}}[y_k(\hat{\varepsilon}_k + \delta) - y_k(\hat{\varepsilon}_k - \delta)]] \tag{14.210}$$

Figure 14.41 illustrates the block diagram of the estimator, which is called coherent DLL [9, 10, 11], as the error signal is obtained by the estimates $\hat{\theta}$ and $\hat{\mathbf{a}}$.

In the three schemes of Figure 14.39, 14.40, and 14.41 the direct section of the DLL gives estimates of $\mathbb{m}_m$ and μ_m at every symbol period T, whereas the feedback loop may operate at the chip period T_{chip}. Observe that by removing the decimation blocks the DLL is able to provide timing estimates at every chip period.

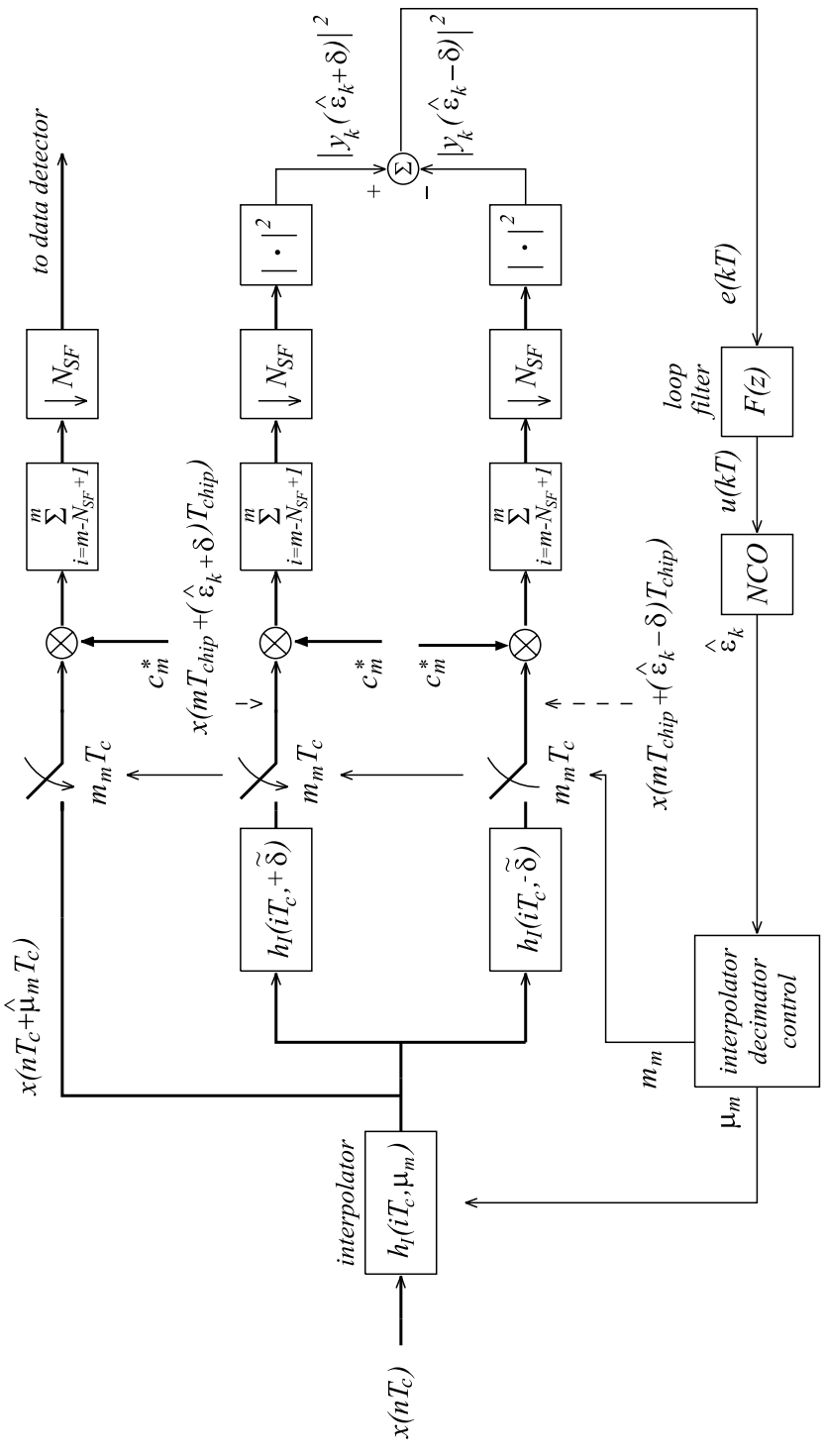


Figure 14.39. Non-coherent DLL.

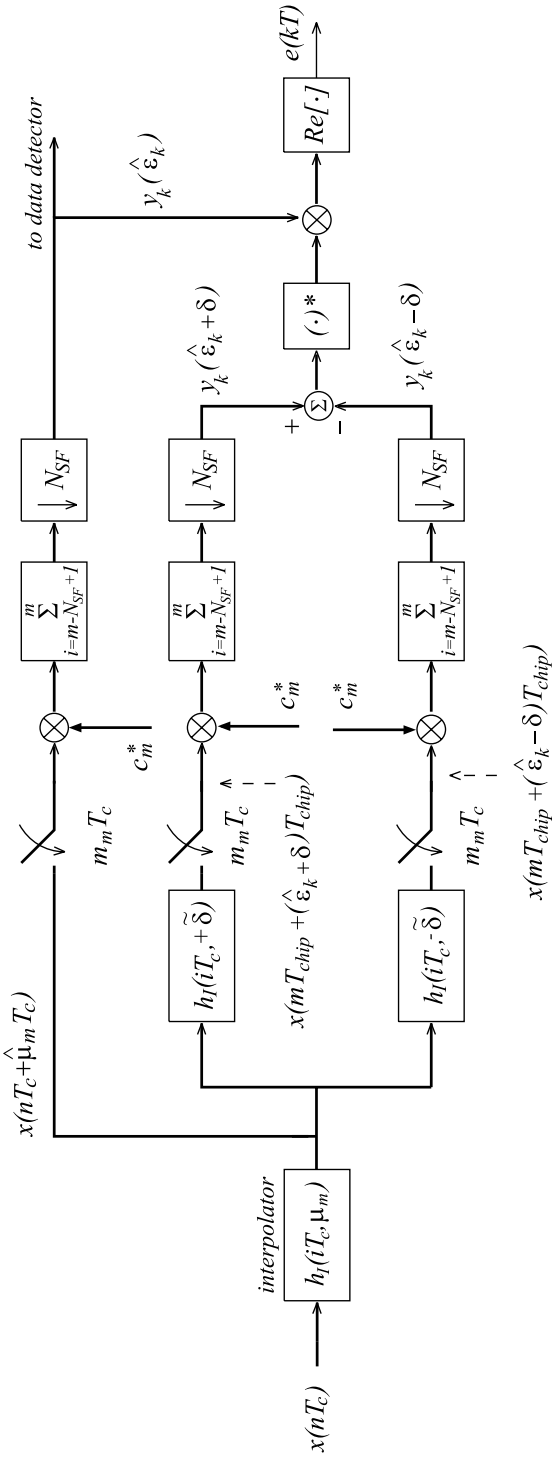


Figure 14.40. Direct section of the non-coherent MCTL.

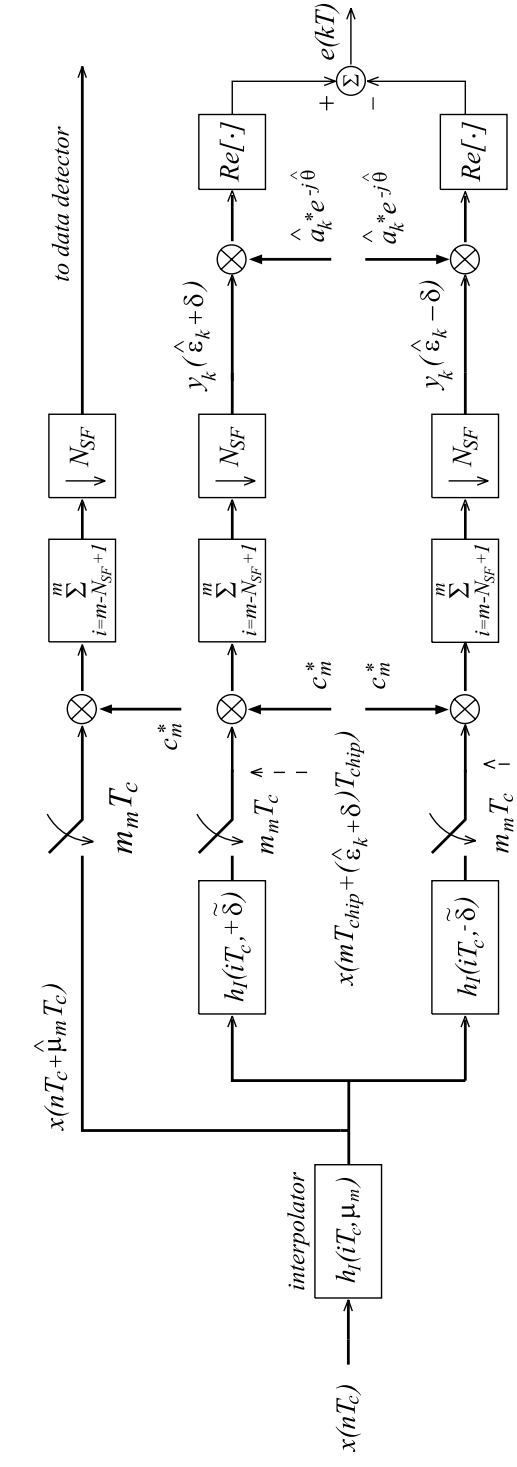


Figure 14.41. Direct section of the coherent DLL.

Bibliography

[1] H. Meyr and G. Ascheid, *Synchronization in digital communications*, vol. 1. New York: John Wiley & Sons, 1990.

[2] L. E. Franks, "Carrier and bit synchronization in data communication—A tutorial review", *IEEE Trans. on Communications*, vol. 28, pp. 1107–1120, Aug. 1980.

[3] J. G. Proakis, *Digital communications*. New York: McGraw-Hill, 3rd ed., 1995.

[4] H. Meyr, M. Moeneclaey, and S. A. Fechtel, *Digital communication receivers*. New York: John Wiley & Sons, 1998.

[5] U. Mengali and A. N. D'Andrea, *Synchronization techniques for digital receivers*. New York: Plenum Press, 1997.

[6] K. H. Mueller and M. S. Muller, "Timing recovery in digital synchronous data receivers", *IEEE Trans. on Communications*, vol. 24, pp. 516–531, May 1976.

[7] M. K. Simon, J. K. Omura, R. A. Scholtz, and B. K. Levitt, *Spread spectrum communications handbook*. New York: McGraw-Hill, 1994.

[8] R. De Gaudenzi, M. Luise, and R. Viola, "A digital chip timing recovery loop for band-limited direct-sequence spread-spectrum signals", *IEEE Trans. on Communications*, vol. 41, pp. 1760–1769, Nov. 1993.

[9] R. De Gaudenzi, "Direct-sequence spread-spectrum chip tracking loop in the presence of unresolvable multipath components", *IEEE Trans. on Vehicular Technology*, vol. 48, pp. 1573–1583, Sept. 1999.

[10] R. A. Yost and R. W. Boyd, "A modified PN code tracking loop: its performance analysis and comparative evaluation", *IEEE Trans. on Communications*, vol. 30, pp. 1027–1036, May 1982.

[11] R. De Gaudenzi and M. Luise, "Decision-directed coherent delay-lock tracking loop for DS-spread spectrum signals", *IEEE Trans. on Communications*, vol. 39, pp. 758–765, May 1991.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 15

Self-training equalization

By the expression *self-training equalization* we refer to channel equalization techniques by which we obtain the initial convergence of the parameters of an adaptive equalizer without resorting to the transmission of a training sequence. Although these techniques generally achieve sub-optimum performance at convergence, they are applied in many cases of practical interest, where the transmission of known training sequences is not viable, as for example in broadcast systems, or may result in an undesirable increase of system complexity [1, 2, 3, 4, 5, 6]. In these cases it is necessary to consider self-training of the adaptive equalizer, where the terms for adjusting the coefficient parameters are obtained by processing the received signal. Usually the receiver performs self-training of the equalizer by referring to the knowledge of the input signal characteristics, for example, the probability distribution of the input symbols.

The subject of self-training equalization in communications systems has received considerable attention since the publication of the article by Sato [7] in 1975; the proposed algorithms have been alternatively called *self-recovering*, *self-adaptive*, or *blind*.

15.1 Problem definition and fundamentals

With reference to the discrete-time equivalent scheme of Figure 7.18 shown in Figure 15.1, we consider a real-valued discrete-time system given by the cascade of a linear channel $\{h_i\}$, in general with non-minimum phase transfer function[1] H, and an equalizer $C \leftrightarrow \{c_i\}$; in general, the impulse responses $\{h_i\}$ and $\{c_i\}$ are assumed with unlimited duration. The overall system is given by $\Psi = H \cdot C \leftrightarrow \{\psi_i\}$, where $\psi_i = \sum_{\ell=-\infty}^{\infty} c_\ell h_{i-\ell}$. The channel input symbol sequence $\{a_k\}$ is modeled as a sequence of i.i.d. random variables with symmetric probability density function $p_a(\alpha)$. The channel output sequence is $\{x_k\}$. In this section, additive noise introduced by the channel is ignored.

We note that a non-minimum phase rational transfer function (see Definition 1.8 on page 26) can be expressed as

$$H(z) = H_0 \frac{P_1(z)\, P_2(z)}{P_3(z)} \tag{15.1}$$

[1] In this chapter, to simplify notation, often the argument of the z-transform is not explicitly indicated.

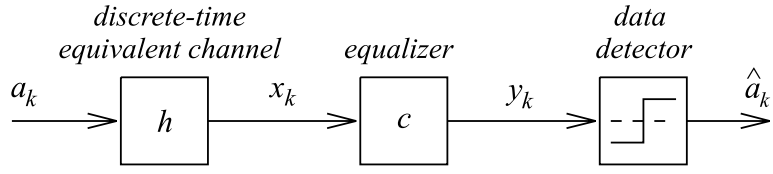


Figure 15.1. Discrete-time equivalent channel and equalizer filter.

where H_0 denotes the gain, $P_3(z)$ and $P_1(z)$ are monic polynomials with zeros inside the unit circle, and $P_2(z)$ is a monic polynomial with zeros outside the unit circle. We introduce the inverse functions with respect to the polynomials $P_1(z)$ and $P_2(z)$ given by $P_1^{-1}(z) = 1/P_1(z)$ and $P_2^{-1}(z) = 1/P_2(z)$, respectively. By expanding $P_1^{-1}(z)$ and $P_2^{-1}(z)$ in Laurent series we obtain, apart from lag factors,

$$\begin{aligned} P_1^{-1}(z) &= \sum_{n=0}^{+\infty} c_{1,n}\, z^{-n} \\ P_2^{-1}(z) &= \sum_{n=-\infty}^{0} c_{2,n}\, z^{-n} \end{aligned} \tag{15.2}$$

both converging in a ring that includes the unit circle. Therefore we have

$$H^{-1}(z) = \frac{1}{H_0}\, P_3(z) \left(\sum_{n=0}^{+\infty} c_{1,n}\, z^{-n} \right) \left(\sum_{n=-\infty}^{0} c_{2,n}\, z^{-n} \right) \tag{15.3}$$

In general from (15.3) we note that, as the system is non-minimum phase, the inverse system cannot be described by a finite number of parameters; in other words, the reconstruction of a transmitted symbol at the equalizer output at a certain instant requires the knowledge of the entire received sequence.

In practice, to obtain an implementable system, the series in (15.3) are truncated to N terms, and an approximation of H^{-1} with a lag equal to $(N_1 - 1)$ modulation intervals is given by

$$H^{-1}(z) \simeq \frac{1}{H_0}\, z^{-(N_1-1)}\, P_3(z) \left(\sum_{n=0}^{N-1} c_{1,n}\, z^{-n} \right) \left(\sum_{n=-(N-1)}^{0} c_{2,n}\, z^{-n} \right) \tag{15.4}$$

Therefore the inverse system can only be defined apart from a lag factor.

The problem of self-training equalization is formulated as follows: from the knowledge of the probability distribution of the channel input symbols $\{a_k\}$ and from the observation of the channel output sequence $\{x_k\}$, we want to find an equalizer C such that the overall system impulse response is ISI free.

Observe that, if the channel H is minimum phase,[2] both H and H^{-1} are causal and stable; in this case the problem of channel identification, and hence the determination of

[2] We recall that all AR models (1.518) are minimum phase.

the sequence $\{a_k\}$, can be solved by whitening (see page 143) the observation $\{x_k\}$ using known procedures that are based on the second-order statistical description of signals. If, as it happens in the general case, the channel H is non-minimum phase, by the second-order statistical description (1.263) it is possible to identify only the amplitude characteristic of the channel transfer function, but not its phase characteristic.

In particular, if the probability density function of the input symbols is Gaussian, the output $\{x_k\}$ is also Gaussian and the process is completely described by a second-order analysis; therefore the above observations are valid and in general the problem of self-training equalization cannot be solved for Gaussian processes. Note that we have referred to a system model in which the sampling frequency is equal to the symbol rate; a solution to the problem using a second-order description with reference to an oversampled model is obtained in [8].

Furthermore, we observe that, as the probability density function of the input symbols is symmetric, the sequence $\{-a_k\}$ has the same statistical description as the sequence $\{a_k\}$; consequently it is not possible to distinguish the desired equalizer H^{-1} from the equalizer $-H^{-1}$. Hence, the inverse system can only be determined apart from the sign and a lag factor. Therefore the solution to the problem of self-training equalization is given by $C = \pm H^{-1}$, which yields the overall system $\Psi = \pm I$, where I denotes the identity, with the exception of a possible lag.

In this chapter we will refer to the following theorem, demonstrated in [9].

Theorem 15.1
Assuming that the probability density function of the input symbols is non-Gaussian, then $\Psi = \pm I$ if the output sample

$$y_k = \sum_{n=-\infty}^{+\infty} c_n \, x_{k-n} \tag{15.5}$$

has a probability density function $p_{y_k}(b)$ equal to the probability density function of the symbols $\{a_k\}$.

Therefore, using this theorem to obtain the solution, it is necessary to determine an algorithm for the adaptation of the coefficients of the equalizer C such that the probability distribution of y_k converges to the distribution of a_k.

We introduce the cost function

$$J = E[\Phi(y_k)] \tag{15.6}$$

where y_k is given by (15.5) and Φ is an even, real-valued function that must be chosen so that the optimum solution, determined by

$$C_{opt}(z) = \arg\min_{C(z)} J \tag{15.7}$$

is found at the points $\pm H^{-1}$, apart from a lag factor.

In most applications, minimization is done by equalizers of finite length N having coefficients at instant k given by

$$\mathbf{c}_k = [c_{0,k}, c_{1,k}, \ldots, c_{N-1,k}]^T \tag{15.8}$$

Let

$$\mathbf{x}_k = [x_k, x_{k-1}, \ldots, x_{k-(N-1)}]^T \tag{15.9}$$

then (15.5) becomes

$$y_k = \sum_{n=0}^{N-1} c_{n,k}\, x_{k-n} = \mathbf{c}_k^T\, \mathbf{x}_k \tag{15.10}$$

Then also (15.7) simplifies into

$$\mathbf{c}_{opt} = \arg\min_{\mathbf{c}} J \tag{15.11}$$

If Θ is the derivative of Φ, the gradient of J with respect to $\mathbf{c}$ is given by (see (2.18))

$$\nabla_{\mathbf{c}} J = E[\mathbf{x}_k\, \Theta(y_k)] \tag{15.12}$$

For the minimization of J we use a stochastic gradient algorithm for which (see (3.40))

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu\, \Theta(y_k)\, \mathbf{x}_k \tag{15.13}$$

where μ is the adaptation gain. Note that the convergence of C to H^{-1} or to $-H^{-1}$ depends on the initial choice of the coefficients.

Before tackling the problem of choosing the function Φ, we consider the following problem.

Minimization of a special function

We consider an overall system $\boldsymbol{\psi}$ with impulse response having unlimited duration. We want to determine the values of the sequence $\{\psi_i\}$ that minimize the following cost function:

$$\mathcal{V} = \sum_{i=-\infty}^{+\infty} |\psi_i| \tag{15.14}$$

subject to the constraint

$$\sum_{i=-\infty}^{+\infty} \psi_i^2 = 1 \tag{15.15}$$

The function $\mathcal{V}$ characterizes the *peak* amplitude of the system *output signal* $\{y_k\}$ and the constraint can be interpreted as a requirement that the solution $\{\psi_i\}$ belongs to the sphere with center the origin and radius $r = 1$ in the parameter space $\{\psi_i\}$.

Letting $\boldsymbol{\psi} = [\ldots, \psi_0, \psi_1, \ldots]^T$, it results in $\nabla_{\boldsymbol{\psi}} \mathcal{V} = [\ldots, \mathrm{sgn}(\psi_0), \mathrm{sgn}(\psi_1), \ldots]^T$; then, if ψ_{max} is the maximum value assumed by ψ_i, the cost function (15.14) presents stationary points for $\psi_i = \pm\psi_{max}$. Now, taking into account the constraint (15.15), it is easily seen that the minimum of (15.14) is reached only when one element of the sequence $\{\psi_i\}$ is different from zero (with a value equal to ψ_{max}) and the others are all zeros. In other words, the only points of minimum are given by $\Psi = \pm I$, and the other stationary points correspond to saddle points. Figure 15.2 illustrates the cost function $\mathcal{V}$ along the unit circle for a system with two parameters.

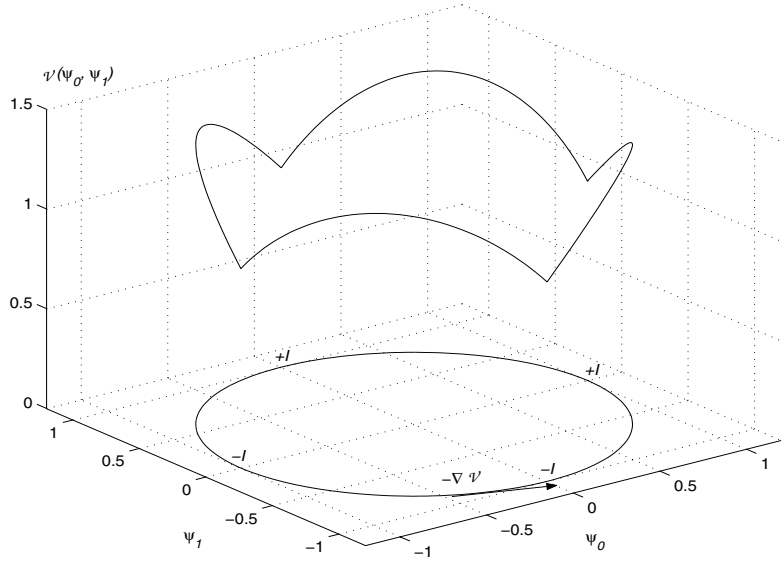


Figure 15.2. Illustration of the cost function $\mathcal{V}$ for the system $\Psi \leftrightarrow \{\psi_0, \psi_1\}$ and of the gradient $-\nabla\mathcal{V}$ projected onto the straight line tangent to the unit circle.

The minimization can also be obtained recursively, updating the parameters $\{\psi_i\}$ as indicated by the direction of steepest descent[3] of the cost function $\mathcal{V}$.

The projection of the gradient $\mathcal{V}$ onto the plane tangent to the unit sphere at the point $\boldsymbol{\psi}$ is given by

$$\nabla_{\boldsymbol{\psi}}\mathcal{V} - [(\nabla_{\boldsymbol{\psi}}\mathcal{V})^T\boldsymbol{\psi}]\boldsymbol{\psi} \tag{15.16}$$

then the recursive equation yields

$$\psi'_{i,k+1} = \psi_{i,k} - \mu\left[\operatorname{sgn}(\psi_{i,k}) - \psi_{i,k}\sum_{\ell=-\infty}^{+\infty}|\psi_{\ell,k}|\right] \tag{15.17}$$

$$\psi_{i,k+1} = \frac{\psi'_{i,k+1}}{\sqrt{\sum_{\ell=-\infty}^{+\infty}(\psi'_{\ell,k+1})^2}} \tag{15.18}$$

Note that, if the term $\psi_{i,k}\sum_\ell|\psi_{\ell,k}|$ is omitted in (15.17), with good approximation the direction of the steepest descent is still followed, provided that the adaptation gain μ is sufficiently small.

From (15.17), for each parameter ψ_i the updating consists of a correction toward zero of a fixed value and a correction in the opposite direction of a value proportional to the

[3] The direction is defined by the gradient vector.

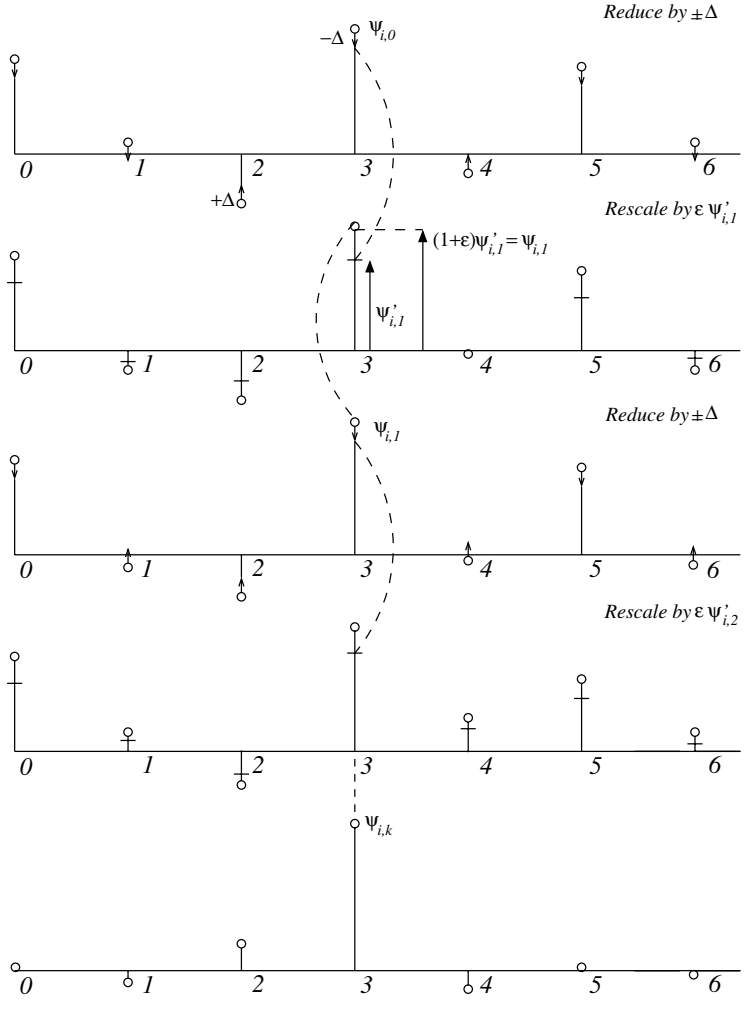


Figure 15.3. Update of the parameters $\{\psi_i\}$, $i = 0, \ldots, 6$.

parameter amplitude. Assuming that the initial point is not a saddle point, by repeated iterations of the algorithm, one of the parameters ψ_i approaches the value one, while all others converge to zero, as shown in Figure 15.3.

We now want to obtain the same adaptation rule for the parameters $\{\psi_i\}$ using the output signal y_k. We define $\boldsymbol{\psi}_k$ as the vector of the parameters of $\boldsymbol{\psi}$ at instant k,

$$\boldsymbol{\psi}_k = [\ldots, \psi_{0,k}, \psi_{1,k}, \ldots]^T \tag{15.19}$$

and

$$\mathbf{a}_k = [\ldots, a_k, a_{k-1}, \ldots]^T \tag{15.20}$$

Therefore

$$y_k = \boldsymbol{\psi}_k^T \, \mathbf{a}_k \tag{15.21}$$

Assume that at the beginning of the adaptation process the overall system Ψ satisfies the condition $\|\boldsymbol{\psi}\|^2 = 1$, but it deviates significantly from the system identity; then the equalizer output signal y_k will occasionally assume positive or negative values which are much larger in magnitude than $\alpha_{max} = \max a_k$. The peak value of y_k is given by $\pm\alpha_{max} \sum_i |\psi_{i,k}|$, obtained with symbols $\{a_{k-i}\}$ equal to $\pm\alpha_{max}$, and indicates that the distortion is too large and must be reduced. In this case a correction of a fixed value towards zero is obtained using the error signal

$$e_k = y_k - \alpha_{max} \operatorname{sgn}(y_k) \tag{15.22}$$

and applying the stochastic gradient algorithm

$$\boldsymbol{\psi}_{k+1} = \boldsymbol{\psi}_k - \mu \, e_k \, \mathbf{a}_k \tag{15.23}$$

If the coefficients are scaled so that the condition $\|\boldsymbol{\psi}_k\|^2 = 1$ is satisfied at every k, we obtain a coefficient updating algorithm that approximates algorithms (15.17) and (15.18).

Obviously algorithm (15.23) cannot be directly applied, as the parameters of the overall system Ψ are not available. However, observe that if the linear transformation H is non-singular, then formally $C = H^{-1}\Psi$. Therefore the overall minima of $\mathcal{V}$ at the points $\Psi = \pm I$ are mapped into overall minima at the points $C = \pm H^{-1}$ of a cost function J that is the image of $\mathcal{V}$ under the transformation given by H^{-1}, as illustrated in Figure 15.4. Furthermore, it is seen that the direction of steepest descent of $\mathcal{V}$ has not been modified by this transformation. Thus the updating terms for the equalizer coefficients are still given by (15.22) and (15.23), if symbols a_k are replaced by the channel output samples x_k.

Then, a general algorithm that converges to the desired solution $C = H^{-1}$ can be formulated as follows:

- observe the equalizer output signal $\{y_k\}$ and determine its peak value;

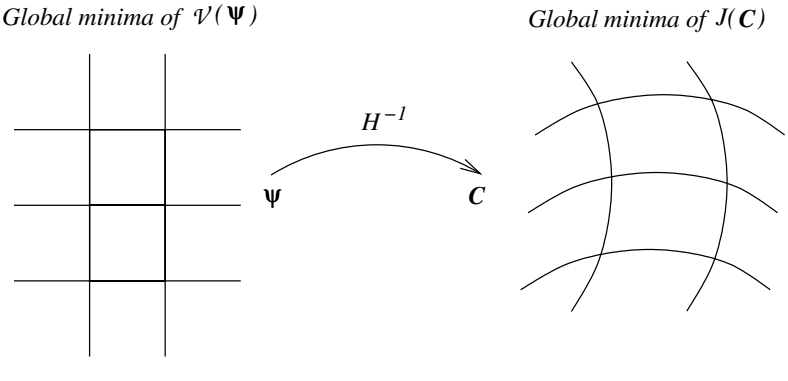


Figure 15.4. Illustration of the transformation $C = H^{-1}\Psi$.

- whenever a peak value occurs, update the coefficients according to the algorithm

$$\begin{aligned} e_k &= y_k - \alpha_{max}\,\mathrm{sgn}(y_k) \\ \mathbf{c}_{k+1} &= \mathbf{c}_k - \mu\, e_k\, \mathbf{x}_k \end{aligned} \tag{15.24}$$

- scale the coefficients so that the statistical power of the equalizer output samples is equal to the statistical power of the channel input symbols.

We observe that it is not practical to implement an algorithm that requires computing the peak value of the equalizer output signal and updating the coefficients only when a peak value is observed. In the next sections we describe algorithms that allow the updating of the coefficients at every modulation interval, thus avoiding the need of computing the peak value of the signal at the equalizer output and of scaling the coefficients.

15.2 Three algorithms for PAM systems

The Sato algorithm

The Sato cost function is defined as

$$J = E\left[\tfrac{1}{2}\, y_k^2 - \gamma_S\, |y_k|\right] \tag{15.25}$$

where

$$\gamma_S = \frac{E[a_k^2]}{E[|a_k|]} \tag{15.26}$$

The gradient of J is given by

$$\nabla_{\mathbf{c}} J = E[\mathbf{x}_k(y_k - \gamma_S\,\mathrm{sgn}(y_k))] \tag{15.27}$$

We introduce the signal

$$\epsilon_{S,k} = y_k - \gamma_S\,\mathrm{sgn}(y_k) \tag{15.28}$$

which assumes the meaning of *pseudo error* that during self-training replaces the error used in the LMS decision directed algorithm. We recall that for the LMS algorithm the error signal is[4]

$$e_k = y_k - \hat{a}_k \tag{15.29}$$

where $\hat{a}_k$ is the detection of the symbol a_k, obtained by a threshold detector from the sample y_k. Figure 15.5 shows the pseudo error $\epsilon_{S,k}$ as a function of the value of the equalizer output sample y_k.

[4] Note that in this chapter, the error signal is defined with opposite sign with respect to the previous chapters.

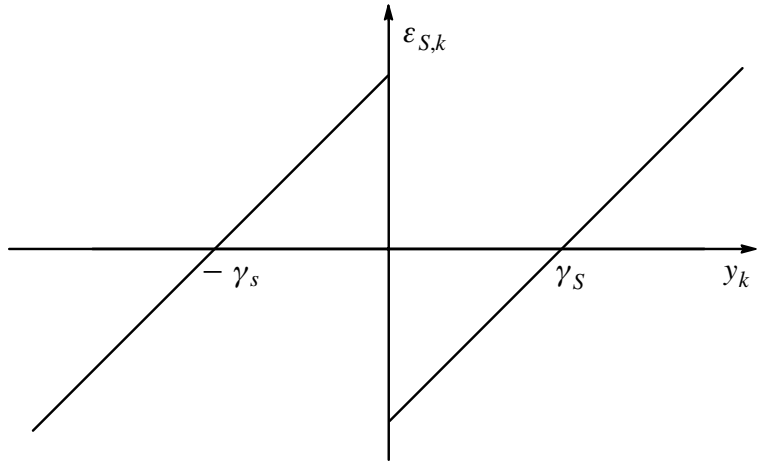


Figure 15.5. Characteristic of the pseudo error $\epsilon_{S,k}$ as a function of the equalizer output.

Therefore the Sato algorithm for the coefficient updating of an adaptive equalizer assumes the expression

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu\, \epsilon_{S,k}\, \mathbf{x}_k \tag{15.30}$$

It was proved by Benveniste, Goursat, and Ruget [9] that, if the probability density function of the output symbols $\{a_k\}$ is sub-Gaussian,[5] then the Sato cost function (15.25) admits as unique points of minimum the systems $C = \pm H^{-1}$, apart from a possible lag; however, note that the uniqueness of the points of minimum of the Sato cost function is obtained by assuming a continuous probability distribution of input symbols. In the case of a discrete probability distribution with an alphabet $\mathcal{A} = \{\pm 1, \pm 3, \ldots, \pm(M-1)\}$, the convergence properties of the Sato algorithm are not always satisfactory.

Another undesirable characteristic of the Sato algorithm is that the pseudo error $\epsilon_{S,k}$ is not equal to zero for $C = \pm H^{-1}$, unless we consider binary transmission. In fact only the gradient of the cost function given by (15.27) is equal to zero for $C = \pm H^{-1}$; moreover, we find that the variance of the pseudo error may assume non-negligible values in the neighborhood of the desired solution.

Benveniste–Goursat algorithm

To mitigate the above mentioned inconvenience, we observe that the error e_k used in the LMS algorithm in the absence of noise becomes zero for $C = \pm H^{-1}$. It is possible to combine the two error signals obtaining the pseudo error proposed by Benveniste and Goursat [10], given by

$$\epsilon_{G,k} = \kappa_1\, e_k + \kappa_2\, |e_k|\, \epsilon_{S,k} \tag{15.31}$$

[5] A probability density function $p_{a_k}(\alpha)$ is sub-Gaussian if it is uniform or if $p_{a_k}(\alpha) = K \exp\{-g(\alpha)\}$, where $g(\alpha)$ is an even function such that both $g(\alpha)$ and $\frac{1}{\alpha}\frac{dg}{d\alpha}$ are strictly increasing in the domain $[0, +\infty)$.

where κ_1 and κ_2 are constants. If the distortion level is high, $|e_k|$ assumes high values and the second term of (15.31) allows convergence of the algorithm during the self-training. Near convergence, for $C \simeq \pm H^{-1}$, the second term has the same order of magnitude as the first and the pseudo error assumes small values in the neighborhood of $C = \pm H^{-1}$.

Note that an algorithm that uses the pseudo error (15.31) allows a smooth transition of the equalizer from the self-training mode to the decision-directed mode. In the case of a sudden change in channel characteristics, the equalizer is found working again in self-training mode. Thus the transitions between the two modes occur without control on the level of distortion of the signal $\{y_k\}$ at the equalizer output.

Stop-and-go algorithm

The stop-and-go algorithm proposed by Picchi and Prati [11] can be seen as a variant of the Sato algorithm that achieves the same objectives of the Benveniste–Goursat algorithm with better convergence properties. The pseudo error for the stop-and-go algorithm is formulated as

$$\epsilon_{P,k} = \begin{cases} e_k & \text{if } \operatorname{sgn}(e_k) = \operatorname{sgn}(\epsilon_{S,k}) \\ 0 & \text{otherwise} \end{cases} \tag{15.32}$$

where $\epsilon_{S,k}$ is the Sato pseudo error given by (15.28), and e_k is the error used in the decision-directed algorithm given by (15.29). The basic idea is that the algorithm converges if updating of the equalizer coefficients is turned off with sufficiently high probability every time the sign of error (15.29) differs from the sign of error $e_{id,k} = y_k - a_k$, that is $\operatorname{sgn}(e_k) \neq \operatorname{sgn}(e_{id,k})$. As $e_{id,k}$ is not available in a self-training equalizer, with the stop-and-go algorithm coefficient updating is turned off whenever the sign of error e_k is different from the sign of Sato error $\epsilon_{S,k}$. Obviously, in this way we also get a non-zero probability that coefficient updating is inactive when the condition $\operatorname{sgn}(e_k) = \operatorname{sgn}(e_{id,k})$ occurs, but this does not usually bias the convergence of the algorithm.

Remarks

At this point we can make the following observations.

- Self-training algorithms based on the minimization of a cost function that includes the term $E[|y_k|^p]$, $p \geq 2$, can be explained referring to the algorithm (15.24), because the effect of raising to the p-th power the amplitude of the equalizer output sample is that of emphasizing the contribution of samples with large amplitude.

- Extension of the Sato cost function (15.25) to QAM systems, which we discuss in Section 15.5, is given by

$$J = E\left[\tfrac{1}{2}\,|y_k|^2 - \gamma_S\,|y_k|\right] \tag{15.33}$$

  where $\gamma_S = E[|a_k|^2]/E[|a_k|]$. In general this term guarantees that, at convergence, the statistical power of the equalizer output samples is equal to the statistical power of the input symbols.

- In the algorithm (15.24), the equalizer coefficients are updated only when we observe a peak value of the equalizer output signal. As the peak value decreases with the progress of the equalization process, updating of the equalizer coefficients ideally depends on a threshold that varies depending on the level of distortion in the overall system impulse response.

15.3 The contour algorithm for PAM systems

The algorithm (15.24) suggests that the equalizer coefficients are updated when the equalizer output sample reaches a threshold value, which in turn depends on the level of distortion in the overall system impulse response. In practice, we define a threshold at instant k as $T_k = \alpha_{max} + \gamma_{CA,k}$, where the term $\gamma_{CA,k} \geq 0$ represents a suitable measure of distortion. Each time the absolute value of the equalizer output sample reaches or exceeds the threshold T_k, the coefficients are updated so that the peak value of the equalizer output is "driven" towards the constellation boundary $\pm\alpha_{max}$; at convergence of the equalizer coefficients, $\gamma_{CA,k}$ vanishes. Figure 15.6 illustrates the evolution of the "contour" T_k for a two-dimensional constellation.

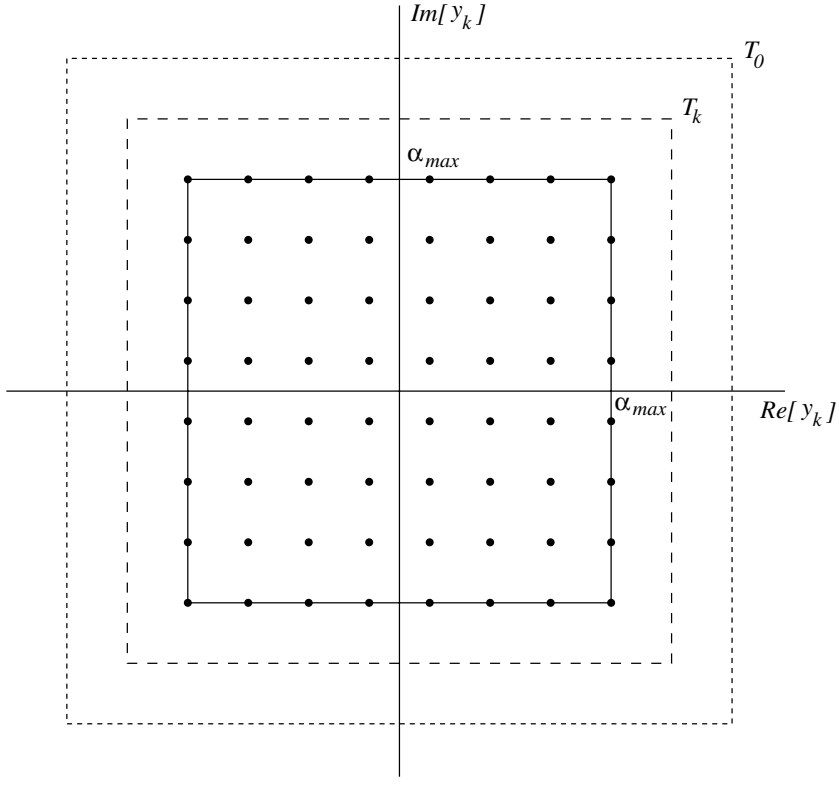


Figure 15.6. Evolution in time of the "contour" T_k for a two-dimensional constellation.

The updating of coefficients described above can be obtained by a stochastic gradient algorithm that is based on a cost function $E[\Phi(y_k)]$ defined on the parameter space $\{\psi_i\}$. We assume that the overall system Ψ initially corresponds to a point on a sphere of arbitrary radius r. With updating terms that on the average exhibit the same sign as the terms found in the general algorithm (15.24), the point on the sphere of radius r moves in such a way to reduce distortion. Moreover, if the radial component of the gradient, i.e. the component that is orthogonal to the surface of the sphere, is positive for $r > 1$, negative for $r < 1$ and vanishes on the sphere of radius $r = 1$, it is not necessary to scale the coefficients and the convergence will take place to the point of global minimum $\Psi = \pm I$. Clearly, the derivative function of $\Phi(y_k)$ with respect to y_k that defines the pseudo error can be determined in various ways. A suitable definition is given by

$$\Theta(y_k) = \begin{cases} y_k - [\alpha_{max} + \gamma_{CA,k}]\,\mathrm{sgn}(y_k) & \text{if } |y_k| \geq \alpha_{max} \\ -\gamma_{CA,k}\,\mathrm{sgn}(y_k) & \text{otherwise} \end{cases} \tag{15.34}$$

For a self-training equalizer, the updating of coefficients as indicated by (15.8) is obtained by the algorithm

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu\,\Theta(y_k)\,\mathbf{x}_k \tag{15.35}$$

In this algorithm, to avoid the computation of the threshold T_k at each iteration, the amplitude of the equalizer output signal is compared with α_{max} rather than with T_k; note that the computation of $\Theta(y_k)$ depends on the event that y_k falls inside or outside the constellation boundary. In the next section, for a two-dimensional constellation, we define in general the *constellation boundary* as the contour line that connects the outer points of the constellation; for this reason we refer to this algorithm as the *contour algorithm* [12].

To derive the algorithms (15.34) and (15.35) from the algorithm (15.24) several approximations are introduced; consequently the convergence properties cannot be directly derived from those of the algorithm (15.24). In Appendix 15.A we show how γ_{CA} should be defined to obtain the desired behavior of the algorithms, (15.34) and (15.35) in the case of systems with input symbols having a uniform continuous distribution.

An advantage of the functional introduced in this section with respect to the Sato cost function is that the variance of the pseudo error vanishes at the points of minimum $\Psi = \pm I$; this means that it is possible to obtain the convergence of the MSE to a steady state value that is close to the achievable minimum value. Furthermore, the radial component of the gradient of $E[\Phi(y_k)]$ vanishes at every point on the unit sphere, whereas the radial component of the gradient in the Sato cost function vanishes on the unit sphere only at the points $\Psi = \pm I$. As the direction of steepest descent does not intersect the unit sphere, the contour algorithm avoids overshooting of the convergence trajectories observed using the Sato algorithm; in other words, the stochastic gradient yields a coefficient updating that is made in the correct direction more often than in the case of the Sato algorithm. Therefore, substantially better convergence properties are expected for the contour algorithm even in systems with a discrete probability distribution of input symbols.

The complexity of the algorithms (15.34) and (15.35) can be deemed prohibitive for practical implementations, especially for self-training equalization in high-speed communication systems, as the parameter $\gamma_{CA,k}$ must be estimated at each iteration. In the next

section, we discuss a simplified algorithm that allows implementation with low complexity; we will see later how the simplified formulation of the contour algorithm can be extended to self-training equalization of partial response and QAM systems.

Simplified realization of the contour algorithm

We assume that the input symbols $\{a_k\}$ form a sequence of i.i.d. random variables with a uniform discrete probability density function. From the scheme of Figure 15.1, in the presence of noise, the channel output signal is given by

$$x_k = \sum_{i=-\infty}^{\infty} h_i \, a_{k-i} + w_k \tag{15.36}$$

where $\{w_k\}$ denotes additive white Gaussian noise. The equalizer output is given by $y_k = \mathbf{c}_k^T \mathbf{x}_k$. To obtain an algorithm that does not require the knowledge of the parameter γ_{CA}, the definition (15.34) suggests introducing the pseudo error

$$\epsilon_{CA,k} = \begin{cases} y_k - \alpha_{max} \operatorname{sgn}(y_k) & \text{if } |y_k| \geq \alpha_{max} \\ -\delta_k \operatorname{sgn}(y_k) & \text{otherwise} \end{cases} \tag{15.37}$$

where δ_k is a non-negative parameter that is updated at every iteration as follows:

$$\delta_{k+1} = \begin{cases} \delta_k - \dfrac{M-1}{M} \Delta & \text{if } |y_k| \geq \alpha_{max} \\ \delta_k + \dfrac{1}{M} \Delta & \text{otherwise} \end{cases} \tag{15.38}$$

and Δ is a positive constant. The initial value δ_0 is not a critical system parameter and can be, for example, chosen equal to zero; the coefficient updating algorithm is thus given by

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu \, \epsilon_{CA,k} \, \mathbf{x}_k \tag{15.39}$$

In comparison to (15.34), now δ_k does not provide a measure of distortion as γ_{CA}. The definition (15.37) is justified by the fact that the term $y_k - [\alpha_{max} + \gamma_{CA}] \operatorname{sgn}(y_k)$ in (15.34) can be approximated as $y_k - \alpha_{max} \operatorname{sgn}(y_k)$, because if the event $|y_k| \geq \alpha_{max}$ occurs the pseudo error $y_k - \alpha_{max} \operatorname{sgn}(y_k)$ can be used for coefficient updating. Therefore, δ_k should increase in the presence of distortion only in the case the event $|y_k| < \alpha_{max}$ occurs more frequently than expected. This behavior of the parameter δ_k is obtained by applying (15.38). Moreover, (15.38) guarantees that δ_k assumes values that approach zero at the convergence of the equalization process; in fact, in this case the probabilities of the events $\{|y_k| < \alpha_{max}\}$ and $\{|y_k| \geq \alpha_{max}\}$ assume approximately the values $(M-1)/M$ and $1/M$, respectively, that correspond to the probabilities of such events for a noisy PAM signal correctly equalized. Figure 15.7 shows the pseudo error $\epsilon_{CA,k}$ as a function of the value of the equalizer output sample y_k.

The contour algorithm has been described for the case of uniformly distributed input symbols; however, this assumption is not necessary. In general, if $\{y_k\}$ represents an equalized signal, the terms $(M-1)/M$ and $1/M$ in (15.38) are, respectively, substituted by

$$p_0 = P[|a_k| < \alpha_{max}] + \tfrac{1}{2} P[|a_k| = \alpha_{max}] \simeq P[|y_k| < \alpha_{max}] \tag{15.40}$$

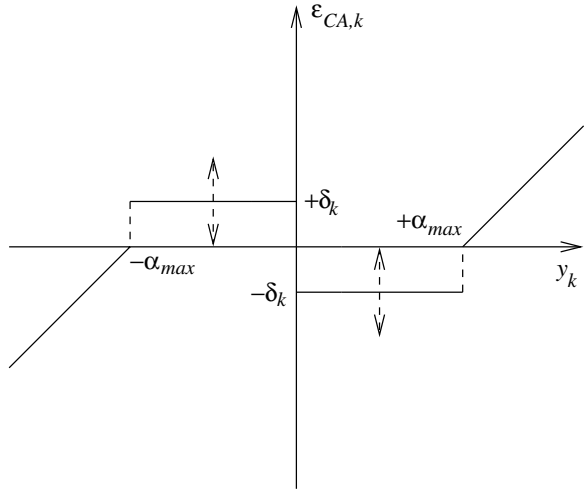


Figure 15.7. Characteristic of the pseudo error $\epsilon_{CA,k}$ as a function of the equalizer output.

and

$$p_1 = \tfrac{1}{2}\, P[|a_k| = \alpha_{max}] \simeq P[|y_k| \geq \alpha_{max}] \tag{15.41}$$

We note that the considered receiver makes use of signal samples at the symbol rate; the algorithm can also be applied for the initial convergence of a fractionally spaced equalizer (see Section 8.4) in case a sampling rate higher than the symbol rate is adopted.

15.4 Self-training equalization for partial response systems

Self-training adaptive equalization has been mainly studied for full response systems; however, self-training equalization methods for partial response systems have been proposed for linear and non-linear equalizers [7, 13, 14]. In general, a self-training equalizer is more difficult to implement for partial response systems (see Appendix 7.A), especially if the input symbol alphabet has more than two elements. Moreover, as self-training is a slow process, the accuracy achieved in the recovery of the timing of the received signal before equalization plays an important role. In fact, if timing recovery is not accomplished, because of the difference between the sampling rate and the modulation rate we have that the sampling phase varies with respect to the timing phase of the remote transmitter clock; in this case we speak of *phase drift* of the received signal; self-training algorithms fail if the phase drift of the received signal is not sufficiently small. In this section, we discuss the extension to partial response systems of the algorithms for PAM systems presented in the previous section.

The Sato algorithm for partial response systems

We consider a multilevel partial response class IV (PR-IV) system, also called a modified duobinary system. Using the D transform, the desired transfer function of the overall system

is given by $\psi(D) = (1 - D^2)$. The objective of an adaptive equalizer for a PR-IV system consists in obtaining an equalized signal of the form

$$y_k = (a_k - a_{k-2}) + w_{y,k} = u_k + w_{y,k} \tag{15.42}$$

where $w_{y,k}$ is a disturbance due to noise and residual distortion. We consider the case of quaternary modulation. Then the input symbols a_k are from the set $\{-3, -1, +1, +3\}$, and the output signal $u_k = a_k - a_{k-1}$, for an ideal channel in the absence of noise, can assume one of the seven values $\{-6, -4, -2, 0, +2, +4, +6\}$.[6]

As illustrated in Figure 15.8, to obtain a pseudo error to be employed in the equalizer coefficient updating algorithm, the equalizer output signal $\{y_k\}$ is transformed into a full-response signal $v_{S,k}$ by the linear transformation

$$v_{S,k} = y_k + \beta_S v_{S,k-2} \tag{15.43}$$

where β_S is a constant that satisfies the condition $0 < \beta_S < 1$. Then the signal $v_{S,k}$ is quantized by a quantizer with two levels corresponding to $\pm\gamma_S$, where γ_S is given by (15.26). The obtained signal is again transformed into a partial response signal that is subtracted from the equalizer output to generate the pseudo error

$$\epsilon_{S,k} = y_k - \gamma_S[\mathrm{sgn}(v_{S,k}) - \mathrm{sgn}(v_{S,k-2})] \tag{15.44}$$

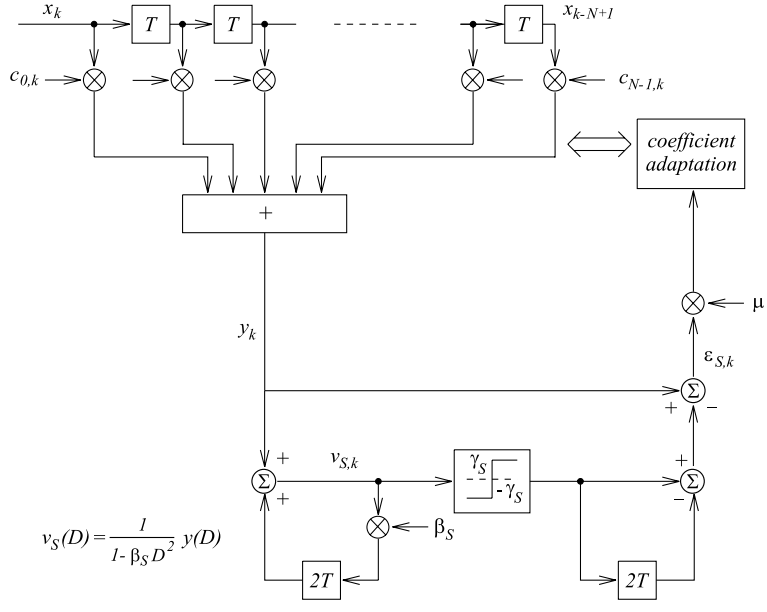


Figure 15.8. Block diagram of a self-training equalizer for a PR-IV system using the Sato algorithm.

[6] In general, for an ideal PR-IV channel in the absence of noise, if the alphabet of the input symbols is $\mathcal{A} = \{\pm 1, \pm 3, \ldots, \pm(M-1)\}$, the output symbols assume one of the $(2M-1)$ values $\{0, \pm 2, \ldots, \pm 2(M-1)\}$.

Then the Sato algorithm for partial response systems is expressed as

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu\, \epsilon_{S,k}\, \mathbf{x}_k \tag{15.45}$$

Contour algorithm for partial response systems

In this case, first, the equalizer output is transformed into a full response signal and a pseudo error is computed. Then the error to compute the terms for coefficient updating is formed.

The method differs in two ways from the Sato algorithm described above. First, the channel equalization, carried out to obtain the full response signals, is obtained by combining linear and non-linear feedback, whereas in the case of the Sato algorithm it is carried out by a linear filter. Second, the knowledge of the statistical properties of the input symbols is used to determine the pseudo error, as suggested by the contour algorithm.

As mentioned above, to apply the contour algorithm the equalizer output signal is transformed into a full response signal using a combination of linear feedback and decision feedback, that is, we form the signal

$$v_k = y_k + \beta_{CA}\, v_{k-2} + (1 - \beta_{CA})\, \hat{a}_{k-2} \tag{15.46}$$

The equation (15.46) and the choice of the parameter β_{CA} are justified in the following way. If $\beta_{CA} = 0$ is selected, we obtain an equalization system with decision feedback, that presents the possibility of significant error propagation. The effect of the choice $\beta_{CA} = 1$ is easily seen using the D transform. From (15.42) and assuming correct decisions, (15.46) can be expressed as

$$v(D) = a(D) + \frac{w_y(D)}{1 - \beta_{CA}\, D^2} \tag{15.47}$$

Therefore with $\beta_{CA} = 1$ we get the linear inversion of the PR-IV channel with infinite noise enhancement at frequencies $f = 0$ and $\pm 1/(2T)$ Hz. The value of β_{CA} is chosen in the interval $0 < \beta_{CA} < 1$, to obtain the best trade-off between linear feedback and decision feedback.

We now apply the contour algorithm (15.37)–(15.39) using the signal v_k rather than the equalizer output signal y_k. The pseudo error is defined as

$$\epsilon^v_{CA,k} = \begin{cases} v_k - \alpha_{max}\, \mathrm{sgn}(v_k) & \text{if } |v_k| \geq \alpha_{max} \\ -\delta^v_k\, \mathrm{sgn}(v_k) & \text{otherwise} \end{cases} \tag{15.48}$$

where δ^v_k is a non-negative parameter that is updated at each iteration as

$$\delta^v_{k+1} = \begin{cases} \delta^v_k - \dfrac{M-1}{M}\, \Delta & \text{if } |v_k| \geq \alpha_{max} \\ \delta^v_k + \dfrac{1}{M}\, \Delta & \text{otherwise} \end{cases} \tag{15.49}$$

The stochastic gradient must be derived taking into consideration the channel equalization performed with linear feedback and decision feedback. We define the error on the M-ary symbol a_k after channel equalization as

$$e^v_k = v_k - a_k \tag{15.50}$$

Assuming correct decisions, by (15.46) it is possible to express the equalizer output as

$$y_k = (a_k - a_{k-2}) + e_k^v - \beta_{CA}\, e_{k-2}^v \tag{15.51}$$

The equation (15.51) shows that an estimate of the term $e_k^v - \beta_{CA}\, e_{k-2}^v$ must be included as error signal in the expression of the stochastic gradient. After initial convergence of the equalizer coefficients, the estimate $\hat{e}_k^v = v_k - \hat{a}_k$ is reliable. Therefore decision-directed coefficient updating can be performed according to the algorithm

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu_{dd}(\hat{e}_k^v - \beta_{CA}\, \hat{e}_{k-2}^v)\, \mathbf{x}_k \tag{15.52}$$

The contour algorithm for coefficient updating during self-training is obtained by substituting the decision-directed error signal e_k^v with the pseudo error $\epsilon_{CA,k}^v$ (15.48),

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu(\epsilon_{CA,k}^v - \beta_{CA}\, \epsilon_{k-2}^v)\, \mathbf{x}_k \tag{15.53}$$

During self-training, satisfactory convergence behavior is usually obtained for $\beta_{CA} \simeq 1/2$. Figure 15.9 shows the block diagram of an equalizer for a PR-IV system with a quaternary alphabet (QPR-IV), with the generation of the error signals to be used in decision-directed and self-training mode.

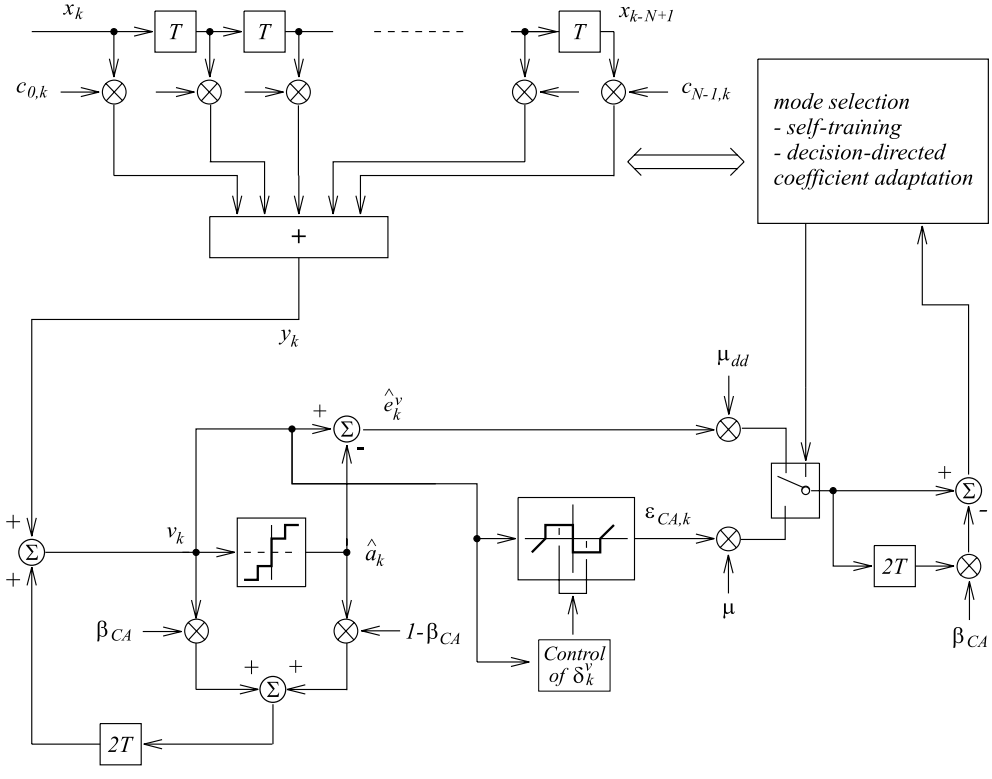


Figure 15.9. Block diagram of a self-training equalizer with the contour algorithm for a QPR-IV system.

In the described scheme, the samples of the received signal can be initially filtered by a filter with transfer function $1/(1 - aD^2)$, $0 < a < 1$, to reduce the correlation among samples. The obtained signal is then input to the equalizer delay line.

15.5 Self-training equalization for QAM systems

We now describe various self-training algorithms for passband transmission systems that employ a two-dimensional constellation.

The Sato algorithm for QAM systems

Consider a QAM transmission system with constellation $\mathcal{A}$ and a sequence $\{a_k\}$ of i.i.d. symbols from $\mathcal{A}$ such that $a_{k,I} = Re[a_k]$ and $a_{k,Q} = Im[a_k]$ are independent and have the same probability distribution. We assume a receiver in which sampling of the received signal occurs at the symbol rate and tracking of the carrier phase is carried out at the equalizer output, as shown in Figure 15.10 (see scheme of Figure 8.37 with a baseband equalizer filter where $\{x_k\}$ is already demodulated and $\theta_k = \hat{\varphi}_k$). If $\tilde{y}_k = \mathbf{c}^T \mathbf{x}_k$ is the equalizer filter output, the sample at the decision point is then given by

$$y_k = \tilde{y}_k \, e^{-j\hat{\varphi}_k} \tag{15.54}$$

We let

$$y_{k,I} = Re[y_k] \quad \text{and} \quad y_{k,Q} = Im[y_k] \tag{15.55}$$

and introduce the Sato cost function for QAM systems,

$$J = E[\Phi(y_{k,I}) + \Phi(y_{k,Q})] \tag{15.56}$$

where

$$\Phi(v) = \tfrac{1}{2} v^2 - \gamma_S |v| \tag{15.57}$$

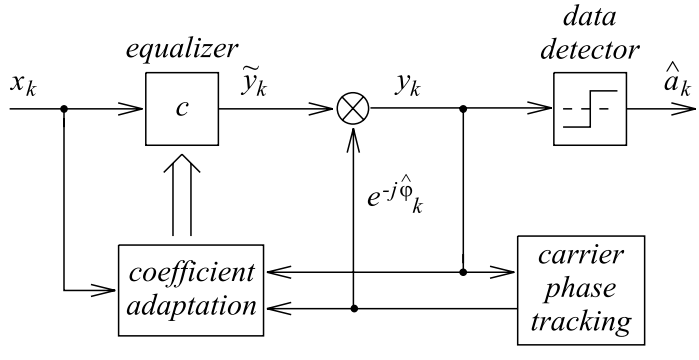


Figure 15.10. Block diagram of a self-training equalizer with the Sato algorithm for a QAM system.

and

$$\gamma_S = \frac{E[a_{k,I}^2]}{E[|a_{k,I}|]} = \frac{E[a_{k,Q}^2]}{E[|a_{k,Q}|]} \tag{15.58}$$

The gradient of (15.56) with respect to $\mathbf{c}$ yields (see also (8.380))

$$\nabla_{\mathbf{c}} J = \nabla_{Re[\mathbf{c}]} J + j\, \nabla_{Im[\mathbf{c}]} J = E[e^{j\hat{\varphi}_k}\, \mathbf{x}_k^* (\Theta(y_{k,I}) + j\, \Theta(y_{k,Q}))] \tag{15.59}$$

where

$$\Theta(v) = \frac{d}{dv}\, \Phi(v) = v - \gamma_S\, \mathrm{sgn}(v) \tag{15.60}$$

The partial derivative with respect to the carrier phase estimate is given by (see also (8.385))

$$\frac{\partial}{\partial \hat{\varphi}}\, J = E[Im(y_k (\Theta(y_{k,I}) + j\, \Theta(y_{k,Q}))^*)] \tag{15.61}$$

Defining the Sato pseudo error for QAM systems as

$$\epsilon_{S,k} = y_k - \gamma_S\, \mathrm{sgn}(y_k) \tag{15.62}$$

and observing that

$$Im(y_k\, \epsilon_{S,k}^*) = -Im(\gamma_S\, y_k\, \mathrm{sgn}(y_k^*)) \tag{15.63}$$

the equalizer coefficient updating and carrier phase estimate are given by

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu\, \epsilon_{S,k}\, e^{j\hat{\varphi}_k}\, \mathbf{x}_k^* \tag{15.64}$$

$$\hat{\varphi}_{k+1} = \hat{\varphi}_k + \mu_\varphi\, Im[\gamma_S\, y_k\, \mathrm{sgn}(y_k^*)] \tag{15.65}$$

where μ_φ is the adaptation gain of the carrier phase tracking loop. The equations (15.64) and (15.65) are analogous to (8.382) and (8.387) for the decision-directed case.

The same observations made for self-training equalization of PAM systems using the Sato algorithm hold for QAM systems. In particular, assuming that the algorithm converges to a point of global minimum of the cost function, we recall that the variance of the pseudo error assumes high values in the neighborhood of the point of convergence. Therefore in the steady state it is necessary to adopt a decision directed algorithm. To obtain smooth transitions between the self-training mode and the decision-directed mode without the need of a further control of the distortion level, in QAM systems we can use extensions of the Benveniste–Goursat and stop-and-go algorithms considered for self-training of PAM systems.

15.5.1 Constant modulus algorithm

In the constant modulus algorithm (CMA) for QAM systems proposed by Godard [15], self-training equalization is based on the cost function

$$J = E[(|\tilde{y}_k|^p - R_p)^2] = E[(|y_k|^p - R_p)^2] \tag{15.66}$$

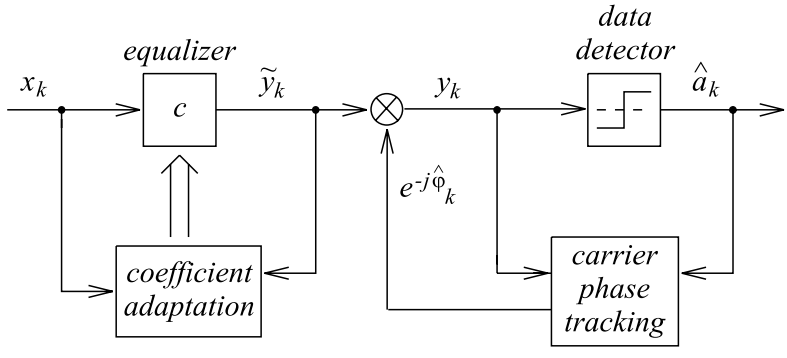


Figure 15.11. Block diagram of a self-training equalizer using the CMA.

where p is a parameter that usually assumes the value $p = 1$ or $p = 2$. We note that, as J depends on the absolute value of the equalizer output raised to the p-th power, the CMA does not require the knowledge of the carrier phase estimate. Figure 15.11 shows the block diagram of a receiver using the CMA.

The gradient of (15.66) is given by

$$\nabla_{\mathbf{c}} J = 2p\, E[(|\tilde{y}_k|^p - R_p)|\tilde{y}_k|^{p-2}\, \tilde{y}_k\, \mathbf{x}_k^*] \tag{15.67}$$

The constant R_p is chosen so that the gradient is equal to zero for a perfectly equalized system; therefore we have

$$R_p = \frac{E[|a_k|^{2p}]}{E[|a_k|^p]} \tag{15.68}$$

For example, for a 64-QAM constellation, we obtain $R_1 = 6.9$ and $R_2 = 58$. The 64-QAM constellation and the circle of radius $R_1 = 6.9$ are illustrated in Figure 15.12.

By using (15.67), we obtain the equalizer coefficient updating law

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu(|\tilde{y}_k|^p - R_p)|\tilde{y}_k|^{p-2}\, \tilde{y}_k\, \mathbf{x}_k^* \tag{15.69}$$

For $p = 1$, (15.69) becomes

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu(|\tilde{y}_k| - R_1)\frac{\tilde{y}_k}{|\tilde{y}_k|}\, \mathbf{x}_k^* \tag{15.70}$$

We note that the Sato algorithm, introduced in Section 15.2, can then be viewed as a particular case of the CMA.

The contour algorithm for QAM systems

Let us consider a receiver in which the received signal is sampled at the symbol rate and the carrier phase recovery is ideally carried out before the equalizer. The scheme of Figure 15.1

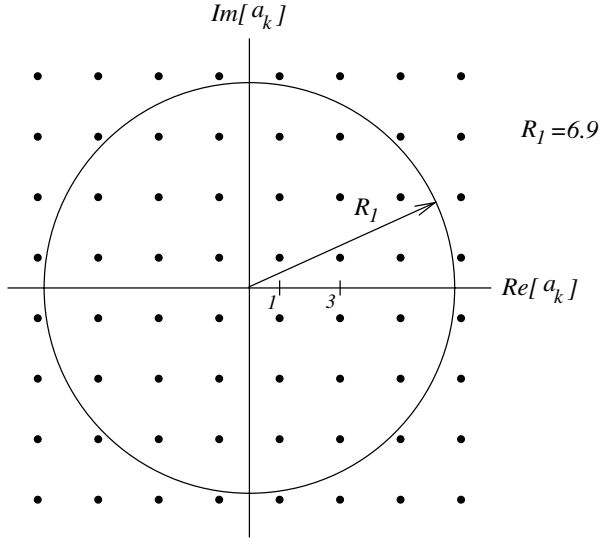


Figure 15.12. The 64-QAM constellation and the circle of radius $R_1 = 6.9$.

is still valid, and the complex-valued baseband equivalent channel output is given by

$$x_k = \sum_{i=-\infty}^{+\infty} h_i \, a_{k-i} + w_k \tag{15.71}$$

The equalizer output is expressed as $y_k = \mathbf{c}^T \mathbf{x}_k$. To generalize the notion of pseudo error of the contour algorithm introduced in Section 15.3 for PAM systems, we define a contour line $\mathcal{C}$ that connects the outer points of the constellation. For simplicity, we assume a square constellation with $L \times L$ points, as illustrated in Figure 15.13 for the case $L = 8$.

Let $\mathcal{S}$ be the region of the complex plane enclosed by the contour line $\mathcal{C}$ and let $\mathcal{C} \notin \mathcal{S}$ by definition. We denote by $y_k^{\mathcal{C}}$ the closest point to y_k on $\mathcal{C}$ every time that $y_k \notin \mathcal{S}$, that is every time the point y_k is found outside the region enclosed by $\mathcal{C}$. The pseudo error (15.37) is now extended as follows:

$$\epsilon_{CA,k} = \begin{cases} y_k - y_k^{\mathcal{C}} & & \text{se } y_k \notin \mathcal{S} \\ \left.\begin{array}{ll} -\delta_k \operatorname{sgn}(y_{k,I}) & \text{if } |y_{k,I}| \geq |y_{k,Q}| \\ -j\,\delta_k \operatorname{sgn}(y_{k,Q}) & \text{if } |y_{k,I}| < |y_{k,Q}| \end{array}\right\} & \text{se } y_k \in \mathcal{S} \end{cases} \tag{15.72}$$

Also in this case δ_k is a non-negative parameter, updated at each iteration as

$$\delta_{k+1} = \begin{cases} \delta_k - p_{\mathcal{S}}\Delta & \text{if } y_k \notin \mathcal{S} \\ \delta_k + (1 - p_{\mathcal{S}})\,\Delta & \text{if } y_k \in \mathcal{S} \end{cases} \tag{15.73}$$

where, by analogy with (15.40), the probability $p_{\mathcal{S}} \simeq P[y_k \in \mathcal{S}]$ is computed assuming that y_k is an equalized signal in the presence of additive noise.

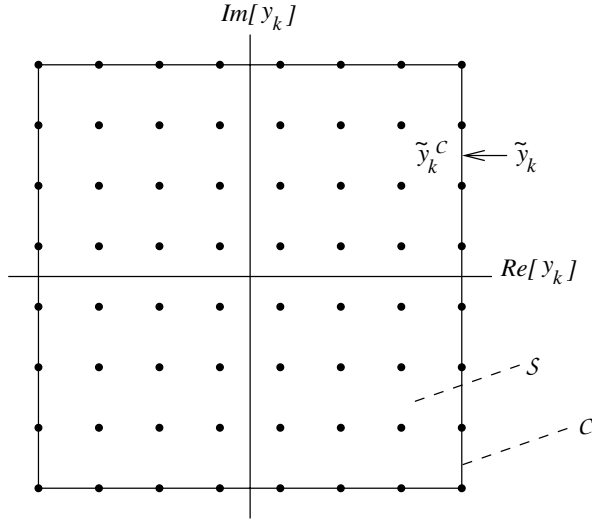


Figure 15.13. Illustration of the contour line and surface $\mathcal{S}$ for a 64-QAM constellation.

Let α_{max} be the maximum absolute value of the real and imaginary parts of the square $L \times L$ symbol constellation. If $|y_{k,I}| \geq \alpha_{max}$ or $|y_{k,Q}| \geq \alpha_{max}$, but not both, the projection of the sample y_k on the contour line $\mathcal{C}$ yields a non-zero pseudo error along one dimension and a zero error in the other dimension. If both $|y_{k,I}|$ and $|y_{k,Q}|$ are larger than α_{max}, $y_k^{\mathcal{C}}$ is chosen as the corner point of the constellation closest to y_k; in this case we obtain a non-zero pseudo error in both dimensions.

Thus the equalizer coefficients are updated according to the algorithm

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu \, \epsilon_{CA,k} \, \mathbf{x}_k^* \tag{15.74}$$

Clearly, the contour algorithm can also be applied to systems that use non-square constellations. In any case, the robust algorithm for carrier phase tracking that is described in the next section requires that the shape of the constellation is non-circular.

Joint contour algorithm and carrier phase tracking

We now apply the idea of generating an error signal with respect to the contour line of a constellation to the problem of carrier phase recovery and frequency offset compensation [12]. With reference to the scheme of Figure 15.10, we denote as $\hat{\varphi}_k$ the carrier phase estimate used for the received signal demodulation at instant k. If carrier recovery follows equalization, the complex equalizer output signal is given by

$$y_k = \tilde{y}_k \, e^{-j\hat{\varphi}_k} \tag{15.75}$$

As for equalizer coefficient updating, reliable information for updating the carrier phase estimate $\hat{\varphi}_k$ is only available if y_k falls outside of the region $\mathcal{S}$. As illustrated in Figure 15.14,

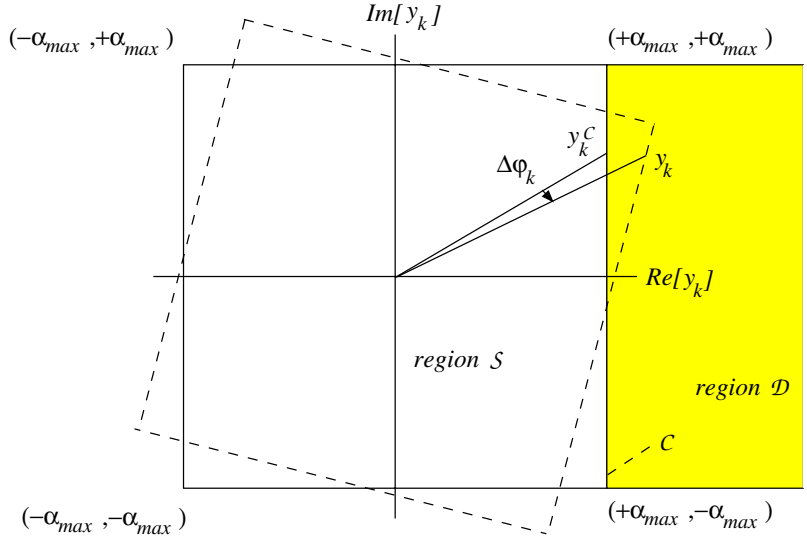


Figure 15.14. Illustration of the rotation of the symbol constellation in the presence of a phase error, and definition of $\Delta\varphi_k$.

the phase estimation error can then be computed as (see also (15.65))

$$\Delta\varphi_k \simeq Im(y_k\, y_k^{\mathcal{C}*}) = -Im[y_k(y_k^* - y_k^{\mathcal{C}*})] \tag{15.76}$$

If y_k falls within $\mathcal{S}$ the phase error is set to zero, that is $\Delta\varphi_k = 0$.

From Figure 15.14 we note that the phase error $\Delta\varphi_k$ is invariant with respect to a rotation of y_k equal to an integer multiple of $\pi/2$. Then to determine $\Delta\varphi_k$ we can first rotate y_k,

$$y'_k = y_k\, e^{j\ell\pi/2} \tag{15.77}$$

where ℓ is chosen such that $Re[y'_n] > |Im[y'_n]|$ (shaded region in Figure 15.14). Furthermore, we observe that the information on the phase error obtained by samples of the sequence $\{y_k\}$ that fall in the corner regions, where $|y_{k,I}| > \alpha_{max}$ and $|y_{k,Q}| > \alpha_{max}$, is not important. Thus we calculate a phase error only if y_k is outside of $\mathcal{S}$, but not in the corner regions, that is if $y'_k \in \mathcal{D}$, with $\mathcal{D} = \{y'_k : Re[y'_k] > \alpha_{max}, |Im[y'_k]| < \alpha_{max}\}$. Then (15.76) becomes

$$\Delta\varphi_k = \begin{cases} -Im[y'_k][Re[y'_k] - \alpha_{max}] & \text{if } y'_k \in \mathcal{D} \\ 0 & \text{otherwise} \end{cases} \tag{15.78}$$

In the presence of a frequency offset equal to $\Omega/(2\pi)$, the probability distribution of the samples $\{y_k\}$ rotates at a rate of $\Omega/(2\pi)$ revolutions per second. For large values of $\Omega/(2\pi)$, the phase error $\Delta\varphi_k$ does not provide sufficient information for the carrier phase tracking system to achieve a lock condition; therefore the update of $\hat{\varphi}_k$ must be made by a second-order PLL, where in the update of the second-order term a factor that is related to the value of the frequency offset must be included (see Section 14.7).

The needed information is obtained observing the statistical behavior of the term $Im[y'_k]$, conditioned by the event $y'_k \in \mathcal{D}$. At instants in which the sampling distribution of y_k is aligned with $\mathcal{S}$, the distribution of $Im[y'_k]$ is uniform in the range $[-\alpha_{max}, \alpha_{max}]$. Between these instants, the distribution of $Im[y'_k]$ exhibits a time varying behavior with a downward or upward trend depending on the sign of the frequency offset, with a minimum variance when the corners of the rotating probability distribution of y_k, which we recall rotates at a rate of $\Omega/(2\pi)$ revolutions per second, cross the coordinate axes. Defining

$$Q(v) = \begin{cases} v & \text{if } |v| < \alpha_{max} \\ 0 & \text{otherwise} \end{cases} \tag{15.79}$$

from the observation of Figure 15.14 a simple method to extract information on $\Omega/(2\pi)$ consists in evaluating

$$\Delta\, Im[y'_k] = Q\{Im[y'_k] - Im[y'_m]\} \quad y'_k \in \mathcal{D} \tag{15.80}$$

where $m < k$ denotes the last time index for which $y'_k \in \mathcal{D}$. In the mean, $\Delta\, Im[y'_k]$ exhibits the sign of the frequency offset.

The equations for the updating of the parameters of a second-order phase-locked loop for the carrier phase recovery then become

$$\begin{cases} \hat{\varphi}_{k+1} = \hat{\varphi}_k + \mu_\varphi\, \Delta\varphi_k + \Delta\hat{\varphi}_{c,k} \\ \Delta\hat{\varphi}_{c,k+1} = \Delta\hat{\varphi}_{c,k} + \mu_{f_1}\, \Delta\varphi_k + \mu_{f_2}\, \Delta\, Im[y'_k] \end{cases} \quad \text{if } y'_k \in \mathcal{D}$$
$$\begin{cases} \hat{\varphi}_{k+1} = \hat{\varphi}_k \\ \Delta\hat{\varphi}_{c,k+1} = \Delta\hat{\varphi}_{c,k} \end{cases} \quad \text{otherwise} \tag{15.81}$$

where μ_φ, μ_{f_1} and μ_{f_2} are suitable adaptation gains; typically, μ_φ is in the range $10^{-4} - 10^{-3}$, $\mu_{f_1} = (1/4)\mu_\varphi^2$, and $\mu_{f_2} \simeq \mu_{f_1}$.

The rotation of y_k given by (15.77) to obtain y'_k also has the advantage of simplifying the error computation for self-training equalizer coefficient adaptation with the contour algorithm.

With no significant effect on performance, we can introduce a simplification similar to that adopted to update the carrier phase, and let the pseudo error equal zero if y_k is found in the corner regions, that is $\epsilon_{CA,k} = 0$ if $Im[y'_k] > \alpha_{max}$. By using (15.72) and (15.73) to compute the pseudo error, the coefficient updating equation (15.74) becomes (see (8.382))

$$\mathbf{c}_{k+1} = \mathbf{c}_k - \mu\, \epsilon_{CA,k}\, e^{j\hat{\varphi}_k}\, \mathbf{x}_k^* \tag{15.82}$$

15.6 Examples of applications

In this section, we give examples of applications that illustrate the convergence behavior and steady state performance of self-training equalizers, with particular regard to the contour algorithm.

We initially consider self-training equalization for PAM transmission systems over unshielded twisted-pair UTP cables with frequency response given by (4.169).

Example 15.6.1

As a first example, we consider a 16-PAM system ($M = 16$) with a uniform probability distribution of the input symbols and symbol rate equal to 25 MBaud; the transmit and receive filters are designed to yield an overall raised cosine channel characteristic for a cable length of 50 m. In the simulations, the cable length is chosen equal to 100 m, and the received signal is disturbed by additive white Gaussian noise. The signal-to-noise ratio at the receiver input is equal to $\Gamma = 36$ dB. Self-training equalization is achieved by a fractionally spaced equalizer having $N = 32$ coefficients, and input signal sampled with sampling period equal to $T/2$. Figure 15.15 shows the convergence of the contour algorithms (15.37) and

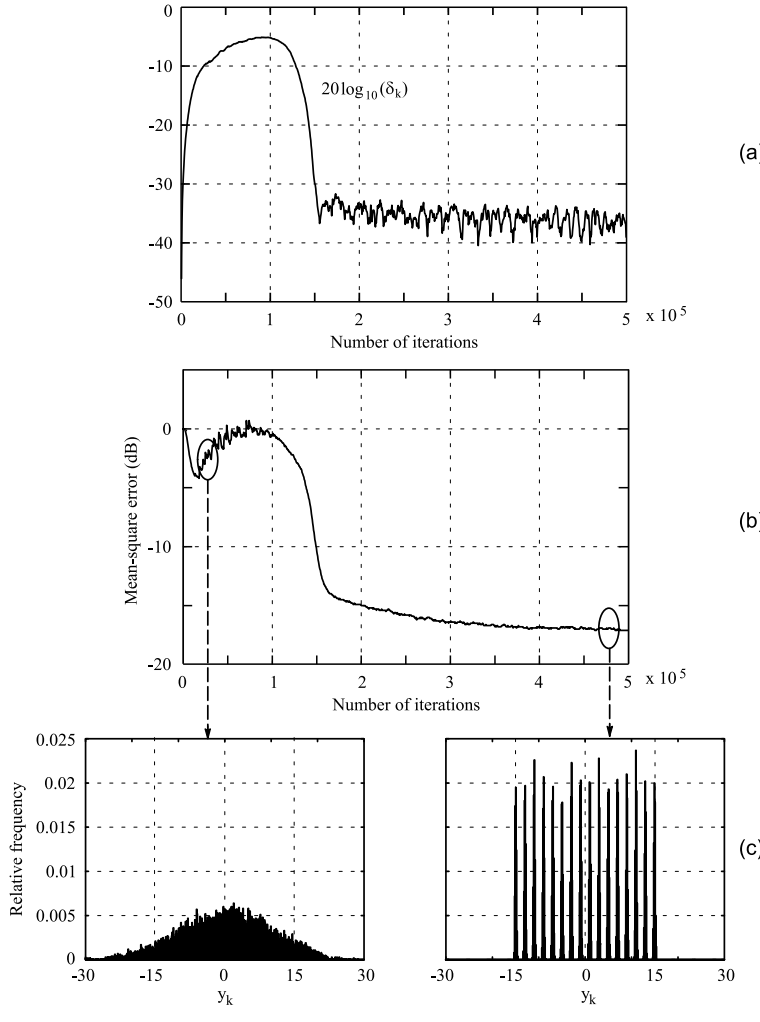


Figure 15.15. Illustration of the convergence of the contour algorithm for a 16-PAM system: (a) behavior of the parameter δ_n, (b) MSE convergence, (c) relative frequency of equalizer output samples at the beginning and the end of the convergence process.

(15.38) for $\delta_0 = 0$ and $\mathbf{c}_0$ chosen equal to the zero vector. The results are obtained for a cable with attenuation (4.148) equal to $\alpha(f)\,|_{f=1} = 3.85 \times 10^{-6}$ [$\mathrm{m}^{-1}\,\mathrm{Hz}^{-1/2}$], parameters of the self-training equalizer given by $\mu = 10^{-5}$ and $\Delta = 2.5 \times 10^{-4}$, and ideal timing recovery.

Example 15.6.2

We consider self-training equalization for a baseband quaternary partial response class IV system ($M = 4$) for transmission at 125 Mbit/s over UTP cables; a VLSI transceiver implementation for this system will be described in Chapter 19. We compare the performance of the contour algorithm, described in Section 15.3, with the Sato algorithm for partial response systems. Various realizations of the MSE convergence of a self-training equalizer with $N = 16$ coefficients are shown in Figures 15.16 and 15.17 for the Sato algorithm and the contour algorithm, respectively. The curves are parameterized by $t = \Delta T/T$, where $T = 16$ ns, and ΔT denotes the difference between the sampling phase of the channel output signal and the optimum sampling phase that yields the minimum MSE; we note that the contour algorithm has a faster convergence with respect to the Sato algorithm and yields significantly lower values of MSE in the steady state. The Sato algorithm can be applied only if timing recovery is achieved prior to equalization; note that the convergence characteristics of the contour algorithm makes self-training equalization possible even in the presence of considerable distortion and phase drift of the received signal.

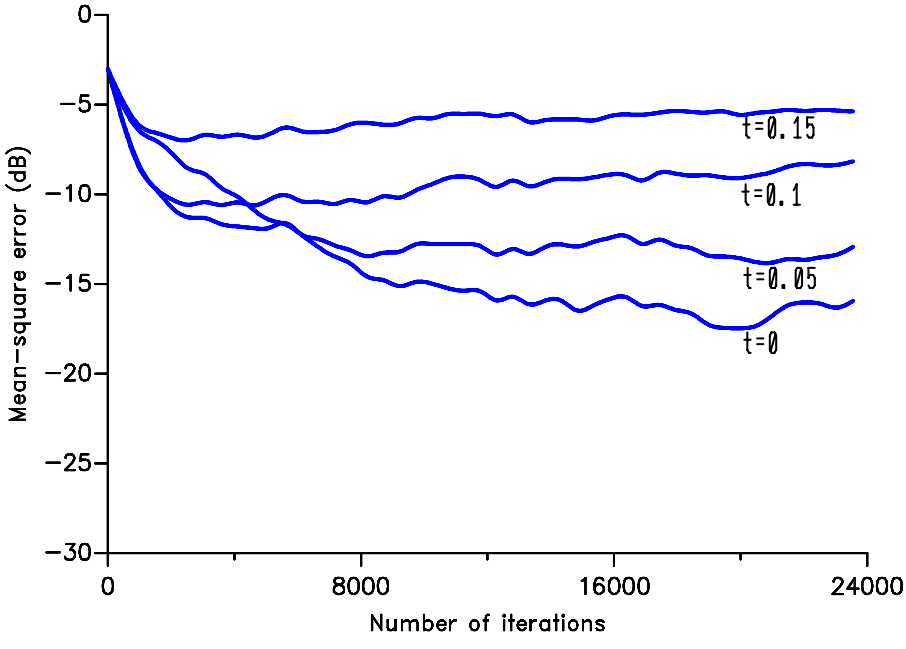


Figure 15.16. MSE convergence with the Sato algorithm for a QPR-IV system [13].

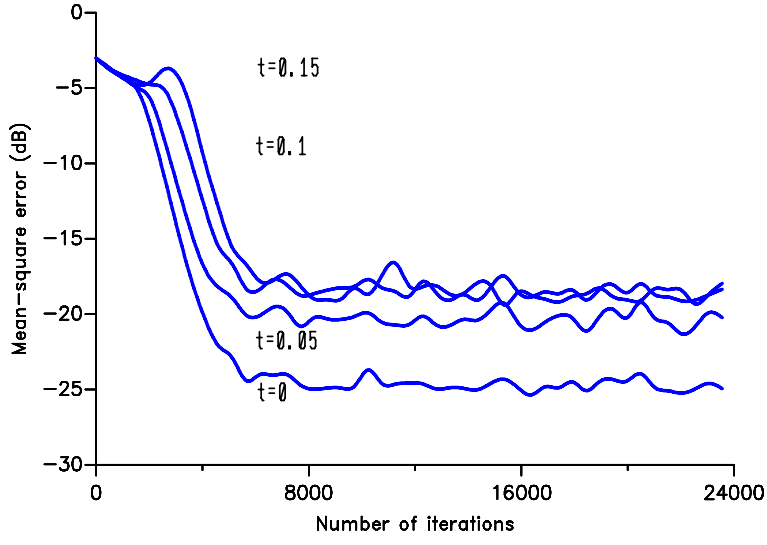


Figure 15.17. MSE convergence with the contour algorithm for a QPR-IV system [13].

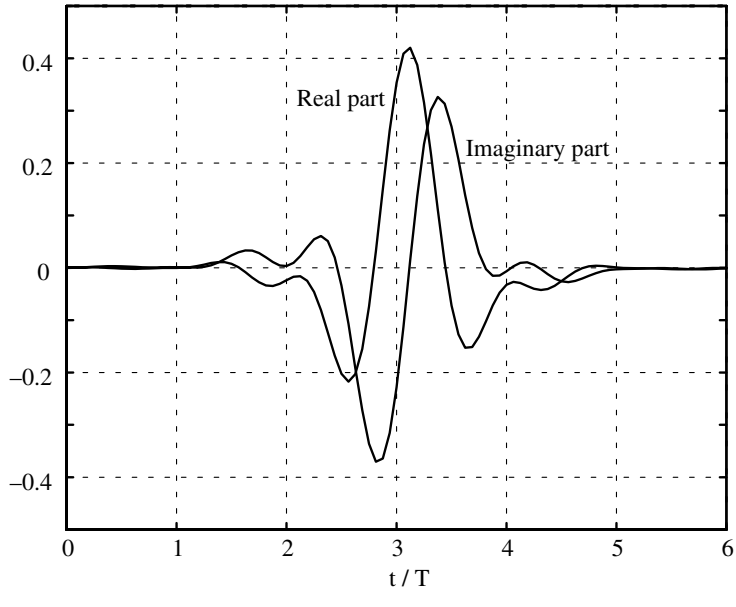


Figure 15.18. Overall baseband equivalent channel impulse response for simulations of a 256-QAM system. [*From [12]*, © *1998 Springer-Verlag London, Ltd.*]

Example 15.6.3
We now examine self-training equalization for a 256-QAM transmission system having a square constellation with $L = 16$ ($M = 256$), and symbol rate equal to 6 MBaud. Along each dimension, symbols ± 3, ± 1 have probability $2/20$, and symbols ± 15, ± 13, ± 11, ± 9, ± 7, and ± 5 have probability $1/20$. The overall baseband equivalent channel impulse response is illustrated in Figure 15.18. The received signal is disturbed by additive white Gaussian noise. The signal-to-noise ratio at the receiver input is equal to $\Gamma = 39$ dB. Signal equalization is obtained by a fractionally spaced equalizer having $N = 32$ coefficients, and input signal sampled with sampling period equal to $T/2$.

Figure 15.19 shows the convergence of the contour algorithm and the behavior of the parameter δ_k for $p_{\mathcal{S}} = 361/400$, various initial values of δ_0, and $\mathbf{c}_0$ given by a vector with all elements equal to zero except for one element. Results are obtained for $\mu = 10^{-4}$, $\Delta = 10^{-4}$, and ideal timing and carrier phase recovery.

Example 15.6.4
With reference to the previous example, we examine the behavior of the carrier phase recovery algorithm, assuming ideal timing recovery. Figure 15.20 illustrates the behavior of the MSE and of the second-order term $\Delta\hat{\varphi}_{c,k}$ for an initial frequency offset of $+2.5$ kHz, $\mu_\varphi = 4 \times 10^{-4}$, $\mu_{f_1} = 8 \times 10^{-8}$, and $\mu_{f_2} = 2 \times 10^{-8}$.

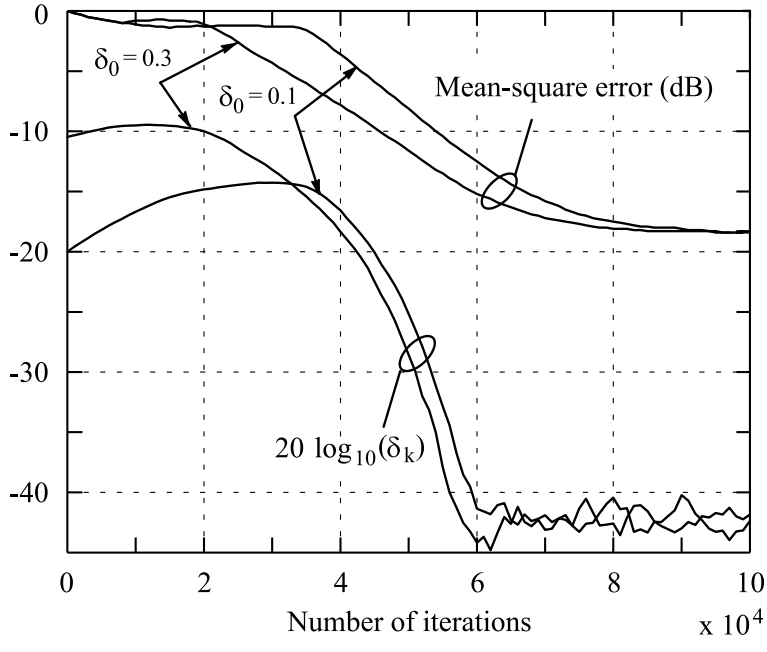


Figure 15.19. Convergence behavior of MSE and parameter δ_k using the contour algorithm for a 256-QAM system with non-uniform distribution of input symbols. [*From [12], © 1998 Springer-Verlag London, Ltd.*]

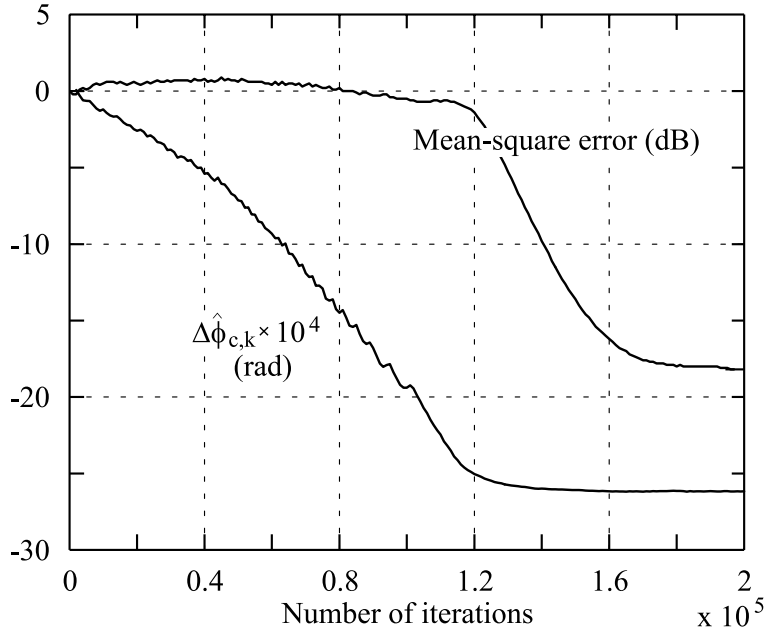


Figure 15.20. Illustration of the convergence behavior of MSE and second-order term $\Delta\hat{\varphi}_{c,k}$ using the contour algorithm in the presence of an initial frequency offset equal to 500 ppm for a 256-QAM system with non-uniform distribution of input symbols. [*From [12], © 1998 Springer-Verlag London, Ltd.*]

Bibliography

[1] H. Ichikawa, J. Sango, and T. Murase, "256 QAM multicarrier 400 Mb/s microwave radio system field tests", in *Proc. 1987 IEEE Int. Conference on Communications*, pp. 1803–1808, 1987.

[2] F. J. Ross and D. P. Taylor, "An enhancement to blind equalization algorithms", *IEEE Trans. on Communications*, vol. 39, pp. 636–639, May 1991.

[3] J. G. Proakis and C. L. Nikias, "Blind equalization", in *Proc. SPIE Adaptive Signal Processing*, vol. 1565, pp. 76–87, July 22–24 1991.

[4] S. Bellini, "Blind equalization and deconvolution", in *Proc. SPIE Adaptive Signal Processing*, vol. 1565, pp. 88–101, July 22–24 1991.

[5] N. Benvenuto and T. W. Goeddel, "Classification of voiceband data signals using the constellation magnitude", *IEEE Trans. on Communications*, vol. 43, pp. 2759–2770, Nov. 1995.

[6] R. Liu and L. Tong, eds, "Special issue on blind systems identification and estimation", *IEEE Proceedings*, vol. 86, Oct. 1998.

[7] Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation systems", *IEEE Trans. on Communications*, vol. 23, pp. 679–682, June 1975.

[8] L. Tong, G. Xu, B. Hassibi, and T. Kailath, "Blind channel identification based on second-order statistics: a frequency-domain approach", *IEEE Trans. on Information Theory*, vol. 41, pp. 329–334, Jan. 1995.

[9] A. Benveniste, M. Goursat, and G. Ruget, "Robust identification of a nonminimum phase system: blind adjustment of a linear equalizer in data communications", *IEEE Trans. on Automatic Control*, vol. 25, pp. 385–399, June 1980.

[10] A. Benveniste and M. Goursat, "Blind equalizers", *IEEE Trans. on Communications*, vol. 32, pp. 871–883, Aug. 1984.

[11] G. Picchi and G. Prati, "Blind equalization and carrier recovery using a 'Stop-and-Go' decision directed algorithm", *IEEE Trans. on Communications*, vol. 35, pp. 877–887, Sept. 1987.

[12] G. Cherubini, S. Ölçer, and G. Ungerboeck, "The contour algorithm for self-training equalization", in *Broadband Wireless Communications, 9th Tyrrhenian Int. Workshop on Digital Communications* (M. Luise and S. Pupolin, eds), Lerici, Italy, pp. 58–69, Sept. 7–10 1997. Berlin: Springer-Verlag, 1998.

[13] G. Cherubini, S. Ölçer, and G. Ungerboeck, "Self-training adaptive equalization for multilevel partial-response transmission systems", in *Proc. 1995 IEEE Int. Symposium on Information Theory*, Whistler, Canada, p. 401, Sept. 17–22 1995.

[14] G. Cherubini, "Nonlinear self-training adaptive equalization for partial-response systems", *IEEE Trans. on Communications*, vol. 42, pp. 367–376, February/March/April 1994.

[15] D. N. Godard, "Self recovering equalization and carrier tracking in two-dimensional data communication systems", *IEEE Trans. on Communications*, vol. 28, pp. 1867–1875, Nov. 1980.

Appendix 15.A *On the convergence of the contour algorithm*

Given $y = y_0 = \sum_{i=-\infty}^{+\infty} c_i\, x_{-i}$, we show that the only minima of the cost function $J = E[\Phi(y)]$ correspond to the equalizer settings $C = \pm H^{-1}$, for which we obtain $\Psi = \pm I$, except for a possible delay. If the systems H and C have finite energy, then also Ψ has finite energy and $E[\Phi(y)]$ may be regarded as a functional $J(\Psi) = E[\Phi(y)]$, where y is expressed as $y = \sum_{i=-\infty}^{+\infty} \psi_i\, a_{-i}$; thus we must prove that the only minima of $\mathcal{V}(\Psi)$ are found at points $\Psi = \pm I$. We consider input symbols with a probability density function $p_{a_k}(\alpha)$ uniform in the interval $[-\alpha_{max}, \alpha_{max}]$.

For the analysis, we express the system Ψ as $\Psi = r\bar{\Psi}$, where $r \geq 0$, and $\bar{\Psi} \leftrightarrow \{\bar{\psi}_i\}$, with $\sum_i \bar{\psi}_i^2 = 1$, denotes the normalized overall system. We consider the cost function J as a functional $\mathcal{V}(\Psi) = E[\Phi(r\bar{y})]$, where $\bar{y} = \bar{y}_0 = \sum_{i=-\infty}^{+\infty} \bar{\psi}_i a_{-i}$ denotes the output of the system $\bar{\Psi}$, and Φ has derivative Θ given by (15.34). Let

$$\tilde{\Theta}(x) = \begin{cases} x - \alpha_{max}\,\mathrm{sgn}(x) & \text{if } |x| \geq \alpha_{max} \\ 0 & \text{otherwise} \end{cases} \tag{15.83}$$

and $p_{\bar{y}_k}(x)$ denote the probability density function of $\bar{y}$.

We express the parameter γ_{CA} as

$$\begin{aligned} \gamma_{CA} &= \gamma_{\bar{\Psi}} + \gamma_r \\ \gamma_{\bar{\Psi}} &= \frac{\displaystyle\int b\,\tilde{\Theta}(b)\, p_{\bar{y}_k}(b)\, db}{\displaystyle\int |b|\, p_{\bar{y}_k}(b)\, db} \\ \gamma_r &= \begin{cases} 1 - r & \text{if } r \leq 1 \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{15.84}$$

Examine the function $\mathcal{V}(\bar{\Psi})$ on the unit sphere $\mathcal{S}$. To claim that the only minima of $\mathcal{V}(\bar{\Psi})$ are found at points $\Psi = \pm I$, we apply Theorem 3.5 of [9]. Consider a pair of indices (i, j), $i \neq j$, and a fixed system with coefficients $\{\bar{\psi}_\ell\}_{\ell \neq i,j}$, such that $R^2 = 1 - \sum_{\ell \neq i,j} \bar{\psi}_\ell^2 > 0$. For $\varphi \in [0, 2\pi)$, let $\bar{\Psi}_\varphi \in \mathcal{S}$ be the system with coefficients $\{\bar{\psi}_\ell\}_{\ell \neq i,j}$, $\bar{\psi}_i = R\cos\varphi$, and $\bar{\psi}_j = R\sin\varphi$; moreover, let $(\partial/\partial\varphi)\mathcal{V}(\bar{\Psi}_\varphi)$ be the derivative of $\mathcal{V}(\bar{\Psi}_\varphi)$ with respect to φ at point $\bar{\Psi} = \bar{\Psi}_\varphi$. As $p_{a_k}(\alpha)$ is sub-Gaussian, it can be shown that

$$\frac{\partial}{\partial\varphi}\,\mathcal{V}(\bar{\Psi}_\varphi) = 0 \qquad \text{for } \varphi = k\,\frac{\pi}{4} \qquad k \in \mathbf{Z} \tag{15.85}$$

and

$$\frac{\partial}{\partial\varphi}\,\mathcal{V}(\bar{\Psi}_\varphi) > 0 \qquad \text{for } 0 < \varphi < \frac{\pi}{4} \tag{15.86}$$

From the above equations we have that the stationary points of $\mathcal{V}(\bar{\Psi}_\varphi)$ correspond to systems characterized by the property that all non-zero coefficients have the same absolute value. Furthermore, using symmetries of the problem, we find the only minima are at $\pm I$, except for a possible delay, and the other stationary points of $\mathcal{V}(\bar{\Psi}_\varphi)$ are saddle points.

The study of the functional $\mathcal{V}$ is then extended to the entire parameter space. As the results obtained for the restriction of $\mathcal{V}$ to $\mathcal{S}$ are also valid on a sphere of arbitrary radius r, we need to study only the radial derivatives of $\mathcal{V}$. For this reason, we consider the function $\tilde{\mathcal{V}}(r) = \mathcal{V}(r\bar{\Psi})$, whose first and second derivatives are

$$\tilde{\mathcal{V}}'(r) = \int b\,\tilde{\Theta}(rb)\,p_{\bar{y}_k}(b)\,db - (\gamma_{\bar{\Psi}} + \gamma_r)\int |b|\,p_{\bar{y}_k}(b)\,db \tag{15.87}$$

and

$$\tilde{\mathcal{V}}''(r) = \int b^2\,\tilde{\Theta}'(rb)\,p_{\bar{y}_k}(b)\,db - \gamma_r'\int |b|\,p_{\bar{y}_k}(b)\,db \tag{15.88}$$

where $\tilde{\Theta}'$ and γ_r' denote derivatives.

Recalling the expressions of $\gamma_{\bar{\Psi}}$ and γ_r given by (15.84), we obtain $\tilde{\mathcal{V}}'(0) < 0$ and $\tilde{\mathcal{V}}''(r) > 0$. Therefore there exists a radius r_0 such that the radial component of the gradient is negative for $r < r_0$ and positive for $r > r_0$. For a fixed point $\bar{\Psi} \in \mathcal{S}$, r_0 is given by the solution of the equation

$$\int b\,\tilde{\Theta}(rb)\,p_{\bar{y}_k}(b)\,db - (\gamma_{\bar{\Psi}} + \gamma_r)\int |b|\,p_{\bar{y}_k}(b)\,db = 0 \tag{15.89}$$

Substituting the expressions of $\gamma_{\bar{\Psi}}$ and γ_r in (15.89), we obtain $r_0 = 1$, $\forall \bar{\Psi} \in \mathcal{S}$. Therefore the only minima of $\mathcal{V}$ are at $\pm I$. Furthermore, as the radial component of the gradient vanishes on $\mathcal{S}$, the steepest descent lines of $\mathcal{V}$ do not cross the unit sphere. Using the same argument given in [9], we conclude that the points $\pm I$ are the only "stable attractors" of the steepest descent lines of the function $\mathcal{V}$, and that the unique stable attractors of the steepest descent lines of J are $\pm H^{-1}$.

Note that the parameter $\gamma_{\bar{\Psi}}$ is related to the distortion of the distribution of the input sequence filtered by a normalized system; this parameter varies along the trajectories of the stochastic gradient algorithm and vanishes as a point of minimum is reached. Moreover, note that the parameter γ_r indicates the deviation of the overall system gain from the desired unit value; if the gain is too small, the gradient is augmented with an additional driving term.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 16

Applications of interference cancellation

The algorithms and structures discussed in this chapter can be applied to both wired and wireless systems, even though transmission systems over twisted-pair cables will be considered to describe examples of applications. Full-duplex data transmission over a single twisted-pair cable permits the simultaneous flow of information in two directions using the same frequency band. Examples of applications of this technique are found in digital communications systems that operate over the telephone network. In a digital subscriber loop, at each end of the full-duplex link, a circuit called hybrid separates the two directions of transmission. To avoid signal reflections at the near and far-end hybrid, a precise knowledge of the line impedance would be required. As the line impedance depends on line parameters that, in general, are not exactly known, an attenuated and distorted replica of the transmit signal leaks to the receiver input as an echo signal. Data-driven adaptive echo cancellation mitigates the effects of impedance mismatch.

A similar problem is caused by cross-talk in transmission systems over voice-grade unshielded twisted-pair cables for local-area network applications, where multipair cables are used to physically separate the two directions of transmission. Cross-talk is a statistical phenomenon due to randomly varying differential capacitive and inductive coupling between adjacent two-wire transmission lines (see Section 4.4.2). At the rates of several megabit-per-second that are usually considered for local-area network applications, *near–end cross-talk* (NEXT) represents the dominant disturbance; hence adaptive NEXT cancellation must be performed to ensure reliable communications.

A different problem shows up in frequency–division duplexing transmission, where different frequency bands are used in the two directions of transmission. In such a case *far–end cross-talk* (FEXT) interference is dominant. This situation occurs, for instance, in *very–high–speed digital subscriber line* (VDSL) systems, where multiple users are connected to a central station via unshielded twisted-pairs located in the same cable binder.

In voiceband data modems the model for the echo channel is considerably different from the echo channel model adopted for baseband transmission. The transmitted signal is a passband QAM signal, and the far-end echo may exhibit significant carrier-phase jitter and carrier-frequency shift, which are caused by signal processing at intermediate points in the telephone network. Therefore a digital adaptive echo canceller for voiceband modems needs to embody algorithms that account for the presence of such additional impairments.

In the first three sections of this chapter,[1] we describe the echo channel models and adaptive echo canceller structures for various digital communications systems, which are classified according to the employed modulation techniques. We also address the trade-offs between complexity, speed of adaptation, and accuracy of cancellation in adaptive echo cancellers. In the last section, we address the problem of FEXT interference cancellation for upstream transmission in a VDSL system. The system is modelled as a *multi–input multi–output* (MIMO) system, to which multi–user detection techniques (see Section 10.4) can be applied.

16.1 Echo and near–end cross-talk cancellation for PAM systems

The model of a full-duplex baseband PAM data transmission system employing adaptive echo cancellation is shown in Figure 16.1. To describe system operations, we consider one end of the full-duplex link. The configuration of an echo canceller for a PAM transmission system (see Section 6.13) is shown in Figure 16.2. The transmitted data consist of a sequence $\{a_k\}$ of i.i.d. real-valued symbols from the M-ary alphabet $\mathcal{A} = \{\pm 1, \pm 3, \ldots, \pm(M-1)\}$. The sequence $\{a_k\}$ is converted into an analog signal by a digital-to-analog (D/A) converter (see Chapter 7). The conversion to a staircase signal by a zero-order hold D/A converter is described by the frequency response $H_{\mathrm{D/A}}(f) = T\sin(\pi f T)/(\pi f T)$, where T denotes the modulation interval. The D/A converter output is filtered by the analog transmit filter and is input to the channel through the hybrid.

The signal $x(t)$ at the output of the low-pass analog receive filter has three components, namely, the signal from the far-end transmitter $r(t)$, the echo $u(t)$, and additive Gaussian noise $w_R(t)$. The signal $x(t)$ is given by

$$x(t) = r(t) + u(t) + w_R(t) = \sum_{k=-\infty}^{+\infty} a_k^R\, h(t-kT) + \sum_{k=-\infty}^{+\infty} a_k\, h_E(t-kT) + w_R(t) \tag{16.1}$$

where $\{a_k^R\}$ is the sequence of symbols from the remote transmitter, and $h(t)$ and $h_E(t) = \{h_{\mathrm{D/A}} * g_E\}(t)$ are the impulse responses of the overall channel and of the echo channel,

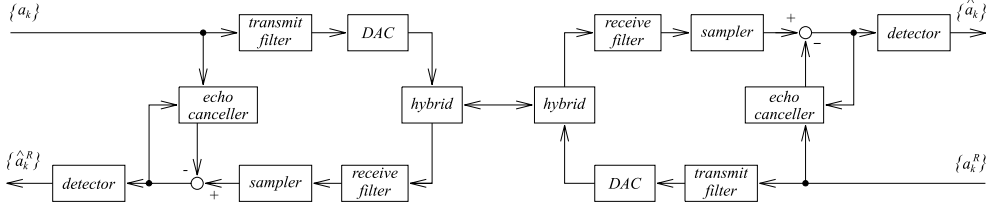


Figure 16.1. Model of a full-duplex PAM transmission system.

[1] The material presented in Sections 16.1–16.3 is reproduced with permission from *G. Cherubini, "Echo Cancellation," in The Mobile Communications Handbook (J. D. Gibson, ed.), Ch. 7, pp. 7.1–7.15, Boca Raton, FL: CRC Press, 1999, 2nd ed.* © *1999 CRC Press, Boca Raton, FL.*

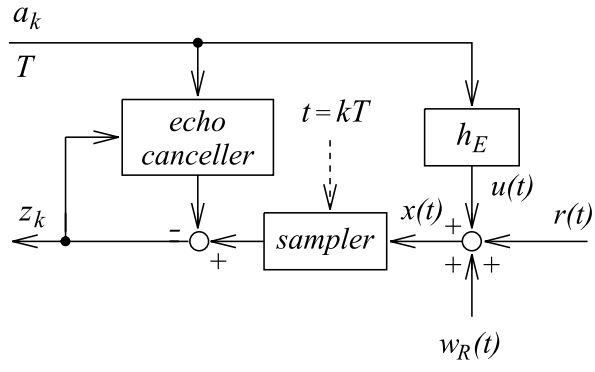


Figure 16.2. Configuration of an echo canceller for a PAM transmission system.

respectively. In the expression of $h_E(t)$, the function $h_{\mathrm{D/A}}(t)$ is the inverse Fourier transform of $H_{\mathrm{D/A}}(f)$. The signal obtained after echo cancellation is processed by a detector that outputs the sequence of detected symbols $\{\hat{a}_k^R\}$.

Cross-talk cancellation and full duplex transmission

In the case of full-duplex PAM data transmission over multi-pair cables for local-area network applications, where NEXT represents the main disturbance, the configuration of a digital NEXT canceller is also obtained as shown in Figure 16.2, with the echo channel replaced by the cross-talk channel. For these applications, however, instead of *mono-duplex* transmission, where one pair is used to transmit only in one direction and the other pair to transmit only in the reverse direction, *dual-duplex* transmission may be adopted. Bi-directional transmission at rate R_b over two pairs is then accomplished by full-duplex transmission of data streams at $R_b/2$ over each of the two pairs. The lower modulation rate and/or spectral efficiency required per pair for achieving an aggregate rate equal to R_b represents an advantage of dual-duplex over mono-duplex transmission. Dual-duplex transmission requires two transmitters and two receivers at each end of a link, as well as separation of the simultaneously transmitted and received signals on each pair, as illustrated in Figure 16.3. In dual-duplex transceivers it is therefore necessary to suppress echoes originated by reflections at the hybrids and at impedance discontinuities in the cable, as well as self NEXT, by adaptive digital echo and NEXT cancellation. Although a dual-duplex scheme might appear to require higher implementation complexity than a mono-duplex scheme, it turns out that the two schemes are equivalent in terms of the number of multiply-and-add operations per second that are needed to perform the various filtering operations.

One of the transceivers in a full-duplex link will usually employ an externally provided reference clock for its transmit and receive operations. The other transceiver will extract timing from the receive signal, and use this timing for its transmitter operations. This is known as *loop timing*, also illustrated in Figure 16.3. If signals were transmitted in opposite directions with independent clocks, signals received from the remote transmitter would generally shift in phase relative to the also received echo signals. To cope with this effect, some

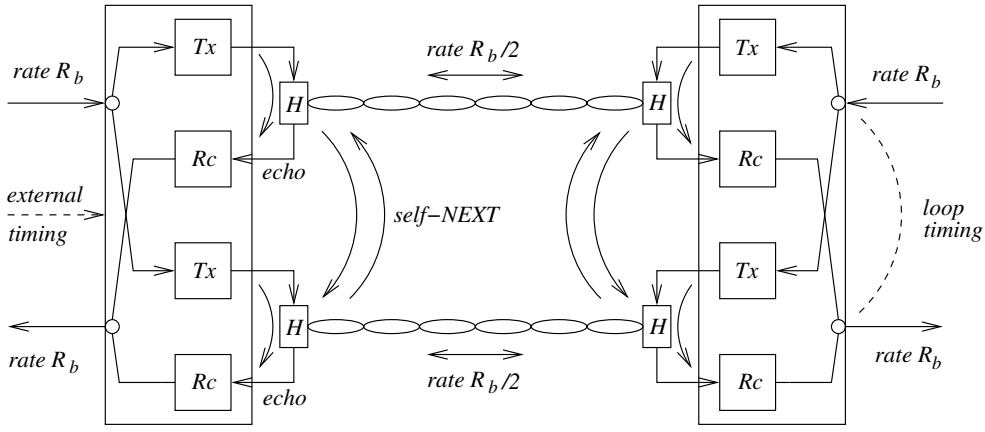


Figure 16.3. Model of a dual-duplex transmission system.

form of interpolation (see Chapter 14) would be required that can significantly increase the transceiver complexity.

Polyphase structure of the canceller

In general, we consider baseband signalling techniques such that the signal at the output of the overall channel has non-negligible excess bandwidth, i.e., non-negligible spectral components at frequencies larger than half of the modulation rate, $|f| \geq 1/(2T)$. Therefore, to avoid aliasing, the signal $x(t)$ is sampled at twice the modulation rate or at a higher sampling rate. Assuming a sampling rate equal to F_0/T, $F_0 > 1$, the i-th sample during the k-th modulation interval is given by

$$x\left((kF_0 + i)\frac{T}{F_0}\right) = x_{kF_0+i} = r_{kF_0+i} + u_{kF_0+i} + w_{kF_0+i} \qquad i = 0, \ldots, F_0 - 1$$

$$= \sum_{n=-\infty}^{+\infty} h_{nF_0+i}\, a^R_{k-n} + \sum_{n=-\infty}^{+\infty} h_{E,nF_0+i}\, a_{k-n} + w_{kF_0+i} \qquad (16.2)$$

where $\{h_{kF_0+i}, i = 0, \ldots, F_0 - 1\}$ and $\{h_{E,kF_0+i}, i = 0, \ldots, F_0 - 1\}$ are the discrete-time impulse responses of the overall channel and the echo channel, respectively, and $\{w_{kF_0+i}, i = 0, \ldots, F_0 - 1\}$ is a sequence of Gaussian noise samples with zero mean and variance σ_w^2. Equation (16.2) suggests that the sequence of samples $\{x_{kF_0+i}, i = 0, \ldots, F_0 - 1\}$ be regarded as a set of F_0 interlaced sequences, each with a sampling rate equal to the modulation rate. Similarly, the sequence of echo samples $\{u_{kF_0+i}, i = 0, \ldots, F_0 - 1\}$ can be regarded as a set of F_0 interlaced sequences that are output by F_0 independent echo channels with discrete-time impulse responses $\{h_{E,kF_0+i}\}, i = 0, \ldots, F_0 - 1$, and an identical sequence $\{a_k\}$ of input symbols [1]. Hence, echo cancellation can be performed by F_0 interlaced echo cancellers, as shown in Figure 16.4. As the performance of each

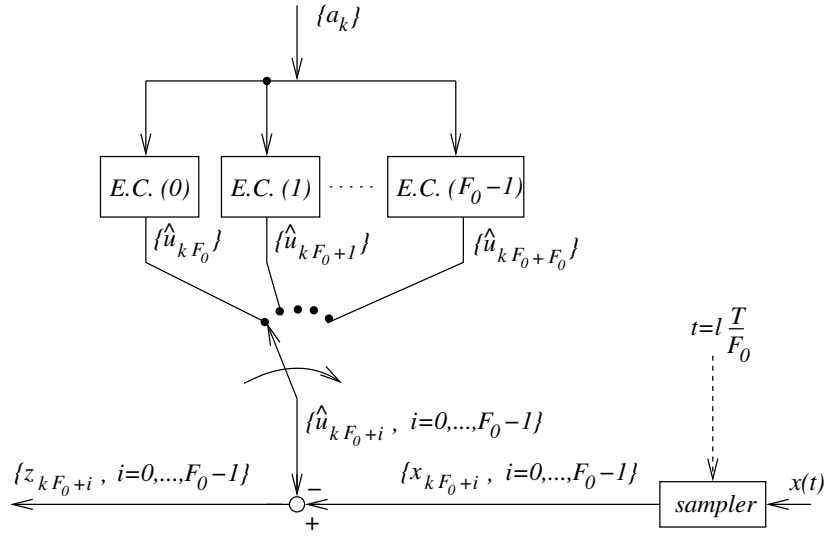


Figure 16.4. A set of F_0 interlaced echo cancellers.

canceller is independent of the other $F_0 - 1$ units, in the remaining part of this section we will consider the operations of a single echo canceller.

Canceller at symbol rate

The echo canceller generates an estimate $\hat{u}_k$ of the echo signal. If we consider a transversal filter implementation, $\hat{u}_k$ is obtained as the inner product of the vector of filter coefficients at time $t = kT$, $\mathbf{c}_k = [c_{0,k}, \ldots, c_{N-1,k}]^T$, and the vector of signals stored in the echo canceller delay line at the same instant, $\mathbf{a}_k = [a_k, \ldots, a_{k-N+1}]^T$, expressed by

$$\hat{u}_k = \mathbf{c}_k^T \, \mathbf{a}_k = \sum_{n=0}^{N-1} c_{n,k} \, a_{k-n} \tag{16.3}$$

The estimate of the echo is subtracted from the received signal. The result is defined as the cancellation error signal

$$z_k = x_k - \hat{u}_k = x_k - \mathbf{c}_k^T \, \mathbf{a}_k \tag{16.4}$$

The echo attenuation that must be provided by the echo canceller to achieve proper system operation depends on the application. For example, for the integrated services digital network (ISDN) U-Interface transceiver, the echo attenuation must be larger than 55 dB [2]. It is then required that the echo signals outside of the time span of the echo canceller delay line be negligible, i.e., $h_{E,n} \approx 0$ for $n < 0$ and $n > N - 1$. As a measure of system performance, we consider the mean-square error J_k at the output of the echo canceller at time $t = kT$, defined by

$$J_k = E[z_k^2] \tag{16.5}$$

For a particular coefficient vector $\mathbf{c}_k$, substitution of (16.4) into (16.5) yields (see (2.17))

$$J_k = E[x_k^2] - 2\mathbf{c}_k^T \mathbf{p} + \mathbf{c}_k^T \mathbf{R}\mathbf{c}_k \tag{16.6}$$

where $\mathbf{p} = E[x_k \mathbf{a}_k]$ and $\mathbf{R} = E[\mathbf{a}_k \mathbf{a}_k^T]$. With the assumption of i.i.d. transmitted symbols, the correlation matrix $\mathbf{R}$ is diagonal. The elements on the diagonal are equal to the variance of the transmitted symbols, $\sigma_a^2 = (M^2 - 1)/3$. From (2.40) the minimum mean-square error is given by

$$J_{min} = E[x_k^2] - \mathbf{c}_{opt}^T \mathbf{R}\mathbf{c}_{opt} \tag{16.7}$$

where the optimum coefficient vector is $\mathbf{c}_{opt} = \mathbf{R}^{-1}\mathbf{p}$.

We note that proper system operation is achieved only if the transmitted symbols are uncorrelated with the symbols from the remote transmitter. If this condition is satisfied, the optimum filter coefficients are given by the values of the discrete-time echo channel impulse response, i.e., $c_{n,opt} = h_{E,n}$, $n = 0, \ldots, N-1$.

Adaptive canceller

By the LMS algorithm, the coefficients of the echo canceller converge in the mean to $\mathbf{c}_{opt}$. The LMS algorithm (see Section 3.1.2) for an N-tap adaptive linear transversal filter is formulated as follows:

$$\mathbf{c}_{k+1} = \mathbf{c}_k + \mu\, z_k\, \mathbf{a}_k \tag{16.8}$$

where μ is the adaptation gain.

The block diagram of an adaptive transversal filter echo canceller is shown in Figure 16.5.

If we define the vector $\mathbf{\Delta c}_k = \mathbf{c}_k - \mathbf{c}_{opt}$, the mean-square error can be expressed by (2.40)

$$J_k = J_{min} + \mathbf{\Delta c}_k^T \mathbf{R}\, \mathbf{\Delta c}_k \tag{16.9}$$

where the term $\mathbf{\Delta c}_k^T \mathbf{R}\, \mathbf{\Delta c}_k$ represents an excess mean-square error due to the misadjustment of the filter settings. Under the assumption that the vectors $\mathbf{\Delta c}_k$ and $\mathbf{a}_k$ are statistically independent, the dynamics of the mean-square error are given by (see (3.272))

$$J_k = \sigma_0^2[1 - \mu\sigma_a^2(2 - \mu N\sigma_a^2)]^k + \frac{2J_{min}}{2 - \mu N\sigma_a^2} \tag{16.10}$$

where σ_0^2 is determined by the initial conditions. The mean-square error converges to a finite steady-state value J_∞ if the stability condition $0 < \mu < 2/(N\sigma_a^2)$ is satisfied. The optimum adaptation gain that yields fastest convergence at the beginning of the adaptation process is $\mu_{opt} = 1/(N\sigma_a^2)$. The corresponding time constant and asymptotic mean-square error are $\tau_{opt} = N$ and $J_\infty = 2J_{min}$, respectively.

We note that a fixed adaptation gain equal to μ_{opt} could not be adopted in practice, as after echo cancellation the signal from the remote transmitter would be embedded in a residual echo having approximately the same power. If the time constant of the convergence

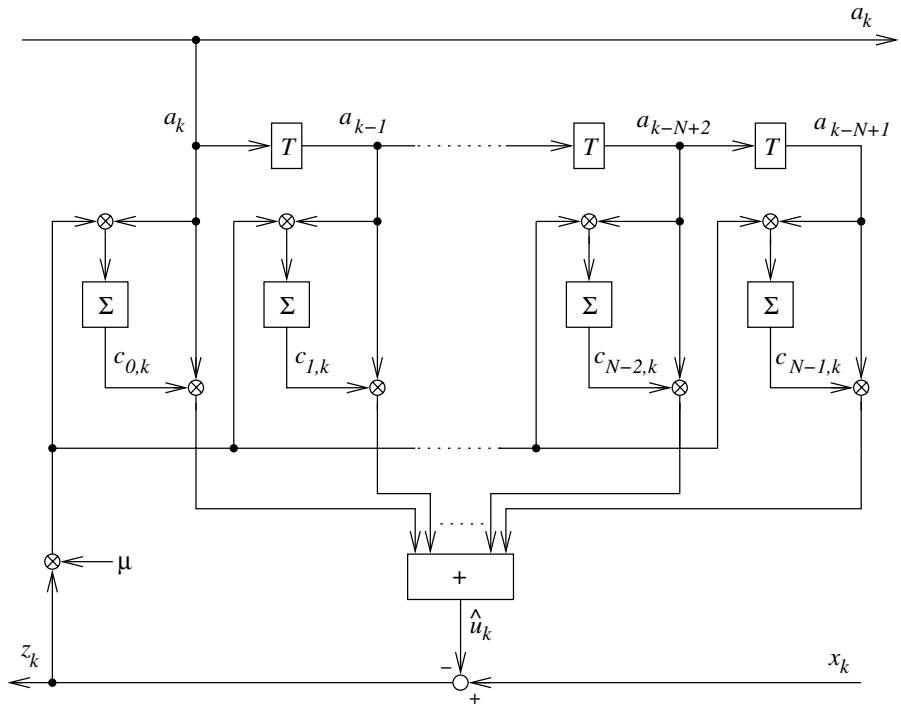


Figure 16.5. Block diagram of an adaptive transversal filter echo canceller.

mode is not a critical system parameter, an adaptation gain smaller than μ_{opt} will be adopted to achieve an asymptotic mean-square error close to J_{min}. On the other hand, if fast convergence is required, a variable adaptation gain will be chosen.

Several techniques have been proposed to increase the speed of convergence of the LMS algorithm. In particular, for echo cancellation in data transmission, the speed of adaptation is reduced by the presence of the signal from the remote transmitter in the cancellation error. To mitigate this problem, the data signal can be adaptively removed from the cancellation error by a decision-directed algorithm [3].

Modified versions of the LMS algorithm have been also proposed to reduce system complexity. For example, the sign algorithm suggests that only the sign of the error signal be used to compute an approximation of the gradient [4]. An alternative means to reduce the implementation complexity of an adaptive echo canceller consists in the choice of a filter structure with a lower computational complexity than the transversal filter.

Canceller structure with distributed arithmetic

At high rates, very large-scale integration (VLSI) technology is needed for the implementation of transceivers for full-duplex data transmission. High-speed echo cancellers and near-end cross-talk cancellers that do not require multiplications represent an attractive solution because of their low complexity. As an example of an architecture suitable for VLSI

implementation, we consider echo cancellation by a distributed-arithmetic filter, where multiplications are replaced by table look-up and shift-and-add operations [5]. By segmenting the echo canceller into filter sections of shorter lengths, various trade-offs concerning the number of operations per modulation interval and the number of memory locations needed to store the look-up tables are possible. Adaptivity is achieved by updating the values stored in the look-up tables by the LMS algorithm. To describe the principles of operations of a distributed-arithmetic echo canceller, we assume that the number of elements in the alphabet of the input symbols is a power of two, $M = 2^W$. Therefore, each symbol is represented by the vector $[a_k^{(0)}, \ldots, a_k^{(W-1)}]$, where $a_k^{(i)} \in \{0, 1\}$, $i = 0, \ldots, W-1$, are independent binary random variables, i.e.,

$$a_k = \sum_{w=0}^{W-1} (2a_k^{(w)} - 1)\, 2^w = \sum_{w=0}^{W-1} b_k^{(w)}\, 2^w \tag{16.11}$$

where $b_k^{(w)} = (2a_k^{(w)} - 1) \in \{-1, +1\}$.

By substituting (16.11) into (16.3) and segmenting the delay line of the echo canceller into L sections with $K = N/L$ delay elements each, we obtain

$$\hat{u}_k = \sum_{\ell=0}^{L-1} \sum_{w=0}^{W-1} 2^w \left[\sum_{m=0}^{K-1} b_{k-\ell K-m}^{(w)}\, c_{\ell K+m,k} \right] \tag{16.12}$$

Note that the summation within parenthesis in (16.12) may assume at most 2^K distinct real values, one for each binary sequence $\{a_{k-\ell K-m}^{(w)}\}$, $m = 0, \ldots, K-1$. If we precompute these 2^K values and store them in a look-up table addressed by the binary sequence, we can substitute the real time summation by a simple reading from the table.

Equation (16.12) suggests that the filter output can be computed using a set of $L2^K$ values that are stored in L tables with 2^K memory locations each. The binary vectors $\mathbf{a}_{k,\ell}^{(w)} = [a_{k-\ell K}^{(w)}, \ldots, a_{k-\ell K-K+1}^{(w)}]$, $w = 0, \ldots, W-1$, $\ell = 0, \ldots, L-1$, determine the addresses of the memory locations where the values that are needed to compute the filter output are stored. The filter output is obtained by WL table look-up and shift-and-add operations.

We observe that $\mathbf{a}_{k,\ell}^{(w)}$ and its binary complement $\bar{\mathbf{a}}_{k,\ell}^{(w)}$ select two values that differ only in their sign. This symmetry is exploited to halve the number of values to be stored. To determine the output of a distributed-arithmetic filter with reduced memory size, we reformulate (16.12) as

$$\hat{u}_k = \sum_{\ell=0}^{L-1} \sum_{w=0}^{W-1} 2^w\, b_{k-\ell K}^{(w)} \left[c_{\ell K,k} + b_{k-\ell K}^{(w)} \sum_{m=1}^{K-1} b_{k-\ell K-m}^{(w)}\, c_{\ell K+m,k} \right] \tag{16.13}$$

Then the binary symbol $b_{k-\ell K}^{(w)}$ determines whether a selected value is to be added or subtracted. Each table has now 2^{K-1} memory locations, and the filter output is given by

$$\hat{u}_k = \sum_{\ell=0}^{L-1} \sum_{w=0}^{W-1} 2^w\, b_{k-\ell K}^{(w)}\, d_k(i_{k,\ell}^{(w)}, \ell) \tag{16.14}$$

where $d_k(n, \ell)$, $n = 0, \ldots, 2^{K-1} - 1$, are the look-up values stored in the ℓ-th table, $\ell = 0, \ldots, L-1$, whose values are

$$d_k(n, \ell) = c_{\ell K, k} + b^{(w)}_{k-\ell K} \sum_{m=1}^{K-1} b^{(w)}_{k-\ell K-m}\, c_{\ell K+m, k} \qquad n = i^{(w)}_{k,\ell}$$

and $i^{(w)}_{k,\ell}$, $w = 0, \ldots, W-1$, $\ell = 0, \ldots, L-1$, are the look-up indices computed as follows:

$$i^{(w)}_{k,\ell} = \begin{cases} \displaystyle\sum_{m=1}^{K-1} a^{(w)}_{k-\ell K-m}\, 2^{m-1} & \text{if } a^{(w)}_{k-\ell K} = 1 \\ \displaystyle\sum_{m=1}^{K-1} \bar{a}^{(w)}_{k-\ell K-m}\, 2^{m-1} & \text{if } a^{(w)}_{k-\ell K} = 0 \end{cases} \tag{16.15}$$

We note that, as long as (16.12) and (16.13) hold for some coefficient vector $[c_{0,k}, \ldots, c_{N-1,k}]$, a distributed-arithmetic filter emulates the operation of a linear transversal filter. For arbitrary values $d_k(n, \ell)$, however, a non-linear filtering operation results.

The expression of the LMS algorithm to update the values of a distributed-arithmetic echo canceller is derived as in (3.280). To simplify the notation we set

$$\hat{u}_k(\ell) = \sum_{w=0}^{W-1} 2^w\, b^{(w)}_{k-\ell K}\, d_k(i^{(w)}_{k,\ell}, \ell) \tag{16.16}$$

hence (16.14) can be written as

$$\hat{u}_k = \sum_{\ell=0}^{L-1} \hat{u}_k(\ell) \tag{16.17}$$

We also define the vector of the values in the ℓ-th look-up table as

$$\mathbf{d}_k(\ell) = [d_k(0, \ell), \ldots, d_k(2^{K-1} - 1, \ell)]^T \tag{16.18}$$

indexed by the variable (16.15).

The values $\mathbf{d}_k(\ell)$ are updated according to the LMS criterion, i.e.,

$$\mathbf{d}_{k+1}(\ell) = \mathbf{d}_k(\ell) - \tfrac{1}{2}\, \mu\, \nabla_{\mathbf{d}_k(\ell)}\, z_k^2 \tag{16.19}$$

where

$$\nabla_{\mathbf{d}_k(\ell)}\, z_k^2 = 2 z_k\, \nabla_{\mathbf{d}_k(\ell)}\, z_k = -2 z_k\, \nabla_{\mathbf{d}_k(\ell)}\, \hat{u}_k = -2 z_k\, \nabla_{\mathbf{d}_k(\ell)}\, \hat{u}_k(\ell) \tag{16.20}$$

The last expression has been obtained using (16.17) and the fact that only $\hat{u}_k(\ell)$ depends on $\mathbf{d}_k(\ell)$. Defining

$$\mathbf{y}_k(\ell) = [y_k(0, \ell), \ldots, y_k(2^{K-1} - 1, \ell)]^T = \nabla_{\mathbf{d}_k(\ell)}\, \hat{u}_k(\ell)$$

(16.19) becomes

$$\mathbf{d}_{k+1}(\ell) = \mathbf{d}_k(\ell) + \mu\, z_k\, \mathbf{y}_k(\ell) \tag{16.21}$$

For a given value of k and ℓ, we assign the following values to the W addresses (16.15):

$$I^{(w)} = i_{k,\ell}^{(w)} \qquad w = 0, 1, \ldots, W-1$$

From (16.16) we get

$$y_k(n, \ell) = \sum_{w=0}^{W-1} 2^w \, b_{k-\ell K}^{(w)} \, \delta_{n-I^{(w)}} \tag{16.22}$$

In conclusion, in (16.21) for every instant k and for each value of the index $w = 0, 1, \ldots, W-1$, the product $2^w \, b_{k-\ell K}^{(w)} \, \mu z_k$ is added to the memory location indexed by $I^{(w)}$. The complexity of the implementation can be reduced by updating, at every iteration k, only the values corresponding to the addresses given by the most significant bits of the symbols in the filter delay line. In this case (16.22) simplifies into

$$y_k(n, \ell) = \begin{cases} 2^{W-1} \, b_{k-\ell K}^{(W-1)} & n = I^{(W-1)} \\ 0 & n \neq I^{(W-1)} \end{cases} \tag{16.23}$$

The block diagram of an adaptive distributed-arithmetic echo canceller with input symbols from a quaternary alphabet is shown in Figure 16.6.

The analysis of the mean-square error convergence behavior and steady-state performance can be extended to adaptive distributed-arithmetic echo cancellers [6]. The dynamics of the mean-square error are in this case given by

$$J_k = \sigma_0^2 \left[1 - \frac{\mu \sigma_a^2}{2^{K-1}} \left(2 - \mu L \sigma_a^2 \right) \right]^k + \frac{2 J_{min}}{2 - \mu L \sigma_a^2} \tag{16.24}$$

The stability condition for the echo canceller is $0 < \mu < 2/(L\sigma_a^2)$. For a given adaptation gain, echo canceller stability depends on the number of tables and on the variance of the transmitted symbols. Therefore, the time span of the echo canceller can be increased without affecting system stability, provided that the number L of tables is kept constant. In that case, however, mean-square error convergence will be slower. From (16.24), we find that the optimum adaptation gain that permits the fastest mean-square error convergence at the beginning of the adaptation process is $\mu_{opt} = 1/(L\sigma_a^2)$. The time constant of the convergence mode is $\tau_{opt} = L2^{K-1}$. The smallest achievable time constant is therefore proportional to the total number of values. As mentioned above, the implementation of a distributed-arithmetic echo canceller can be simplified by updating at each iteration only the values that are addressed by the most significant bits of the symbols stored in the delay line. The complexity required for adaptation can thus be reduced at the price of a slower rate of convergence.

16.2 Echo cancellation for QAM systems

Although most of the concepts presented in the preceding section can be readily extended to echo cancellation for communications systems employing QAM, the case of

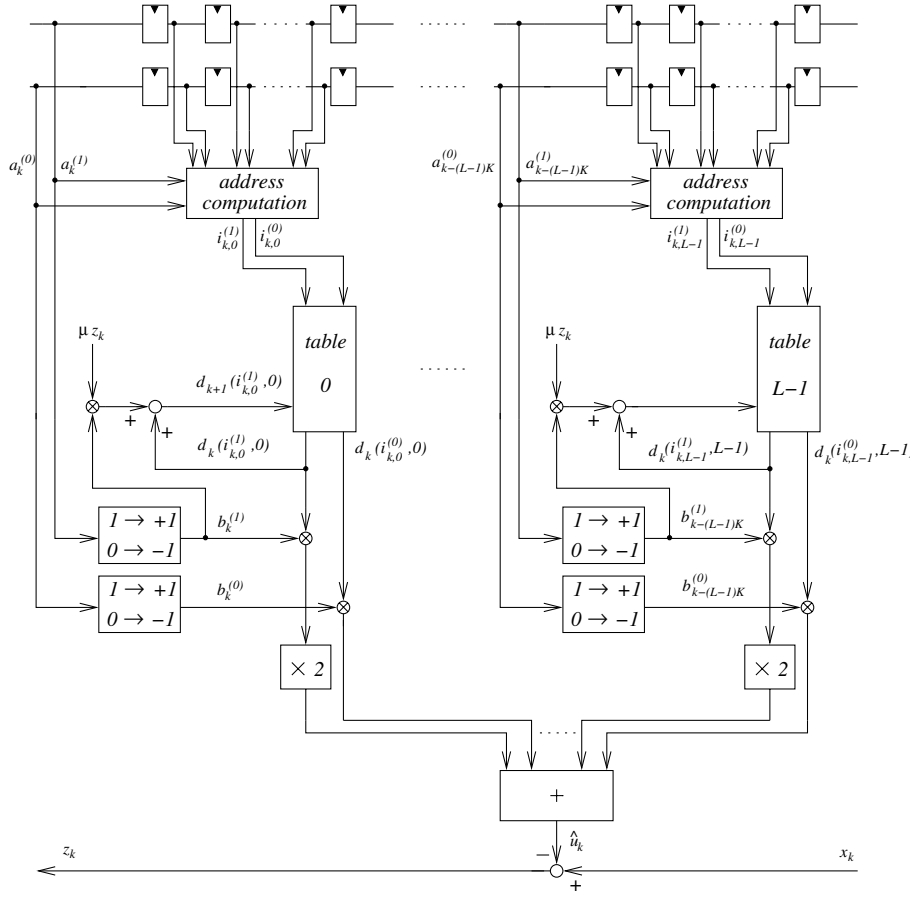


Figure 16.6. Block diagram of an adaptive distributed-arithmetic echo canceller.

full-duplex transmission over a voiceband data channel requires a specific discussion. We consider the system model shown in Figure 16.7. The transmitter generates a sequence $\{a_k\}$ of i.i.d. complex-valued symbols from a two-dimensional constellation $\mathcal{A}$, that are modulated by the carrier $e^{j2\pi f_0 kT}$, where T and f_0 denote the modulation interval and the carrier frequency, respectively. The discrete-time signal at the output of the transmit passband phase splitter filter may be regarded as an analytic signal, which is generated at the rate of Q_0/T samples/s, $Q_0 > 1$. The real part of the analytic signal is converted into an analog signal by a D/A converter and input to the channel. We note that by transmitting the real part of a complex-valued signal, positive and negative-frequency components become folded. The attenuation in the image band of the transmit filter thus determines the achievable echo suppression. In fact, the receiver cannot extract aliasing image-band components from desired passband frequency components, and the echo canceller is able to suppress only echo arising from transmitted passband components.

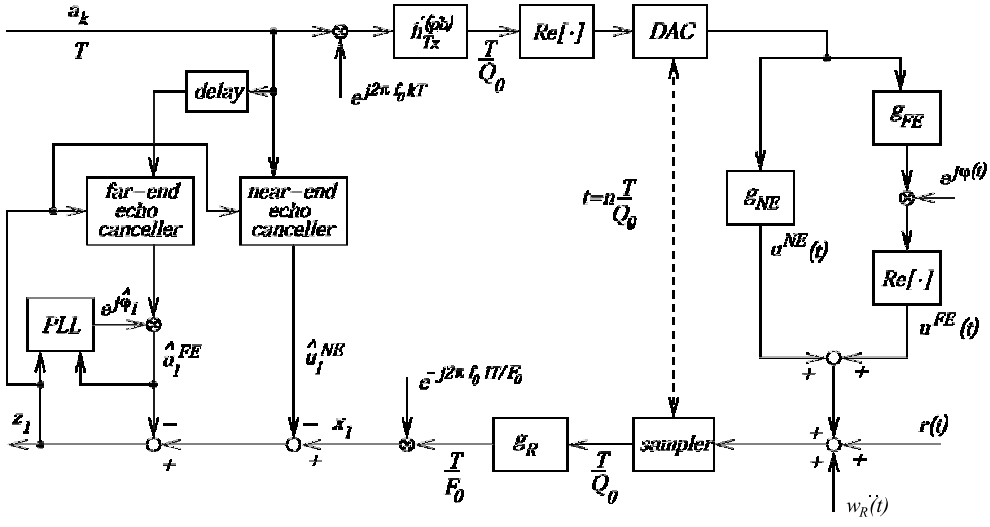


Figure 16.7. Configuration of an echo canceller for a QAM transmission system.

The output of the echo channel is represented as the sum of two contributions. The near-end echo $u^{NE}(t)$ arises from the impedance mismatch between the hybrid and the transmission line, as in the case of baseband transmission. The far-end echo $u^{FE}(t)$ represents the contribution due to echoes that are generated at intermediate points in the telephone network. These echoes are characterized by additional impairments, such as jitter and frequency shift, which are accounted for by introducing a carrier-phase rotation equal to $\varphi(t)$ in the model of the far-end echo.

At the receiver, samples of the signal at the channel output are obtained synchronously with the transmitter timing, at the sampling rate of Q_0/T samples/s. The discrete-time received signal is converted to a complex-valued baseband signal $\{x_{kF_0+i}\}$, $i = 0, \ldots, F_0 - 1$, at the rate of F_0/T samples/s, $1 < F_0 < Q_0$, through filtering by the receive phase splitter filter, decimation, and demodulation. From delayed transmit symbols, estimates of the near and far-end echo signals after demodulation, $\{\hat{u}^{NE}_{kF_0+i}\}$, $i = 0, \ldots, F_0 - 1$, and $\{\hat{u}^{FE}_{kF_0+i}\}$, $i = 0, \ldots, F_0 - 1$, respectively, are generated using F_0 interlaced near and far-end echo cancellers. The cancellation error is given by

$$z_\ell = x_\ell - (\hat{u}^{NE}_\ell + \hat{u}^{FE}_\ell) \tag{16.25}$$

A different model is obtained if echo cancellation is accomplished before demodulation. In this case, two equivalent configurations for the echo canceller may be considered. In one configuration, the modulated symbols are input to the transversal filter, which approximates the passband echo response. Alternatively, the modulator can be placed after the transversal filter, which is then called a baseband transversal filter [7].

In the considered implementation, the estimates of the echo signals after demodulation are given by

$$\hat{u}^{NE}_{kF_0+i} = \sum_{n=0}^{N_{NE}-1} c^{NE}_{nF_0+i,k}\, a_{k-n} \qquad i = 0, \ldots, F_0 - 1 \tag{16.26}$$

$$\hat{u}^{FE}_{kF_0+i} = \left(\sum_{n=0}^{N_{FE}-1} c^{FE}_{nF_0+i,k}\, a_{k-n-D_E} \right) e^{j\hat{\varphi}_{kF_0+1}} \qquad i = 0, \ldots, F_0 - 1 \tag{16.27}$$

where $[c^{NE}_{0,k}, \ldots, c^{NE}_{F_0N_{NE}-1,k}]$ and $[c^{FE}_{0,k}, \ldots, c^{FE}_{F_0N_{FE}-1,k}]$ are the coefficients of the F_0 interlaced near and far-end echo cancellers, respectively, $\{\hat{\varphi}_{kF_0+i}\}$, $i = 0, \ldots, F_0 - 1$, is the sequence of far-end echo phase estimates, and D_{FE} denotes the bulk delay accounting for the round-trip delay from the transmitter to the point of echo generation. To prevent overlap of the time span of the near-end echo canceller with the time span of the far-end echo canceller, the condition $D_{FE} > N_{NE}$ must be satisfied. We also note that, because of the different nature of near and far-end echo generation, the time span of the far-end echo canceller needs to be larger than the time span of the near-end canceller, i.e., $N_{FE} > N_{NE}$.

Adaptation of the filter coefficients in the near and far-end echo cancellers by the LMS algorithm leads to

$$c^{NE}_{nF_0+i,k+1} = c^{NE}_{nF_0+i,k} + \mu\, z_{kF_0+i} (a_{k-n})^* \qquad n = 0, \ldots, N_{NE} - 1 \qquad i = 0, \ldots, F_0 - 1 \tag{16.28}$$

and

$$c^{FE}_{nF_0+i,k+1} = c^{FE}_{nF_0+i,k} + \mu\, z_{kF_0+i} (a_{k-n-D_{FE}})^*\, e^{-j\hat{\varphi}_{kF_0+i}} \qquad n = 0, \ldots, N_{FE} - 1 \qquad i = 0, \ldots, F_0 - 1 \tag{16.29}$$

respectively.

The far-end echo phase estimate is computed by a second-order phase-lock loop algorithm (see Section 14.7), where the following gradient approach is adopted:

$$\begin{cases} \hat{\varphi}_{\ell+1} = \hat{\varphi}_\ell - \frac{1}{2}\, \mu_\varphi\, \nabla_{\hat{\varphi}} |z_\ell|^2 + \Delta\varphi_\ell \pmod{2\pi} \\ \Delta\varphi_{\ell+1} = \Delta\varphi_\ell - \frac{1}{2}\, \mu_f\, \nabla_{\hat{\varphi}} |z_\ell|^2 \end{cases} \tag{16.30}$$

where $\ell = kF_0 + i$, $i = 0, \ldots, F_0 - 1$, μ_φ and μ_f are parameters of the loop, and

$$\nabla_{\hat{\varphi}} |z_\ell|^2 = \frac{\partial |z_\ell|^2}{\partial \hat{\varphi}_\ell} = -2\, Im\{z_\ell (\hat{u}^{FE}_\ell)^*\} \tag{16.31}$$

We note that the algorithm (16.30) requires F_0 iterations per modulation interval, i.e., we cannot resort to interlacing to reduce the complexity of the computation of the far-end echo phase estimate.

16.3 Echo cancellation for OFDM systems

We discuss echo cancellation for OFDM with reference to a DMT system (see Chapter 9), as shown in Figure 16.8. Let $\{h_i\}$, $i = 0, \ldots, N_c - 1$, be the channel impulse response, sampled with period $T/\mathcal{M}$, with length $N_c \ll \mathcal{M}$ and $\{h_{E,i}\}$, $i = 0, \ldots, N - 1$, be the discrete-time echo impulse response whose length is $N < \mathcal{M}$, where $\mathcal{M}$ denotes the number of subchannels of the DMT system. To simplify the notation, the length of the cyclic prefix will be set to $L = N_c - 1$. Recall that in a DMT transmitter the block of $\mathcal{M}$ samples at the IDFT output in the k-th modulation interval, $[A_k[0], \ldots, A_k[\mathcal{M}-1]]$, is cyclically extended by copying the last L samples at the beginning of the block. After a P/S conversion, wherein the L samples of the cyclic extension are the first to be output, the $L + \mathcal{M}$ samples of the block are transmitted into the channel. At the receiver the sequence is split into blocks of length $L + \mathcal{M}$, $[x_{k(\mathcal{M}+L)}, \ldots, x_{(k+1)(\mathcal{M}+L)-1}]$. These blocks are separated in such a way that the last $\mathcal{M}$ samples depend only on a single cyclically extended block, then the first L samples are discarded.

Case $N \leq L+1$. We initially assume that the length of the echo channel impulse response is $N \leq L + 1$. Furthermore, we assume that the boundaries of the received blocks are placed such that the last $\mathcal{M}$ samples of the k-th received block are expressed by the vector (see (9.72))

$$\mathbf{x}_k = \Xi_k^R \, \mathbf{h} + \Xi_k \, \mathbf{h}_E + \mathbf{w}_k \tag{16.32}$$

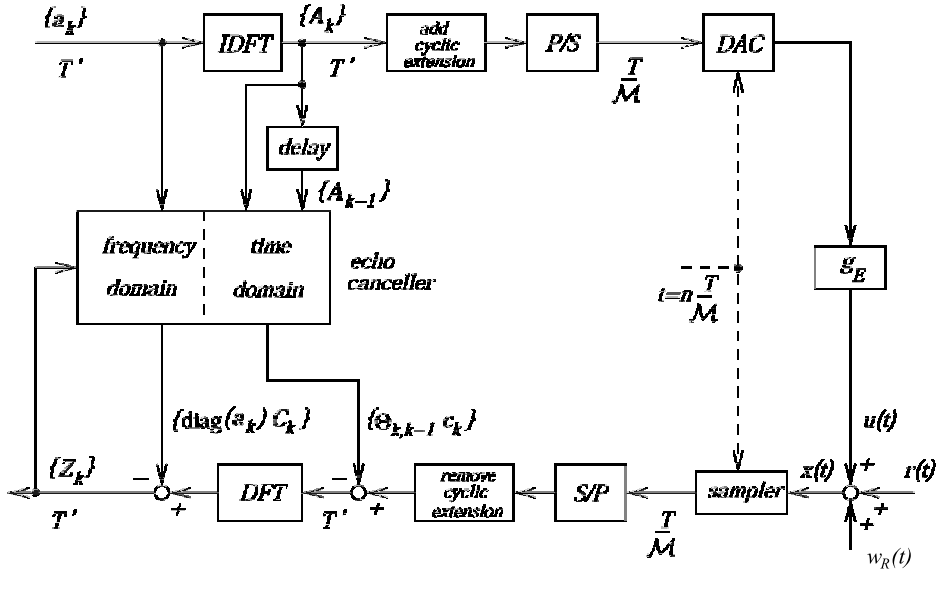


Figure 16.8. Configuration of an echo canceller for a DMT transmission system.

where $\mathbf{h} = [h_0, \ldots, h_L, 0, \ldots, 0]^T$ is the vector of the overall channel impulse response extended with $\mathcal{M} - L - 1$ zeros, $\mathbf{h}_E = [h_{E,0}, \ldots, h_{E,N-1}, 0, \ldots, 0]^T$ is the vector of the overall echo impulse response extended with $\mathcal{M} - N$ zeros, $\mathbf{w}_k$ is a vector of AWGN samples, $\boldsymbol{\Xi}_k^R$ is the circulant matrix with elements given by the signals from the remote transmitter

$$\boldsymbol{\Xi}_k^R = \begin{bmatrix} A_k^R[0] & A_k^R[\mathcal{M}-1] & \ldots & A_k^R[1] \\ A_k^R[1] & A_k^R[0] & \ldots & A_k^R[2] \\ \vdots & \vdots & & \vdots \\ A_k^R[\mathcal{M}-1] & A_k^R[\mathcal{M}-2] & \ldots & A_k^R[0] \end{bmatrix} \tag{16.33}$$

and $\boldsymbol{\Xi}_k$ is the circulant matrix with elements generated by the local transmitter

$$\boldsymbol{\Xi}_k = \begin{bmatrix} A_k[0] & A_k[\mathcal{M}-1] & \ldots & A_k[1] \\ A_k[1] & A_k[0] & \ldots & A_k[2] \\ \vdots & \vdots & & \vdots \\ A_k[\mathcal{M}-1] & A_k[\mathcal{M}-2] & \ldots & A_k[0] \end{bmatrix} \tag{16.34}$$

In the frequency domain, the echo is expressed as

$$\mathbf{U}_k = \text{diag}(\mathbf{a}_k)\, \mathbf{H}_E \tag{16.35}$$

where $\boldsymbol{H}_E$ denotes the DFT of the vector $\boldsymbol{h}_E$. In this case, the echo canceller provides an echo estimate that is given by

$$\hat{\mathbf{U}}_k = \text{diag}(\mathbf{a}_k)\, \mathbf{C}_k \tag{16.36}$$

where $\mathbf{C}_k$ denotes the DFT of the vector $\mathbf{c}_k$ of the N coefficients of the echo canceller filter extended with $\mathcal{M} - N$ zeros.

In the time domain, (16.36) corresponds to the estimate

$$\hat{\mathbf{u}}_k = \boldsymbol{\Xi}_k\, \mathbf{c}_k \tag{16.37}$$

Case $N > L + 1$. In practice, however, we need to consider the case $N > L + 1$. The expression of the cancellation error is then given by

$$\mathbf{x}_k = \boldsymbol{\Xi}_k^R\, \mathbf{h} + \boldsymbol{\Psi}_{k,k-1}\, \mathbf{h}_E + \mathbf{w}_k \tag{16.38}$$

where $\boldsymbol{\Psi}_{k,k-1}$ is a circulant matrix given by

$$\boldsymbol{\Psi}_{k,k-1} = \begin{bmatrix} A_k[0] & A_k[\mathcal{M}-1] & \ldots & A_k[\mathcal{M}-L] & A_{k-1}[\mathcal{M}-1] & \ldots & A_{k-1}[L+1] \\ A_k[1] & A_k[0] & \ldots & A_k[\mathcal{M}-L+1] & A_k[\mathcal{M}-L] & \ldots & A_{k-1}[L+2] \\ \vdots & & & & & & \\ A_k[\mathcal{M}-1] & A_k[\mathcal{M}-2] & \ldots & A_k[\mathcal{M}-L-1] & A_k[\mathcal{M}-L-2] & \ldots & A_k[0] \end{bmatrix} \tag{16.39}$$

From (16.38) the expression of the cancellation error in the time domain is then given by

$$\mathbf{z}_k = \mathbf{x}_k - \mathbf{\Psi}_{k,k-1}\,\mathbf{c}_k \tag{16.40}$$

We now introduce the Toeplitz triangular matrix

$$\mathbf{\Theta}_{k,k-1} = \mathbf{\Psi}_{k,k-1} - \mathbf{\Xi}_k \tag{16.41}$$

Substitution of (16.41) into (16.40) yields

$$\mathbf{z}_k = \mathbf{x}_k - \mathbf{\Theta}_{k,k-1}\,\mathbf{c}_k - \mathbf{\Xi}_k\,\mathbf{c}_k \tag{16.42}$$

In the frequency domain, (16.42) can be expressed as

$$\mathbf{Z}_k = \mathbf{F}_{\mathcal{M}}(\mathbf{x}_k - \mathbf{\Theta}_{k,k-1}\,\mathbf{c}_k) - \mathrm{diag}(\mathbf{a}_k)\,\mathbf{C}_k \tag{16.43}$$

Equation (16.43) suggests a computationally efficient, two-part echo cancellation technique. First, in the time domain, a short convolution is performed and the result subtracted from the received signals to compensate for the insufficient length of the cyclic extension. Second, in the frequency domain, cancellation of the residual echo is performed over a set of $\mathcal{M}$ independent echo subchannels. Observing that (16.43) is equivalent to

$$\mathbf{Z}_k = \mathbf{X}_k - \tilde{\mathbf{\Psi}}_{k,k-1}\,\mathbf{C}_k \tag{16.44}$$

where $\tilde{\mathbf{\Psi}}_{k,k-1} = \mathbf{F}_{\mathcal{M}}\mathbf{\Psi}_{k,k-1}\mathbf{F}_{\mathcal{M}}^{-1}$, the echo canceller adaptation by the LMS algorithm in the frequency domain takes the form

$$\mathbf{C}_{k+1} = \mathbf{C}_k + \mu\,\tilde{\mathbf{\Psi}}_{k,k-1}^{H}\,\mathbf{Z}_k \tag{16.45}$$

where μ is the adaptation gain.

We note that, alternatively, echo canceller adaptation may also be performed by the simplified algorithm [8]

$$\mathbf{C}_{k+1} = \mathbf{C}_k + \mu\,\mathrm{diag}(\mathbf{a}_k^*)\,\mathbf{Z}_k \tag{16.46}$$

which entails a substantially lower computational complexity than the LMS algorithm, at the price of a slower rate of convergence.

In DMT systems it is essential that the length of the channel impulse response be much less than the number of subchannels, so that the reduction in rate due to the cyclic extension may be considered negligible. Therefore, time-domain equalization is adopted in practice to shorten the length of the channel impulse response. From (16.43), however, we observe that transceiver complexity depends on the relative lengths of the echo and of the channel impulse responses. To reduce the length of the cyclic extension as well as the computational complexity of the echo canceller, various methods have been proposed to shorten both the channel and the echo impulse responses jointly [9].

16.4 Multiuser detection for VDSL

In this section, we address the problem of multiuser detection for upstream VDSL transmission (see Chapter 17), where FEXT signals at the input of a VDSL receiver are viewed as interferers that share the same channel as the remote user signal [10].

We assume knowledge of the FEXT responses at the central office and consider a decision-feedback equalizer (DFE) structure with cross-coupled linear feedforward (FF) equalizers and feedback (FB) filters for cross-talk suppression. DFE structures with cross-coupled filters have also been considered for interference suppression in wireless CDMA communications [11] and fast Ethernet transmission (see Appendix 19.A). Here we determine the optimum DFE coefficients in a minimum mean-square error (MMSE) sense assuming that each user adopts OFDM modulation for upstream transmission. A system with reduced complexity may be considered for practical applications, in which for each user and each subchannel only the most significant interferers are suppressed.

To obtain a receiver structure for multiuser detection that exhibits moderate complexity, we assume that each user adopts FMT modulation with $\mathcal{M}$ subchannels for upstream transmission (see Chapter 9). Hence the subchannel signals exhibit non-zero excess bandwidth as well as negligible spectral overlap. Assuming upstream transmission by U users, the system illustrated in Figure 16.9 is considered. In general, the sequences of subchannel signal samples at the multicarrier demodulator output are obtained at a sampling rate equal to a rational multiple F_0 of the modulation rate $1/T$. To simplify the analysis, here an integer $F_0 \geq 2$ is assumed.

We introduce the following definitions:

1. $\{a_k^{(u)}[i]\}$, sequence of i.i.d. complex-valued symbols from a QAM constellation $\mathcal{A}^{(u)}[i]$ transmitted by user u over subchannel i, with variance $\sigma^2_{a^{(u)}[i]}$;

2. $\{\hat{a}_k^{(u)}[i]\}$, sequence of detected symbols of user u at the output of the decision element of subchannel i;

3. $h_n^{(u)}[i]$, overall impulse response of subchannel i of user u;

4. $h_{\mathrm{FEXT},n}^{(u,v)}[i]$, overall FEXT response of subchannel i, from user v to user u;

5. $G^{(u)}[i]$, gain that determines the power of the signal of user u transmitted over subchannel i;

6. $\{\tilde{w}_n^{(u)}[i]\}$, sequence of additive Gaussian noise samples with correlation function $\mathrm{r}_{w^{(u)}[i]}(m)$.

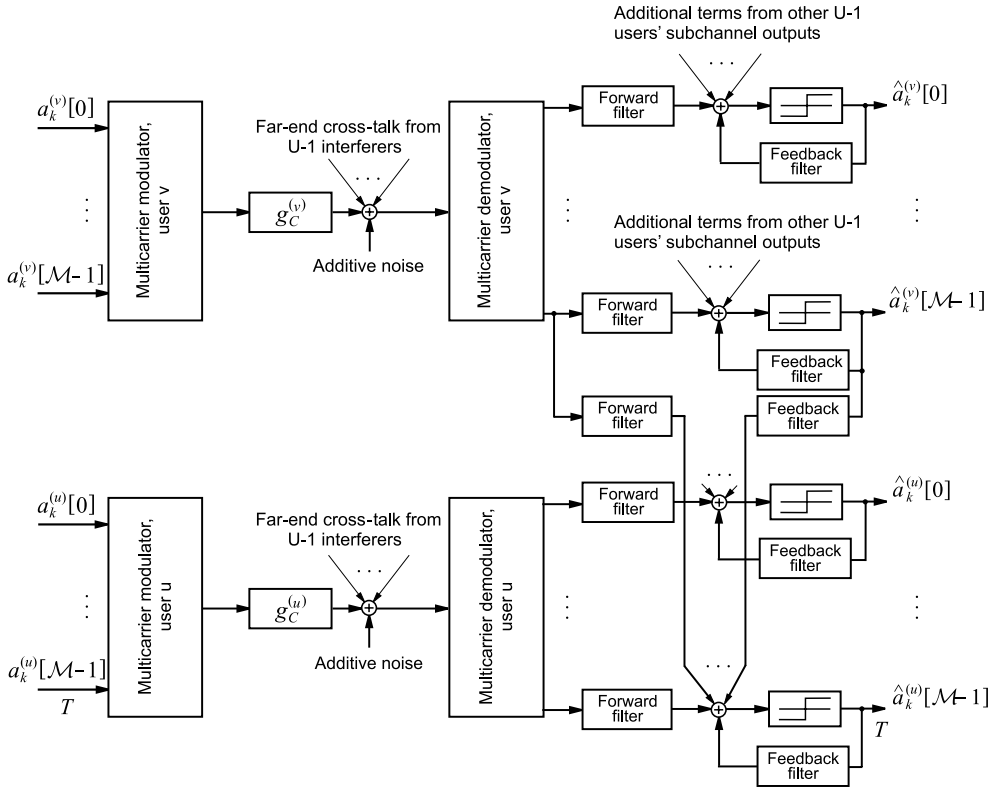


Figure 16.9. Block diagram of transmission channel and DFE structure. [*From [10], © 2001 IEEE*.]

At the output of subchannel i of the user-u demodulator, the complex baseband signal is given by

$$x_n^{(u)}[i] = G^{(u)}[i] \sum_{k=-\infty}^{\infty} h_{n-kF_0}^{(u)}[i]\, a_k^{(u)}[i] + \\ + \sum_{\substack{v=1 \\ v \neq u}}^{U} G^{(v)}[i] \sum_{k=-\infty}^{\infty} h_{\mathrm{FEXT},n-kF_0}^{(u,v)}[i] a_k^{(v)}[i] + \tilde{w}_n^{(u)}[i] \tag{16.47}$$

For user u, symbol detection at the output of subchannel i is achieved by a DFE structure such that the input to the decision element is obtained by combining the output signals of U linear filters and U feedback filters from all users, as illustrated in Figure 16.9.

In practice, to reduce system complexity, for each user only a subset of all other user signals (interferers) is considered as an input to the DFE structure [10]. The selection of the subset signals is based on the power and the number of the interferers. This strategy, however, results in a loss of performance, as some strong interferers may not be considered.

This effect is similar to the near–far problem in CDMA systems. To alleviate this problem, it is necessary to introduce power control of the transmitted signals; a suitable method will be described in the next section.

To determine the DFE filter coefficients, we assume M_1 and M_2 coefficients for each FF and FB filter, respectively. We define the following vectors:

1. $\mathbf{x}_{kF_0}^{(u)}[i] = [x_{kF_0}^{(u)}[i], \ldots, x_{kF_0-M_i+1}^{(u)}[i]]^T$, signal samples stored at instant kF_0 in the delay lines of the FF filters with input given by the demodulator output of user u at subchannel i;
2. $\mathbf{c}^{(u,v)}[i] = [c_0^{(u,v)}[i], \ldots, c_{M_1-1}^{(u,v)}[i]]^T$, coefficients of the FF filter from the demodulator output of user v to the decision element input of user u at subchannel i;
3. $\mathbf{b}^{(u,v)}[i] = [b_1^{(u,v)}[i], \ldots, b_{M_2}^{(u,v)}[i]]^T$, coefficients of the FB filter from the decision element output of user v to the decision element input of user u at subchannel i;
4. $\hat{\mathbf{a}}_k^{(u)}[i] = \left[\hat{a}^{(u)}_{k-1-\left\lceil \frac{D^{(u)}[i]}{F_0} \right\rceil}[i], \ldots, \hat{a}^{(u)}_{k-M_2-\left\lceil \frac{D^{(u)}[i]}{F_0} \right\rceil}[i] \right]^T$, symbol decisions stored at instant k in the delay lines of the FB filters with input given by the decision element output of user u at subchannel i, where $D^{(u)}[i]$ is a suitable integer delay related to the DFE; we assume no decision errors, that is $\hat{a}_k^{(u)}[i] = a_k^{(u)}[i]$.

The input to the decision element of user u at subchannel i at instant k is then expressed by (see Section 8.5)

$$y_k^{(u)}[i] = \mathbf{c}^{(u,u)^T}[i]\,\mathbf{x}_{kF_0}^{(u)}[i] + \mathbf{b}^{(u,u)^T}[i]\,\mathbf{a}_k^{(u)}[i] + \\ + \sum_{\substack{v=1 \\ v \neq u}}^{U} \{\mathbf{c}^{(u,v)^T}[i]\,\mathbf{x}_{kF_0}^{(v)}[i] + \mathbf{b}^{(u,v)^T}[i]\,\mathbf{a}_k^{(v)}[i]\} \tag{16.48}$$

and the error signal is given by

$$e_k^{(u)}[i] = y_k^{(u)}[i] - a^{(u)}_{k-\left\lceil \frac{D^{(u)}[i]}{F_0} \right\rceil} \tag{16.49}$$

Without loss of generality, we extend the technique developed in Section 8.5 for the single-user fractionally-spaced DFE to determine the optimum coefficients of the DFE structure for user $u = 1$. We introduce the following vectors and matrices:

1. $\mathbf{h}_m^{(u)}[i] = G^{(u)}[i][h^{(u)}_{mF_0+M_1-1+D^{(u)}[i]}[i], \ldots, h^{(u)}_{mF_0+D^{(u)}[i]}[i]]^T$, vector of M_1 samples of the impulse response of subchannel i of user u.
2. $\mathbf{h}^{(u,v)}_{\mathrm{FEXT},m}[i] = G^{(v)}[i][h^{(u,v)}_{\mathrm{FEXT},mF_0+M_1-1+D^{(u,v)}[i]}[i], \ldots, h^{(u,v)}_{\mathrm{FEXT},mF_0+D^{(u,v)}[i]}[i]]^T$, vector of M_1 samples of the FEXT impulse response of subchannel i from user v to user u; assuming the differences between the propagation delays of the signal of user u and of the cross-talk signals originated by the other users are negligible, we have $D^{(u,v)}[i] = D^{(u)}[i]$.

3.

$$\mathbf{R}^{(1,1)}[i] = E[\mathbf{x}_k^{(1)^*}[i]\,\mathbf{x}_k^{(1)^T}[i]] - \sum_{m=1}^{M_2}\left(\sigma^2_{a^{(1)}[i]}\mathbf{h}_m^{(1)*}[i]\,\mathbf{h}_m^{(1)^T}[i] + \sum_{v=2}^{V}\sigma^2_{a^{(v)}[i]}\mathbf{h}_{\mathtt{FEXT},m}^{(1,v)^*}[i]\,\mathbf{h}_{\mathtt{FEXT},m}^{(1,v)^T}[i]\right) \tag{16.50}$$

$$\mathbf{R}^{(l,l)}[i] = E[\mathbf{x}_k^{(l)^*}[i]\,\mathbf{x}_k^{(l)^T}[i]] - \sum_{m=1}^{M_2}(\sigma^2_{a^{(l)}[i]}\mathbf{h}_m^{(l)^*}[i]\,\mathbf{h}_m^{(l)^T}[i] + \sigma^2_{a^{(1)}[i]}\mathbf{h}_{\mathtt{FEXT},m}^{(l,1)^*}[i]\,\mathbf{h}_{\mathtt{FEXT},m}^{(l,1)^T}[i]) \qquad l = 2,\ldots U \tag{16.51}$$

$$\mathbf{R}^{(l,1)}[i] = E[\mathbf{x}_k^{(l)^*}[i]\,\mathbf{x}_k^{(1)^T}[i]] - \sum_{m=1}^{M_2}\left(\sigma^2_{a^{(1)}[i]}\mathbf{h}_{\mathtt{FEXT},m}^{(l,1)^*}[i]\,\mathbf{h}_m^{(1)^T}[i] + \sigma^2_{a^{(l)}[i]}\mathbf{h}_{\mathtt{FEXT},m}^{(l)^*}[i]\,\mathbf{h}_{\mathtt{FEXT},m}^{(l,1)^T}[i] + \sum_{\substack{p=2\\p\neq l}}^{U}\sigma^2_{a^{(p)}[i]}\mathbf{h}_{\mathtt{FEXT},m}^{(l,p)^*}[i]\,\mathbf{h}_{\mathtt{FEXT},m}^{(1,p)^T}[i]\right) \qquad l = 2,\ldots U \tag{16.52}$$

$$\mathbf{R}^{(l,j)}[i] = E[\mathbf{x}_k^{(l)^*}[i]\,\mathbf{x}_k^{(j)^T}[i]] - \sum_{m=1}^{M_2}(\sigma^2_{a^{(1)}[i]}\mathbf{h}_{\mathtt{FEXT},m}^{(l,1)^*}[i]\,\mathbf{h}_{\mathtt{FEXT},m}^{(j,1)^T}[i] + \sigma^2_{a^{(j)}[i]}\mathbf{h}_{\mathtt{FEXT},m}^{(l,j)^*}[i]\,\mathbf{h}_m^{(j)^T}[i]) \qquad 1 < j < l \leq U \tag{16.53}$$

where all above matrices are $M_1 \times M_1$ square matrices.

4.

$$\mathbf{R}^{(1)}[i] = \begin{bmatrix} \mathbf{R}^{(1,1)}[i] & \mathbf{R}^{(1,2)}[i] & \ldots & \mathbf{R}^{(1,U)}[i] \\ \mathbf{R}^{(2,1)}[i] & \mathbf{R}^{(2,2)}[i] & \ldots & \mathbf{R}^{(2,U)}[i] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}^{(U,1)}[i] & \mathbf{R}^{(U,2)}[i] & \ldots & \mathbf{R}^{(U,U)}[i] \end{bmatrix} \tag{16.54}$$

where $\mathbf{R}^{(1)}[i]$ in general is a positive semi-definite Hermitian matrix, for which we assume here the inverse exists.

5.

$$\mathbf{p}^{(1)}[i] = \sigma^2_{a^{(1)}[i]}[\mathbf{h}_0^{(1)}[i], \mathbf{h}^{(2,1)}_{\mathtt{FEXT},0}[i], \ldots, \mathbf{h}^{(U,1)}_{\mathtt{FEXT},0}[i]]^T \tag{16.55}$$

Defining the vectors $\mathbf{c}^{(1)}[i] = [\mathbf{c}^{(1,1)^T}[i], \mathbf{c}^{(1,2)^T}[i], \ldots, \mathbf{c}^{(1,U)^T}[i]]^T$, and $\mathbf{b}^{(1)}[i] = [\mathbf{b}^{(1,1)^T}[i], \ldots, \mathbf{b}^{(1,U)^T}[i]]^T$, the optimum coefficients are given by

$$\mathbf{c}^{(1)}_{opt}[i] = [\mathbf{R}^{(1)}[i]]^{-1}\mathbf{p}^{(1)}[i] \tag{16.56}$$

and

$$\mathbf{b}^{(1)}_{opt}[i] = \begin{bmatrix} \mathbf{h}_1^{(1)^T}[i], \mathbf{h}^{(2,1)^T}_{\mathtt{FEXT},1}[i], \ldots, \mathbf{h}^{(U,1)^T}_{\mathtt{FEXT},1}[i] \\ \vdots \\ \mathbf{h}^{(1)^T}_{M_1}[i], \mathbf{h}^{(2,1)^T}_{\mathtt{FEXT},M_1}[i], \ldots, \mathbf{h}^{(U,1)^T}_{\mathtt{FEXT},M_1}[i] \\ \mathbf{h}^{(1,2)^T}_{\mathtt{FEXT},1}[i], \mathbf{h}_1^{(2)^T}[i], \ldots, \mathbf{h}_1^{(U,2)^T}[i] \\ \vdots \\ \mathbf{h}^{(1,2)^T}_{\mathtt{FEXT},M_1}[i], \mathbf{h}^{(2)^T}_{\mathtt{FEXT},M_1}[i], \ldots, \mathbf{h}^{(U,2)^T}_{\mathtt{FEXT},M_1}[i] \\ \vdots \\ \mathbf{h}^{(1,U)^T}_{\mathtt{FEXT},1}[i], \mathbf{h}^{(2,U)^T}_{\mathtt{FEXT},1}[i], \ldots, \mathbf{h}_1^{(U)^T}[i] \\ \vdots \\ \mathbf{h}^{(1,U)^T}_{\mathtt{FEXT},M_1}[i], \mathbf{h}^{(2,U)^T}_{\mathtt{FEXT},M_1}[i], \ldots, \mathbf{h}^{(U)^T}_{\mathtt{FEXT},M_1}[i] \end{bmatrix} \mathbf{c}^{(1)}_{opt}[i] \tag{16.57}$$

The MMSE value at the decision point of user 1 on subchannel i is thus given by

$$J^{(1)}_{min}[i] = \sigma^2_{a^{(1)}[i]} - \mathbf{p}^{(1)^H}[i]\mathbf{c}^{(1)}_{opt}[i] \tag{16.58}$$

The performance of an OFDM system is usually measured in terms of *achievable bit rate* for given channel and cross-talk characteristics (see Chapter 13). The number of bits per modulation interval than can be loaded with a bit-error probability of 10^{-7} on subchannel i is given by (see (13.15))

$$\mathfrak{b}^{(1)}[i] = \log_2\left(1 + \frac{\sigma^2_{a^{(1)}[i]}}{J^{(1)}_{min}[i]}\, 10^{(G_{code} - \overline{\Gamma}_{gap,dB})/10}\right) \tag{16.59}$$

where G_{code} is the coding gain assumed to be the same for all users and all subchannels. The achievable bit rate for user 1 is therefore given by

$$R_b^{(1)} = \frac{1}{T} \sum_{i=0}^{\mathcal{M}-1} \mathfrak{b}^{(1)}[i] \ \text{bit/s} \tag{16.60}$$

16.4.1 Upstream power back-off

Upstream *power back-off* (PBO) methods are devised to allow remote users in a VDSL system to achieve a fair distribution of the available capacity in the presence of FEXT [12]. The upstream VDSL transmission rates, which are achievable with PBO methods, usually depend on parameters, for example, a reference length or the integral of the algorithm of the received signal power spectral density, that are obtained as the result of various trade-offs between services to be offered and allowed maximum line length. However, the application of such PBO methods results in a suboptimum allocation of the signal power for upstream transmission. It is desirable to devise a PBO algorithm with the following characteristics:

- for each individual user, the transmit signal PSD is determined by taking into account the distribution of known target rates and estimated line lengths of users in the network, and
- the total power for upstream signals is kept to a minimum to reduce interference with other services in the same cable binder.

In the preceding section, we found the expression (16.60) of the achievable upstream rate for a user in a VDSL system with U users, assuming perfect knowledge of FEXT impulse responses and multiuser detection. However, PBO may be applied by assuming only the knowledge of the statistical behavior of FEXT coupling functions with no attempt to cancel interference. In this case, the achievable bit rate of user u is given by

$$R_b^{(u)} = \int_{\mathcal{B}} \log_2 \left[1 + \frac{\mathcal{P}^{(u)}(f)|\mathcal{H}^{(u)}(f)|^2}{\sum\limits_{\substack{v=1 \\ v \neq u}}^{U} \mathcal{P}^{(v)}(f)|\mathcal{H}_{\text{FEXT}}^{(u,v)}(f)|^2 + N_0} 10^{(G_{code} - \Gamma_{gap,dB})/10} \right] df \tag{16.61}$$

where $\mathcal{P}^{(u)}(f)$ denote the PSD of the signal transmitted by user u, $\mathcal{H}^{(u)}(f)$ is the frequency response of the channel for user u, $\mathcal{H}_{\text{FEXT}}^{(u,v)}(f)$ is the FEXT frequency response from user v to user u, and N_0 is the PSD of additive white Gaussian noise. From Section 4.4.2, the expression of the average FEXT power coupling function is given by

$$|\mathcal{H}_{\text{FEXT}}^{(u,v)}(f)|^2 = k_t f^2 \min(L_u, L_v)|\mathcal{H}^{(v)}(f)|^2 \tag{16.62}$$

where L_u and L_v denote the lengths of the lines of user u and v, respectively, and k_t is a constant.

Assuming that the various functions are constant within each subchannel band for OFDM modulation, we approximate (16.61) as

$$R_b^{(u)} = \sum_{i=0}^{\mathcal{M}-1} \frac{1}{T} \log_2 \left[1 + \frac{\mathcal{P}^{(u)}(f_i)|\mathcal{H}^{(u)}(f_i)|^2}{\sum\limits_{\substack{v=1 \\ v \neq u}}^{U} \mathcal{P}^{(v)}(f_i)|\mathcal{H}_{\text{FEXT}}^{(u,v)}(f_i)|^2 + N_0} 10^{(G_{code} - \Gamma_{gap,dB})/10} \right] \tag{16.63}$$

where f_i denotes the center frequency of subchannel i.

Let $\overline{\mathcal{P}}^{(u)}(f_i)$ denote the PSD of the signal transmitted by user u on subchannel i with gain $G^{(u)}[i] = 1$. Then the PBO problem can be formulated as follows: find the minimum of the function

$$\sum_{u=1}^{U} \int_{\mathcal{B}} P^{(u)}(f)\, df \simeq \sum_{u=1}^{U} \sum_{i=0}^{\mathcal{M}-1} \frac{1}{T} (G^{(u)}[i])^2 \overline{\mathcal{P}}^{(u)}(f_i) \tag{16.64}$$

subject to the constraints:

1.
$$0 \leq (G^{(u)}[i])^2 \overline{\mathcal{P}}^{(u)}(f_i) \leq \mathcal{P}_{max} \qquad u = 1, \ldots, U \quad i = 0, \ldots, \mathcal{M} - 1 \tag{16.65}$$
and

2.
$$R_b^{(u)} \geq R_{b,\text{target}}^{(u)} \qquad u = 1, \ldots, U \tag{16.66}$$
where $\mathcal{P}_{max}$ is a constant maximum PSD value and $R_{b,\text{target}}^{(u)}$ is the target rate for user u.

In (16.66), $R_b^{(u)}$ is given by (16.60) or (16.63), depending on the receiver implementation.

Finding the optimum upstream transmit power distribution for each user is therefore equivalent to solving a non-linear programming problem in the $U\mathcal{M}$ parameters $G^{(u)}[i]$, $u = 1, \ldots, U$, $i = 0, \ldots, \mathcal{M} - 1$. The optimum values of these parameters that minimize (16.64) can be found by simulated annealing [13, 14].

16.4.2 Comparison of PBO methods

ETSI has defined two modes of operation, named A and B, for PBO [12]. For a scenario using upstream VDSL transmission of two adjacent links with unequal lengths, mode A states that the signal-to-noise ratio degradation to either link shall not exceed 3 dB relative to the equal-length FEXT case. Mode B requires that the signal-to-noise ratio on the longer line shall not be degraded relative to the equal-length FEXT case; furthermore, degradation to the signal-to-noise ratio on the shorter line shall be bounded such that the shorter line can support at least the upstream rate supported on the longer line. Several methods compliant

with either mode A or B have been proposed. PBO methods are also classified into methods that allow shaping of the PSD of the transmitted upstream VDSL signal, e.g., the *equalized FEXT* method, and methods that lead to an essentially flat PSD of the transmitted signal over each individual upstream band, e.g., the *average log* method. Both the equalized FEXT and the average log method, which are described below, comply with mode B.

The equalized FEXT method requires that the PSD of user u be computed as [12]

$$\mathcal{P}^{(u)}(f) = \min\left[\frac{L_{\text{ref}}|\mathcal{H}_{\text{ref}}(f)|^2}{L_v|\mathcal{H}^{(u)}(f)|^2}\mathcal{P}_{max}, \mathcal{P}_{max}\right] \tag{16.67}$$

where L_{ref} and $\mathcal{H}_{\text{ref}}$ denote a reference length and a reference channel frequency response, respectively.

The average log method requires that, for an upstream channel in the frequency band (f_1, f_2), user u adopt a constant PSD given by [15]

$$\mathcal{P}^{(u)}(f) = \mathcal{P}^{(u)}_{(f_1,f_2)} \qquad f \in (f_1, f_2) \tag{16.68}$$

where $\mathcal{P}^{(u)}_{(f_1,f_2)}$ is a constant PSD level chosen such that it satisfies the condition

$$\int_{f_1}^{f_2} \log_2[\mathcal{P}^{(u)}_{(f_1,f_2)}|\mathcal{H}^{(u)}(f)^2|]\,df = K_{(f_1,f_2)} \tag{16.69}$$

where $K_{(f_1,f_2)}$ is a constant.

In this section, the achievable rates of VDSL upstream transmission using the optimum algorithm (16.64) and the average log method are compared for various distances and services. The numerical results presented in this section are derived assuming a 26-gauge telephone twisted-pair cable (see Table 4.2). The noise models for the alien-cross-talk disturbers at the line termination and at the network termination are taken as specified in [16] for the fiber-to-the-exchange case. Additive white Gaussian noise with a power spectral density of –140 dBm/Hz is assumed. We consider upstream VDSL transmission of $U = 40$ users over the frequency band given by the union of $\mathcal{B}_1$ = (2.9 MHz, 5.1 MHz) and $\mathcal{B}_2$ = (7.05 MHz, 12.0 MHz), similar to those specified in [12]. The maximum PSD value is $\mathcal{P}_{max} = -60$ dBm/Hz. FEXT power coupling functions are determined according to (16.62), where $k_t = 6.65\times10^{-21}$. Upstream transmission is assumed to be based on FMT modulation with bandwidth of the individual subchannels equal to 276 kHz and excess bandwidth of 12.5%; for an efficient implementation, a frequency band of (0, 17.664 MHz) is assumed, with $\mathcal{M} = 64$ subchannels, of which only 26 are used. For the computation of the achievable rates, for an error probability of 10^{-7} a signal-to-noise ratio gap to capacity equal to $\overline{\Gamma}'_{\text{gap,dB}} = \overline{\Gamma}_{\text{gap,dB}} + 6 = 15.8$ dB, which includes a 6 dB margin against additional noise sources that may be found in the DSL environment [17], and $G_{\text{code}} = 5.5$ dB are assumed.

For each of the methods and for given target rates we consider two scenarios: the users are i) all the same distance L from the central office, and ii) uniformly distributed at ten different nodes, having distances $jL_{max}/10$, $j = 1, \ldots, 10$, from the central office. To assess the performance of each method, the maximum line length L_{max} is found, such that all users can reliably achieve a given target rate $R_{b,\text{target}} = 13$ MBit/s. The achievable rates are also computed for the case that all users are at the same distance from the central office and no PBO is applied.

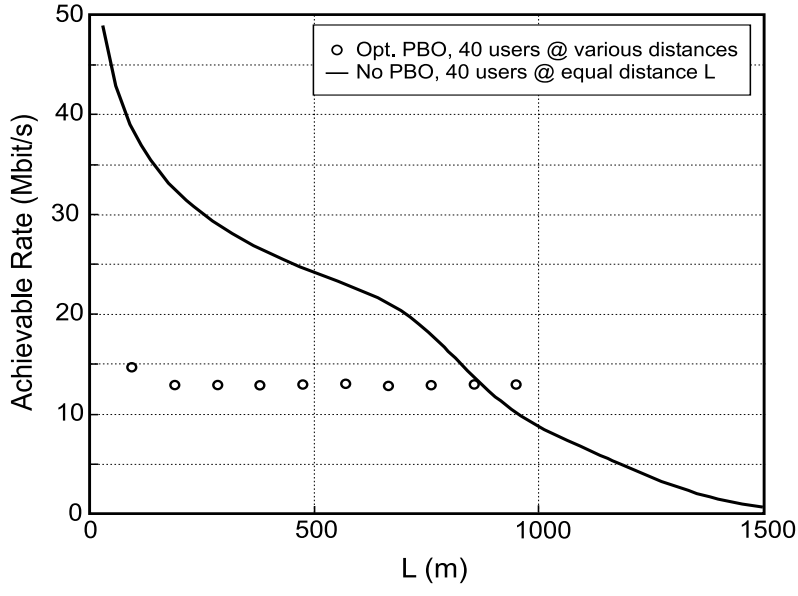


Figure 16.10. Achievable rates of individual users versus cable length using the optimum upstream PBO algorithm for a target rate of 13 Mbit/s. [*From [10], © 2001 IEEE.*]

For the optimum algorithm, the achievable rates are computed using (16.63). Furthermore, different subchannel gains may be chosen for the two bands, but transmission gains within each band are equal. Figure 16.10 shows the achievable rates for each group of four users with the optimum algorithm for the given target rate. The maximum line length L_{max} for scenario ii) turns out to be 950 m. For application to scenario ii), the optimum algorithm requires the computation of 20 parameters. Note that for all users at the same distance from the central office, i.e., scenario i), the optimum algorithm requires the computation of two gains equal for all users. For scenario i), the achievable rate is equal to the target rate up to a certain characteristic length L_{max}, which corresponds to the length for which the target rate is achieved without applying any PBO. Also note that L_{max} for scenario ii) is larger than the characteristics length found for scenario i).

Figure 16.11 illustrates the achievable rates with the average log algorithm (16.69). Joint optimization of the two parameters $K_{\mathcal{B}_1}$ and $K_{\mathcal{B}_2}$ for maximum reach under scenario ii) yields $K_{\mathcal{B}_1} = 0.02$ mW, $K_{\mathcal{B}_2} = 0.05$ mW, and $L_{max} = 780$ m. By comparison with Figure 16.10, we note that for the VDSL transmission spectrum plan considered, optimum upstream PBO leads to an increase in the maximum reach of up to 20%. This increase depends on the distribution of target rates and line lengths of the users in the network.

At this point, further observations can be made on the application of PBO.

- Equal upstream services have been assumed for all users. The optimum algorithm described is even better suited for mixed-service scenarios.

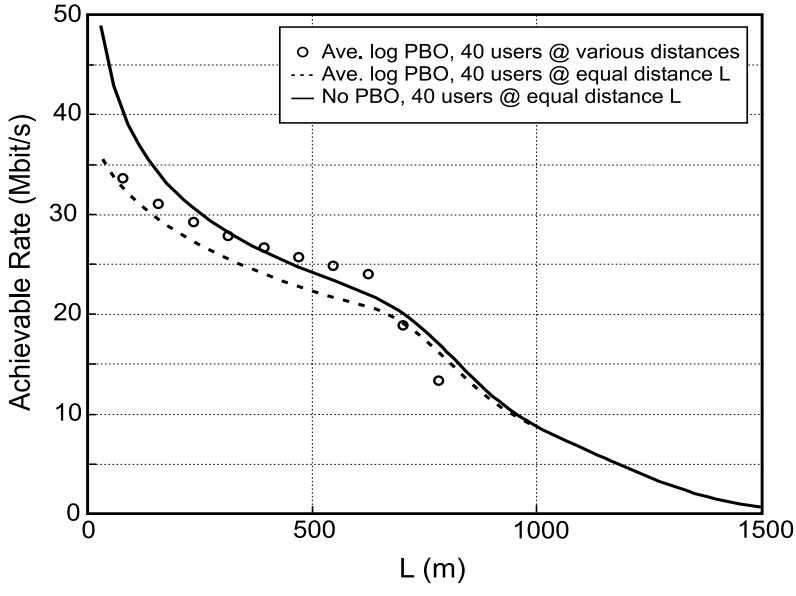


Figure 16.11. Achievable rates of individual users versus cable length using the average log upstream PBO method for a target rate of 13 Mbit/s. [*From [10], © 2001 IEEE.*]

- The application of PBO requires the transmit PSDs of the individual user signals to be recomputed at the central office whenever one or more users join the network or drop out of the network.

To illustrate system performance achievable with multiuser detection, we consider $U = 20$ users, uniformly distributed at ten different nodes having distances $jL_{max}/10$, $j = 1, \ldots, 10$, from the central office, where identification of FEXT impulse responses is performed. For the computation of the achievable rates of individual users, the FEXT impulse responses are generated by a statistical model, and $L_{max} = 500$ m is assumed. Furthermore, to assess the relative merits of multiuser detection and coding, the achievable rates are computed for the two cases of uncoded and coded transmission. For coded transmission, a powerful coding technique yielding 8.5 dB coding gain for an error probability of 10^{-7} is assumed. Figure 16.12 illustrates the achievable rates for perfect suppression of all interferers, which corresponds to the single-user bound. For comparison, the achievable rates for the case that all users are the same distance away from the central office and neither multiuser detection nor coding are applied are also given. Figure 16.13 illustrates the achievable rates for perfect suppression of the ten worst interferers and no application of PBO. We observe that, without PBO, the partial application of multiuser detection does not lead to a significant increase of achievable rates for all users, even assuming large coding gains. Finally, Figure 16.14 depicts the achievable rates obtained for perfect suppression of the ten worst interferers and application of the optimum PBO algorithm with target rate of $R_{b,\text{target}} = 75$ Mbit/s for all users. The target rate is achieved by all users with the joint application of multiuser detection, coding, and power back-off.

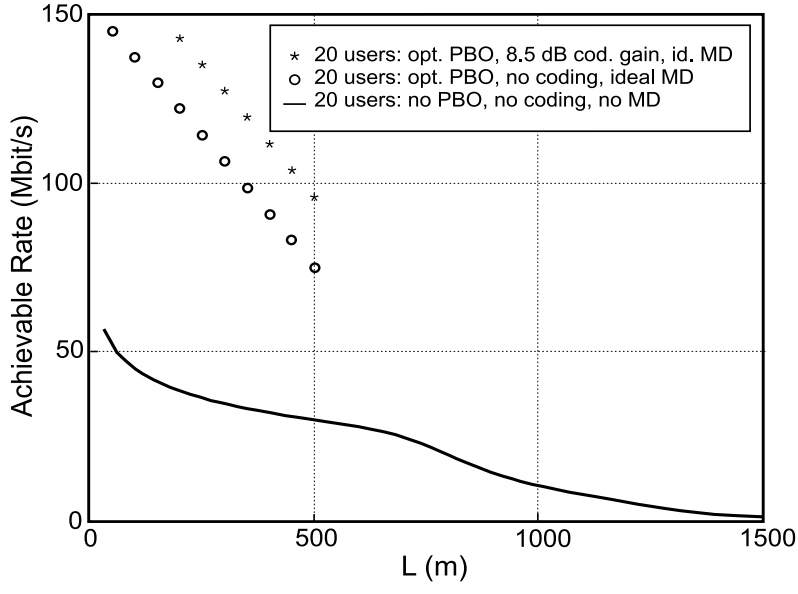


Figure 16.12. Achievable rates of individual users versus cable length with all interferers suppressed. [*From [10], © 2001 IEEE*.]

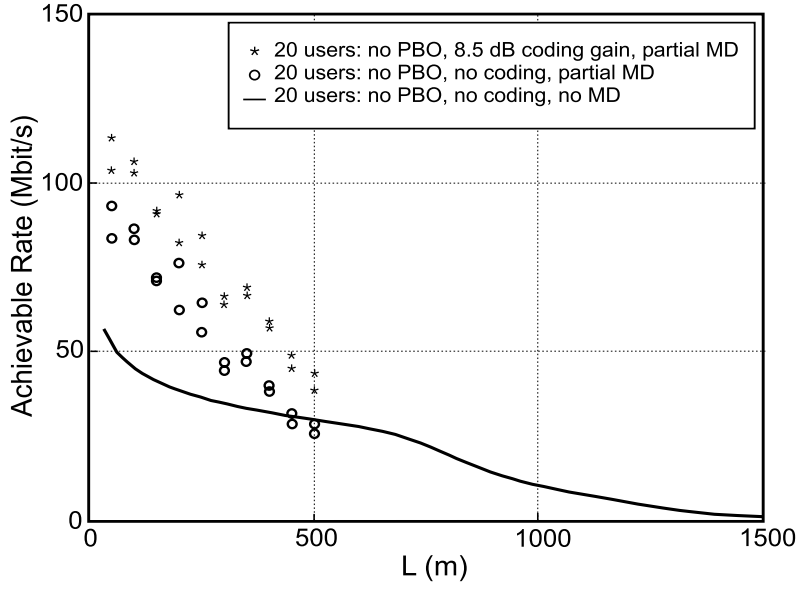


Figure 16.13. Achievable rates of individual users versus cable length with ten interferers suppressed and no PBO applied. [*From [10], © 2001 IEEE*.]

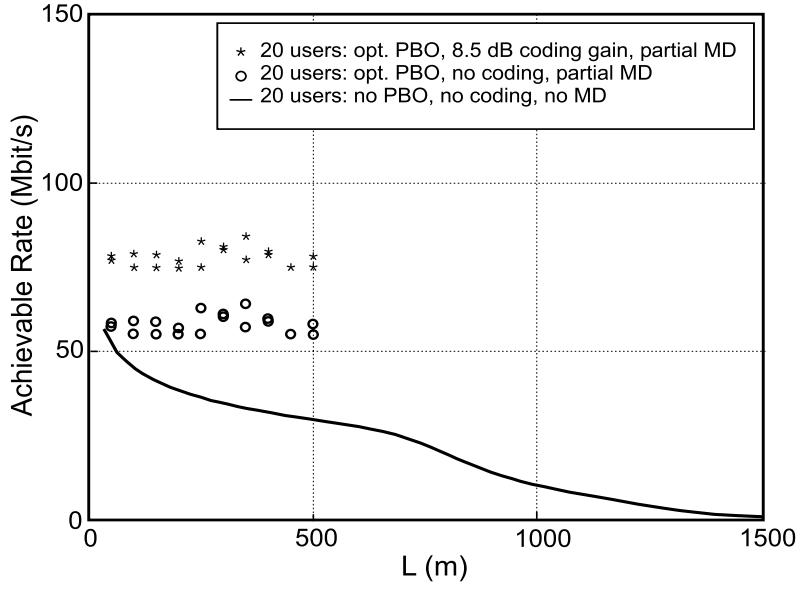


Figure 16.14. Achievable rates of individual users versus cable length with ten interferers suppressed and optimum PBO applied for a target rate of 75 Mbit/s. [*From [10], © 2001 IEEE*.]

To summarize the results of this section, a substantial increase in performance with respect to methods that do not require the identification of FEXT responses is achieved by resorting to reduced-complexity multiuser detection in conjunction with power back-off. This approach yields a performance close to the single-user bound shown in Figure 16.12.

Bibliography

[1] D. G. Messerschmitt and E. A. Lee, *Digital communication*. Boston, MA: Kluwer Academic Publishers, 2nd ed., 1994.

[2] D. G. Messerschmitt, "Design issues for the ISDN U-Interface transceiver", *IEEE Journal on Selected Areas in Communications*, vol. 4, pp. 1281–1293, Nov. 1986.

[3] D. D. Falconer, "Adaptive reference echo-cancellation", *IEEE Trans. on Communications*, vol. 30, pp. 2083–2094, Sept. 1982.

[4] D. L. Duttweiler, "Adaptive filter performance with nonlinearities in the correlation multiplier", *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 30, pp. 578–586, Aug. 1982.

[5] M. J. Smith, C. F. N. Cowan, and P. F. Adams, "Nonlinear echo cancelers based on transposed distributed arithmetic", *IEEE Trans. on Circuits and Systems*, vol. 35, pp. 6–18, Jan. 1988.

[6] G. Cherubini, "Analysis of the convergence behavior of adaptive distributed-arithmetic echo cancelers", *IEEE Trans. on Communications*, vol. 41, pp. 1703–1714, Nov. 1993.

[7] S. B. Weinstein, "A passband data-driven echo-canceler for full-duplex transmission on two-wire circuits", *IEEE Trans. on Communications*, vol. 25, pp. 654–666, July 1977.

[8] M. Ho, J. M. Cioffi, and J. A. C. Bingham, "Discrete multitone echo cancellation", *IEEE Trans. on Communications*, vol. 44, pp. 817–825, July 1996.

[9] P. J. W. Melsa, R. C. Younce, and C. E. Rohrs, "Impulse response shortening for discrete multitone transceivers", *IEEE Trans. on Communications*, vol. 44, pp. 1662–1672, Dec. 1996.

[10] G. Cherubini, "Optimum upstream power back-off and multiuser detection for VDSL", in *Proc. GLOBECOM '01*, San Antonio, TX, Nov. 2001.

[11] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous code-division multiple-access channel", *IEEE Trans. on Communications*, vol. 41, pp. 285–290, Feb. 1993.

[12] "Access transmission systems on metallic access cables; Very high speed Digital Subscriber Line (VDSL); Part 2: Transceiver specification", *ETSI Technical Specification 101 270-2 V1.1.1*, May 2000.

[13] S. Kirkpatrik, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing approach", *Science*, vol. 220, pp. 671–680, May 1983.

[14] D. Vanderbilt and S. Louie, "A Monte Carlo simulated annealing approach to optimization over continuous variables", *J. Comp. Phys.*, vol. 56, pp. 259–271, 1984.

[15] "Constant average log: robust new power back-off method", *Contribution D.815 (WP1/15), ITU-T SG 15, Question 4/15*, Apr. 1999.

[16] "ETSI VDSL specifications (Part 1) functional requirements", *Contribution D.535 (WP1/15), ITU-T SG 15, Question 4/15*. June 21 V- July 2, 1999.

[17] T. Starr, J. M. Cioffi, and P. J. Silverman, *Digital subscriber line technology*. Upper Saddle River, NJ: Prentice-Hall, 1999.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 17

Wired and wireless network technologies

Wired and wireless network technologies allow users to obtain services that are offered by various providers, for example, telephony, Internet access at a rate of several Megabit per second, on demand television programs, and telecommuting. Links make use of channels that allow full duplex transmission and exhibit sufficient capacity in the two directions of transmission. Wired network technologies make use of the existing cable infrastructure, consisting of 1) UTP cables originally laid in the customer service area (local loops) for access to the *public switched telephone network* (PSTN) over a telephone channel with band limited to about 4 kHz; 2) UTP cables installed in buildings or small geographic areas for the links between stations of a *local-area network* (LAN); and 3) coaxial cables currently used to distribute analog TV signals via cable TV networks.

Wireless network technologies have gained widespread popularity among users; for example, technologies that allow the mobility of users are *digital-enhanced cordless telecommunications* (DECT) (see Chapter 18), *personal access communications systems* (PACS), and *universal mobile telecommunications systems* (UMTS). Wireless LANs have been developed to extend or replace wired LANs in environments where portable network stations are needed, allowing users to freely move around. The *multichannel multipoint distribution service* (MMDS) and the *local multipoint distribution service* (LMDS) are proposed as "wireless cable" networks that can offer high transmission capacity in geographical areas where the terrain does not present obstacles to the propagation of microwave radio signals. Hybrid structures are also considered, where the downstream transmission from a central office to the user and the upstream transmission in the opposite direction are performed using different transmission media, for example, services provided by *direct broadcast satellite* (DBS) in combination with upstream transmission over the PSTN.

17.1 Wired network technologies

17.1.1 Transmission over unshielded twisted pairs in the customer service area

Modem

The word "modem" is the contraction of *mod*ulator-*dem*odulator. The function of a modem is to convert a data signal into an analog passband signal that can be transmitted over the

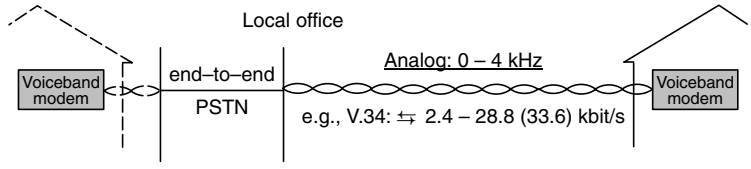


Figure 17.1. Illustration of a link between voiceband modems over the PSTN.

PSTN. A modem is defined *full duplex* if it can transmit and receive simultaneously over the same telephone channel, or *half duplex* in the other case (see Section 6.13.1 on page 522). A link between voiceband modems over the PSTN is illustrated in Figure 17.1.

One of the technologies developed for data transmission over the PSTN is described by the ITU-T standard V.34. The V.34 modem uses a QAM scheme with 4-dimensional TCM (see Chapter 12) and flexible precoding for transmission over channels with ISI (see Chapter 13). We recall that flexible precoding permits achieving coding and shaping gains with minimum transmit power penalty for arbitrary constellations, provided that the transfer function of the overall discrete time system does not have zeros on the unit circle. Among the innovations introduced by the V.34 modem, we recall the initial probing of the channel, during which the passband is determined; based on the probing results, the symbol rate, which is in the range from 2400 to 3429 Baud, and the carrier frequency are defined. The information bit rate for the standard version V.34bis is in the range from 2.4 to 33.6 kbit/s; the selected value depends on the symbol rate and on the channel signal-to-noise ratio.

The assumption that the link between two modems in the PSTN is completely "analog" does not lead to an accurate model for the majority of telephone channels available today; in fact the same network is essentially digital and transports signals that represent speech or data with a bit rate of 64 kbit/s. If modems used by service providers are linked to the PSTN by a digital channel that allows data transmission at a bit rate of 64 kbit/s, then only one "analog" local loop is found between the user and the rest of the network, as illustrated in Figure 17.2. Modems designed for the channel model just described are usually called *PCM modems* and can transmit data at a bit rate of 40–56 kbit/s over channels with a frequency band that goes from 0 Hz to about 4 kHz (see Section 17.1.1). This technology is described by the ITU-T standard X.90.

In Table 17.1 the characteristics of some modems are briefly described. For full duplex modems *frequency division duplexing* (FDD) or echo cancellation (EC) is used. The column labelled "ITU-T std" identifies the international standards defined by the *International Telecommunications Union—Telecommunications Standardization Sector*. The acronym TC stands for *trellis coding* (see Chapter 12); for a description of the various modulation techniques we refer to Chapter 6.

Digital subscriber line

In customer service areas UTP cables also represent a low cost alternative to optical fibers for links that allow data transmission at a considerably higher bit rates than those achievable by modems for transmission over the telephone channel, over distances that can reach 6 km; in fact, although optical fibers have substantially better transmission characteristics

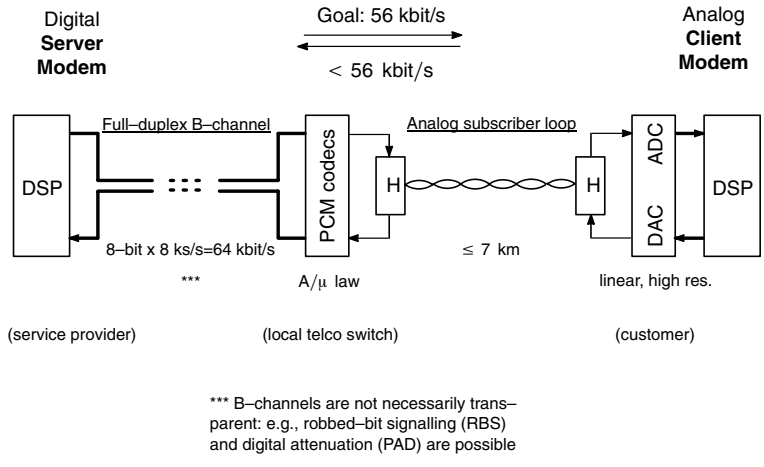


Figure 17.2. Illustration of a link between the PCM modem and the server modem.

Table 17.1 Commercial modems.

Bit-rate (bit/s)	Symbol rate	Duplex (method)	ITU-T std	Modulation
≤300	≤300	full (FDD)	V.21	2-FSK
1200	1200	half	V.23	2-FSK
1200	600	full (FDD)	V.22	4-PSK
2400	1200	half	V.26	4-PSK
2400	600	full (FDD)	V.22bis	16-QAM
2400	1200	full (EC)	V.26ter	4-PSK
4800	1600	half	V.27	8-PSK
4800	2400	full (EC)	V.32	4-QPSK
9600	2400	half	V.29	16-AM/PM
9600	2400	full (EC)	V.32	32-QAM + TC
14400	2400	full (EC)	V.32bis	128-QAM + TC
≤28800	≤3429	full (EC)	V.fast (V.34)	1024-QAM + TC
≤56000	<4000	full (EC)	X.90	PAM + TC

(see Chapter 4), a reliable link over a local loop is preferable in many cases given the large number of already installed cables [1, 2, 3]. Figure 17.3 illustrates a link between two *integrated services digital network* (ISDN) modems for two rates: basic rate (BR) at 160 kbit/s and primary rate (PR) at 1.544 or 2.048 Mbit/s.

The various *digital subscriber line* (DSL) technologies (in short xDSL) allow full duplex transmission between user and central office at bit rates that may be different in the two directions [4, 5]. For example, the *high bit rate digital subscriber line* (HDSL) offers a solution for full duplex transmission at a bit rate of 1.544 Mbit/s, also called T1 rate

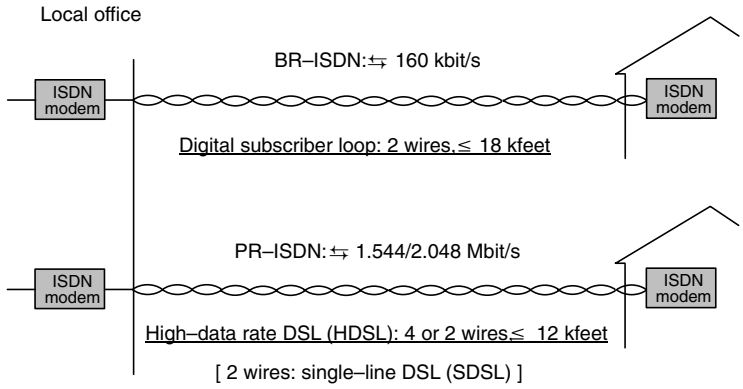


Figure 17.3. Illustration of links between ISDN modems on the subscriber line (BR = basic rate and PR = primary rate).

(see Section 6.13), over two twisted pairs and up to distances of 4500 m; the *single-line high-speed DSL* (SHDSL) provides full duplex transmission at rates up to 2.32 Mbit/s over a single twisted pair, up to distances of 2000 m.

A third example is given by the *asymmetric digital subscriber line* (ADSL) technology (see Figure 17.4); originally ADSL was proposed for the transmission of video-on-demand signals; later it emerged as a technology capable of providing a large number of services. For example, ADSL-3 is designed for downstream transmission of four compressed video signals, each having a bit rate of 1.5 Mbit/s, in addition to the full duplex transmission of a signal with a bit rate of 384 kbit/s, a control signal with a bit rate of 16 kbit/s, and an analog telephone signal, up to distances of 3600 m.

A further example is given by the *very high-speed DSL* (VDSL) technology, mainly designed for the *fiber-to-the-curb* (FTTC) architecture. The considered data rates are up to 26 Mbit/s downstream i.e. from the central office or optical network unit to the remote terminal, and 4.8 Mbit/s upstream for asymmetric transmission, and up to 14 Mbit/s for symmetric transmission, up to distances not exceeding a few hundred meters. Figure 17.5 illustrates the FTTC architecture, where the link between the user and an *optical network unit* (ONU) is obtained by a UTP cable with maximum length of 300 m, and the link between the ONU and a local central office is realized by optical fiber; in the figure, links between the user and the ONU that are realized by coaxial cable or optical fiber are also indicated, as well as the direct link between the user and the local central office using optical fiber with a *fiber-to-the-home* (FTTH) architecture.

Different baseband and passband modulation techniques are considered for high speed transmission over UTP cables in the customer service area. For example, the Study Group T1E1.4 of *Committee T1* chose 2B1Q quaternary PAM modulation (see Example 6.5.1 on page 479) for HDSL, and DMT modulation (see Chapter 9) for ADSL. Among the organizations that deal with the standardization of DSL technologies we also mention the Study Group TM6 of the *European Telecommunications Standard Institute* (ETSI) and the Study Group 15 of the ITU-T. Table 17.2 summarizes the characteristics of DSL technologies; spectral allocations of the various signals are illustrated in Figure 17.6.

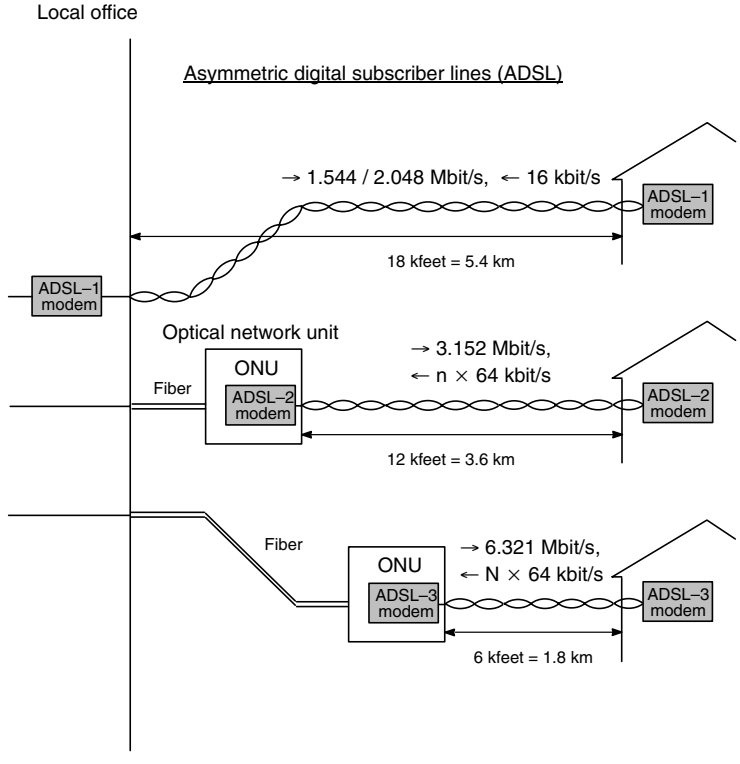


Figure 17.4. Illustration of links between ADSL modems over the subscriber line.

We now discuss more in detail the VDSL technology. Reliable and cost effective VDSL transmission at a few tens of Megabit per second is made possible by the use of frequency-division duplexing (FDD) (see Section 6.13), which avoids signal disturbance by near-end cross-talk (NEXT), a particularly harmful form of interference at VDSL transmission frequencies. Ideally, using FDD, transmissions on neighboring pairs within a cable binder couple only through far-end cross-talk (FEXT) (see also Section 16.4), the level of which is significantly below that of NEXT. In practice, however, other forms of signal coupling come into play because upstream and downstream transmissions are placed spectrally as close as possible to each other in order to avoid wasting useful spectrum. Closely packed transmission bands exacerbate interband interference by echo and NEXT from similar systems (self-NEXT), possibly leading to severe performance degradation. Fortunately, it is possible to design modulation schemes that make efficient use of the available spectrum and simultaneously achieve a sufficient degree of separation between transmissions in opposite directions by relying solely on digital signal processing techniques. This form of FDD is sometimes referred to as digital duplexing.

The concept of "divide and conquer" has been used many times to facilitate the solution of very complex problems; therefore it appears unavoidable that digital duplexing for VDSL will be realized by the sophisticated version of this concept represented by multicarrier transmission, although single carrier methods have been also proposed. As discussed in

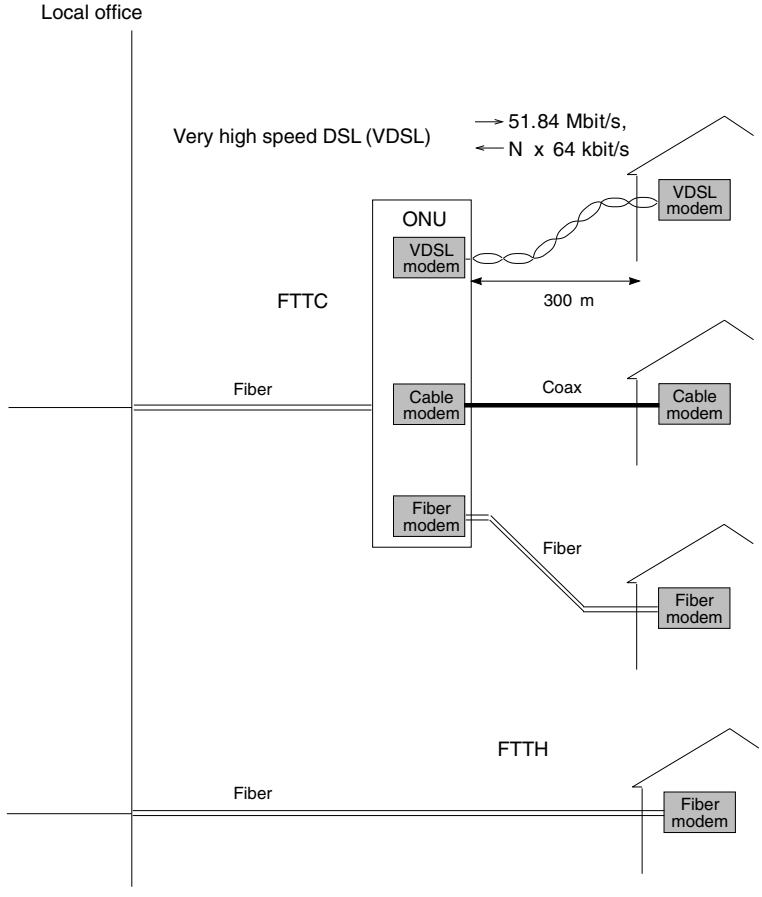


Figure 17.5. Illustration of FTTC and FTTH architectures.

Table 17.2 Characteristics of DSL technologies.

Acronym	Standard	Modulation	Bit rate (Mbit/s)	Distance (m)
basic rate ISDN		2B1Q	0.144	$\leq$6000
HDSL	G.991.1	2B1Q	1.544, 2.048	$\leq$4000
SHDSL	G.shdsl	TC-PAM	0.192÷2.32	$\leq$2000
ADSL	G.992.1	DMT	*downstream* 6.144 *upstream* 0.640	$\leq$3600
ADSL lite	G.992.1	DMT	*downstream* 1.5 *upstream* 0.512	*best effort service*
VDSL			*downstream* $\leq$ 26 *upstream* $\leq$ 14	$\leq$1500

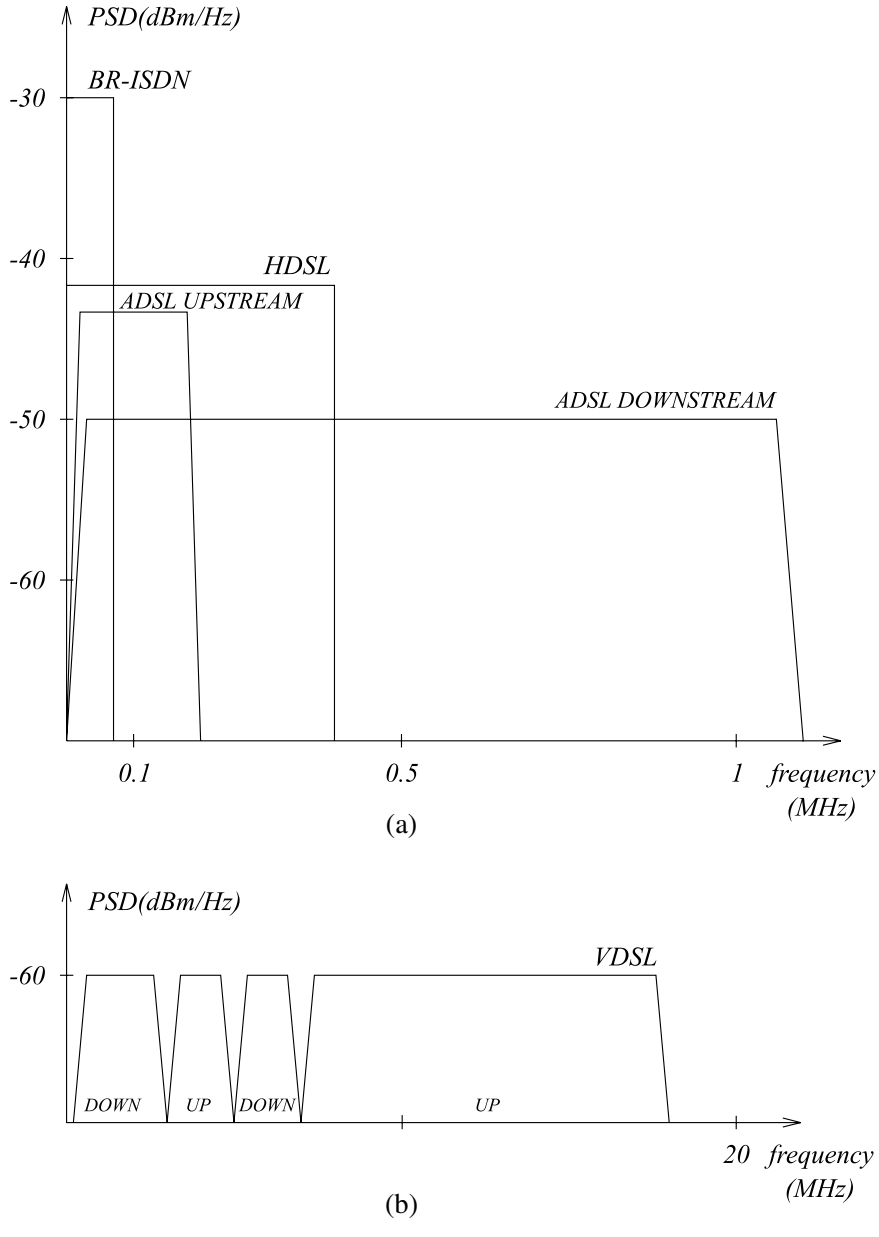


Figure 17.6. Spectral allocation of signals for xDSL technologies.

Section 9.5, various variants of multicarrier transmission exist. The digital duplexing method for VDSL known as Zipper [6] is based on discrete-multitone (DMT) modulation; here we consider *filtered multitone* (FMT) modulation (see Section 9.5), which involves a different set of trade-offs for achieving digital duplexing in VDSL and offers system as well as performance advantages over DMT [7, 8].

The key advantages of FMT modulation for VDSL can be summarized as follows. First, there is flexibility to adapt to a variety of spectrum plans for allocating bandwidth for upstream and downstream transmission by proper assignment of the subchannels. This feature is also provided by DMT modulation, but not as easily by single-carrier modulation systems. Second, FMT modulation allows a high-level of subchannel spectral containment and thereby avoids disturbance by echo and self-NEXT. Furthermore, disturbance by a narrowband interferer, e.g., from AM or HAM radio sources, does not affect neighboring subchannels as the side lobe filter characteristics are significantly attenuated. Third, FMT modulation does not require synchronization of the transmissions at both ends of a link or at the binder level, as is sometimes needed for DMT modulation.

As an example of system performance, we consider the bit rate achievable for different values of the length of a twisted pair, assuming symmetrical transmission at 22.08 MBaud and full duplex transmission by FDD based on FMT modulation. The channel model is obtained by considering a line with attenuation equal to 11.1 dB/100 m at 11.04 MHz (see Section 4.4.1), with 49 near-end cross-talk interference signals, 49 far-end cross-talk interference signals, and additive white Gaussian noise with a PSD of -140 dBm/Hz. The transmitted signal power is assumed equal to 10 dBm. The FMT system considered here employs the same linear-phase prototype filter for the realization of transmit and receive polyphase filter banks, designed for $\mathcal{M} = 256$, $K = 288$, and $\gamma = 10$; we recall that with these values of $\mathcal{M}$ and K the excess bandwidth within each subchannel is equal to 12.5%. Per-subchannel equalization is obtained by a Tomlinson–Harashima precoder (see Section 13.3.1) with 9 coefficients at the transmitter and a fractionally spaced linear equalizer with 26 $T/2$ spaced coefficients at the receiver. With these parameter values, and using the bit loading technique of Section 13.2, the system achieves bit rates of 24.9 Mbit/s, 10.3 Mbit/s, and 6.5 Mbit/s for the three lengths of 300 m, 1000 m, and 1400 m, respectively [9]. We refer to Section 16.4 for a description of the general case where users are connected at different distances from the central office and power control is applied.

17.1.2 High speed transmission over unshielded twisted pairs in local area networks

High speed transmission over UTP cables installed in buildings is studied by different standardization organizations. For example, the ATM Forum and the Study Group 13 of the ITU-T consider transmission at bit rates of 155.52 Mbit/s and above for the definition of the *asynchronous transfer mode* (ATM) interface between the user and the network. The IEEE 802.3 Working Group investigates the transmission at 1 Gbit/s over four twisted pairs, type UTP-5, for Ethernet (1000BASE-T) networks.

UTP cables were classified by the EIA/TIA according to the transmission characteristics (see Chapter 4). We recall that UTP-3, or voice-grade, cables exhibit a signal attenuation and a cross-talk coupling much greater than that of UTP-5, or data-grade, cables. For LAN applications, the maximum cable length for a link between stations is 100 m. Existing cabling systems use bundles of twisted pairs, usually 4 or 25 pairs, and signals may cross line discontinuities, represented by connectors. For transmission over UTP-3 cables, in

order to meet limits on emitted radiation, the signal band must be confined to frequencies below 30 MHz and sophisticated signal processing techniques are required to obtain reliable transmission.

Standards for Ethernet networks that use the *carrier sense multiple access with collision detection* (CSMA/CD) protocol are specified by the IEEE 802.3 Working Group for different transmission media and bit rates. With the CSMA/CD protocol, a station can transmit a data packet only if no signal from other stations is being transmitted on the transmission medium. As the probability of collision between messages cannot be equal to zero because of the signal propagation delay, a transmitting station must continuously monitor the channel; in the case of a collision, it transmits a special signal called a jam signal to inform the other stations of the event, and then stops transmission. Retransmission takes place after a random delay. The 10BASE-T standard for operations at 10 Mbit/s over two unshielded twisted pairs of category 3 or higher defines one of the most widely used implementations of Ethernet networks; this standard considers conventional *mono duplex* (see Section 16.1) transmission, where each pair is utilized to transmit only in one direction using simple Manchester line coding (see Appendix 7.A) to transmit data packets, as shown in Figure 17.7. Transmitters are not active outside of the packets transmission intervals, except for transmission of a signal called *link beat* that is occasionally sent to assure the link connection.

The request for transmission speeds higher than 10 Mbit/s motivated the IEEE 802.3 Working Group to define standards for fast Ethernet that maintain the CSMA/ CD protocol and allow transmission at 100 Mbit/s and above. For example, the 100BASE-FX standard defines a physical layer (PHY) for Ethernet networks over optical fibers. The 100BASE-TX standard instead considers conventional mono duplex transmission over two twisted pairs of category 5; the bit rate of 100 Mbit/s is obtained by transmission with a modulation rate of 125 MBaud and multilevel transmission (MLT-3) line coding combined with a channel code with rate 4/5 and scrambling, as illustrated in Figure 17.8. We also mention the 100BASE-T4 standard, which considers transmission over four twisted pairs of category 3; the bit rate of 100 Mbit/s is obtained in the following way by using an 8B6T code with a

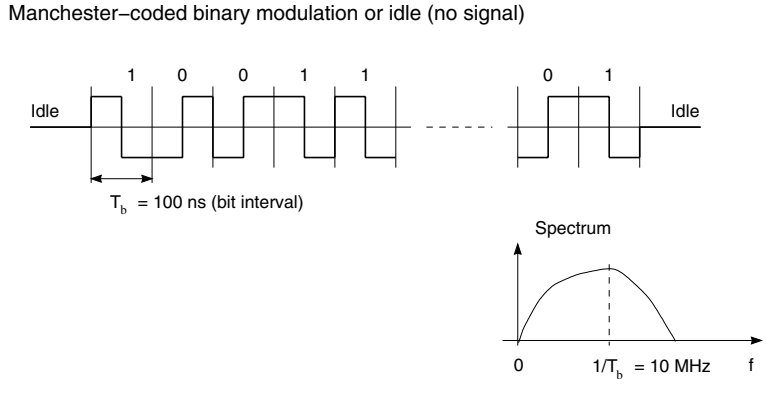


Figure 17.7. Illustration of 10BASE-T signal characteristics.

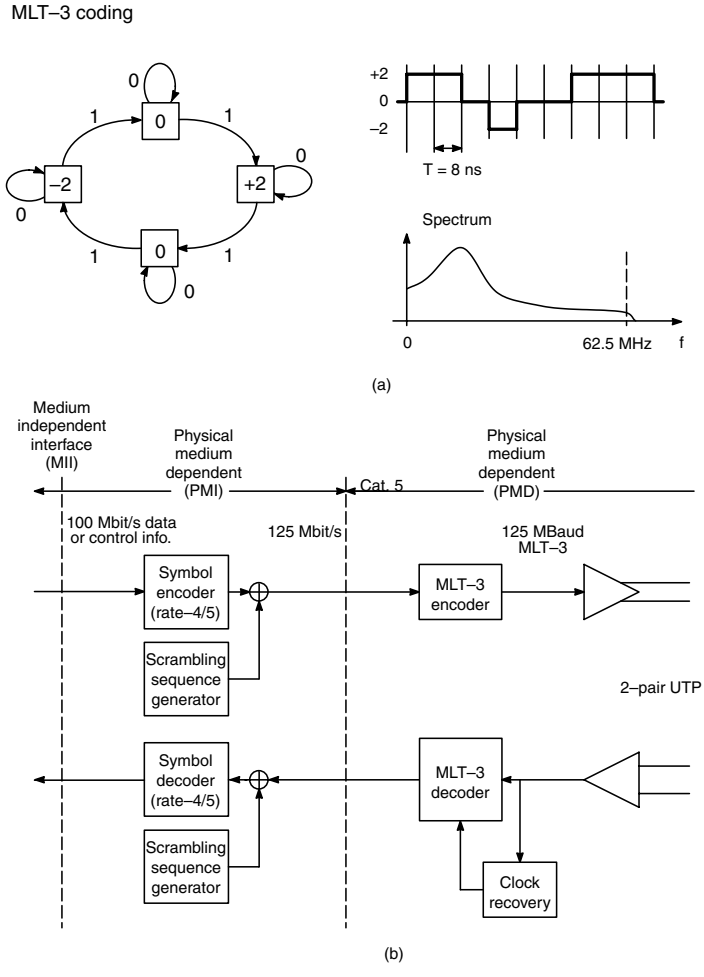


Figure 17.8. Illustration of (a) 100BASE-TX signal characteristics and (b) 100BASE-TX transceiver block diagram.

modulation rate equal to 25 MBaud: on the first two pairs data are transmitted at a bit rate of 33.3 Mbit/s in half duplex fashion, while on the two remaining pairs data are transmitted at 33.3 Mbit/s in mono duplex fashion.

A further version of fast Ethernet is represented by the 100BASE-T2 standard, which allows users of the 10BASE-T technology to increase the bit rate from 10 to 100 Mbit/s without modifying the cabling from category 3 to 5, or using four pairs for a link over UTP-3 cables. The bit rate of 100 Mbit/s is achieved with *dual duplex* transmission over two twisted pairs of category 3, where each pair is used to transmit in the two directions (see Figure 17.9). The 100BASE-T2 standard represents the most advanced technology for high

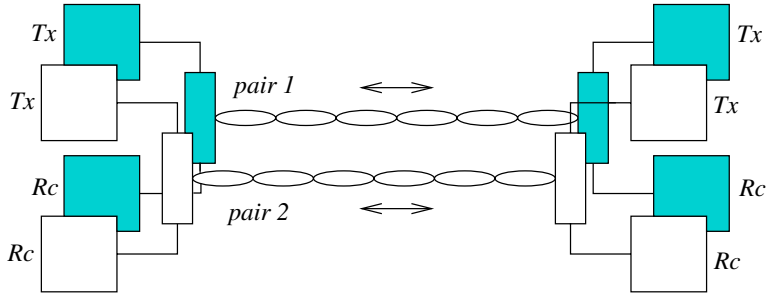


Figure 17.9. Illustration of a dual duplex transmission system.

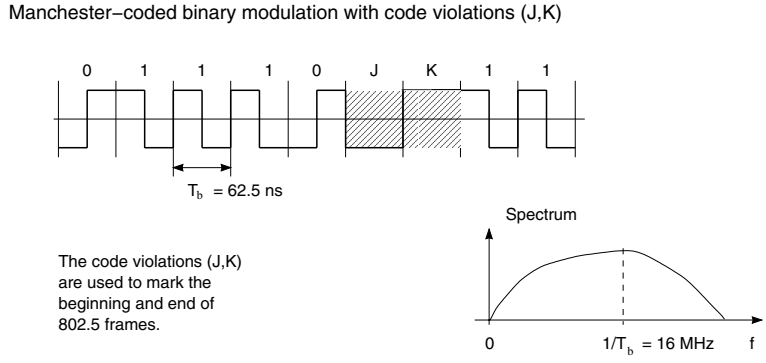


Figure 17.10. Illustration of signal characteristics for transmission over token ring networks.

speed transmission over UTP-3 cables in LANs; the transceiver design for 100BASE-T2 will be illustrated in Chapter 19.

Other important examples of LANs are the *token ring* and the *fiber distributed data interface* (FDDI) networks; standards for token ring networks are specified by the IEEE 802.5 Working Group, and standards for FDDI networks are specified by ANSI. In these networks the access protocol is based on the circulation of a token. A station is allowed to transmit a data packet only after having received the token; once the transmission has been completed, the token is passed to the next station. The IEEE 802.5 standard specifies operations at 16 Mbit/s over two unshielded twisted pairs of category 3 or higher, with mono duplex transmission. For the transmission of data packets Manchester line coding is adopted; the token is indicated by coding violations, as illustrated in Figure 17.10. The ANSI standard for FDDI networks specifies a physical layer for mono duplex transmission at 100 Mbit/s over two unshielded twisted pairs of category 5 identical to that adopted for the Ethernet 100BASE-TX standard.

Table 17.3 summarizes the characteristics of existing standards for high speed transmission over UTP cables.

Table 17.3 Scheme summarizing characteristics of standards for high speed transmission over UTP cables.

Acronym	Standard	Bit rate (Mbit/s)	Cable type
"Legacy LANs"			
10BASE-T Ethernet	IEEE 802.3	10	2-pair UTP-3
16TR Token Ring	IEEE 802.5	16	2-pair UTP-3
Fiber Distributed Data Interface			
FDDI	ANSI X3T9.5	100	2-pair UTP-5
Fast Ethernet			
100BASE-TX	IEEE 802.3	100	2-pair UTP-5
100BASE-T4	IEEE 802.3	100	4-pair UTP-3
100BASE-T2	IEEE 802.3	100	2-pair UTP-3
1000BASE-T4	IEEE 802.3	1000	4-pair UTP-5
AnyLAN			
100 VG	IEEE 802.12	100	4-pair UTP-3
ATM User-Network Interfaces			
ATM-25	ATM Forum	25.6	2-pair UTP-3
ATM-51	ATM Forum	51.84	2-pair UTP-3
ATM-100	ATM Forum	100	2-pair UTP-5
ATM-155*	ATM Forum	155.52	2-pair UTP-3

*Cannot operate over UTP-3 in the presence of *alien-NEXT* interference.

17.1.3 Hybrid fiber/coaxial cable networks

A *hybrid fiber/coax* (HFC) network is a multiple-access network, in which a *head-end controller* (HC) broadcasts data and information for *medium-access control* (MAC) over a set of channels in the downstream direction to a certain number of stations, and these stations send information to the HC over a set of shared channels in the upstream direction [10, 11]. The topology of an HFC network is illustrated in Figure 17.11: an HFC network is a point-to-multipoint, tree and branch access network in the downlink, with downstream frequencies in the 50–860 MHz band, and a multipoint-to-point, bus access network in the uplink, with upstream frequencies in the 5–42 MHz band. Examples of frequency allocations are shown in Figure 17.12. The maximum round-trip delay between the HC and a station is of the order of 1 ms. The IEEE 802.14 Working Group is one of the standardization bodies working on the specifications for the PHY and MAC layers of HFC networks. A set of PHY and MAC layer specifications adopted by North American cable operators is described in [12].

In the downstream direction transmission takes place in broadcast made over channels with bandwidth equal to 6 or 8 MHz, characterized by low distortion and high signal-to-noise ratio, typically $\geq$42 dB. The J.83 document of the ITU-T defines two QAM transmission schemes

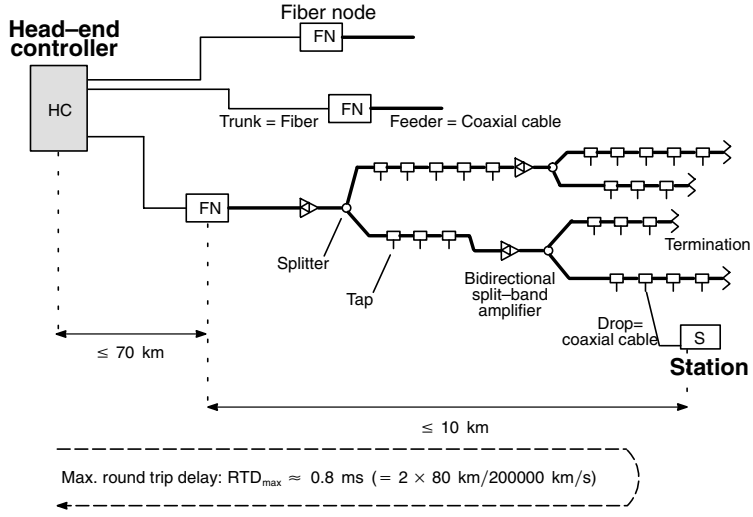


Figure 17.11. Illustration of the HFC network topology.

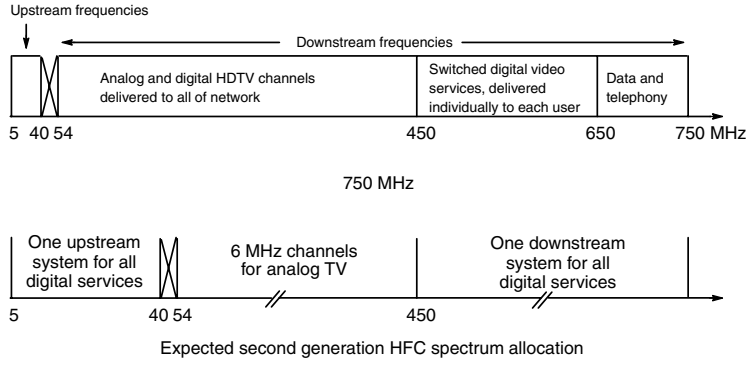


Figure 17.12. Examples of frequency allocations in HFC networks.

for transmission in the downstream direction, with a bit rate in the range from 30 to 45 Mbit/s [13]; by these schemes spectral efficiencies of 5–8 bit/s/Hz are therefore obtained.

In the upstream direction the implementation of PHY and MAC layers is considerably more difficult than in the downstream. In fact, we can make the following observations:

- signals are transmitted in bursts from the stations to the HC; therefore it is necessary for the HC receiver to implement fast synchronization algorithms;
- signals from individual stations must be received by the HC at well-defined instants of arrival and power levels; therefore, procedures are required for the determination

of the round-trip delay between the HC and each station, as well as for the control of the power of the signal transmitted by each station, as channel attenuation in the upstream direction may present considerable variations, of the order of 60 dB;

- the upstream channel is usually disturbed by impulse noise and narrowband interference signals; moreover, the distortion level is much higher than in the downstream channel.

Interference signals in the upstream channel are caused by domestic appliances and HF radio stations; these signals accumulate along the paths from the stations to the HC and exhibit time-varying characteristics; they are usually called *ingress noise*. Because of the high level of disturbance signals, the spectral efficiency of the upstream transmission is limited to 2–4 bit/s/Hz.

The noise spectrum suggests that the upstream transmission is characterized by the possibility of changing the frequency band of the transmitted signal (*frequency agility*), and of selecting different modulation rates and spectral efficiencies. In [12], a QAM scheme for upstream transmission that uses a 4 or 16 point constellation and a maximum modulation rate of 2.56 MBaud is defined; the carrier frequency, modulation rate and spectral efficiency are selected by the HC and transmitted to the stations as MAC information.

Ranging and power adjustment for uplink transmission

We now describe the registration procedure of a cable modem at the HC. The time base relative to the upstream transmission is divided into intervals by the mechanism adopted by the HC for the allocation of resources to the stations. Each interval is constituted by an integer number of subintervals usually called *mini-slots*; a mini-slot then represents the smallest time interval for the definition of an opportunity for upstream transmission.

In [12], a TDMA scheme is considered where uplink transmission is divided into a stream of mini-slots. Each mini-slot is numbered relative to a master reference clock maintained by the HC. The HC distributes timing information to the cable modems by means of time synchronization messages, which include time stamps. From these time stamps, the stations establish a local time base locked to the time base of the HC. For uplink transmission, access to the mini-slots is controlled by *allocation map* (MAP) messages, which describe transmission opportunities on available uplink channels. A MAP message includes a variable number of *information elements* (IE), each of which defines the modality of access to a range of mini-slots in an uplink channel, as illustrated in Figure 17.13. Each station has a unique address of 48 bits; with each active station is also associated, for service request, a 14-bit *service identifier* (SID).

At the beginning of the registration procedures, a station tunes its receiver to the downstream channel on which it receives SYNC messages from the HC; the acquired local timing is delayed with respect to the HC timing due to the signal propagation delay. The station monitors the downstream channel until it receives a MAP message with an IE of initial maintenance, which specifies the time interval during which new stations may send a *ranging request* (RNG-REQ) message to join the network. The duration of the initial

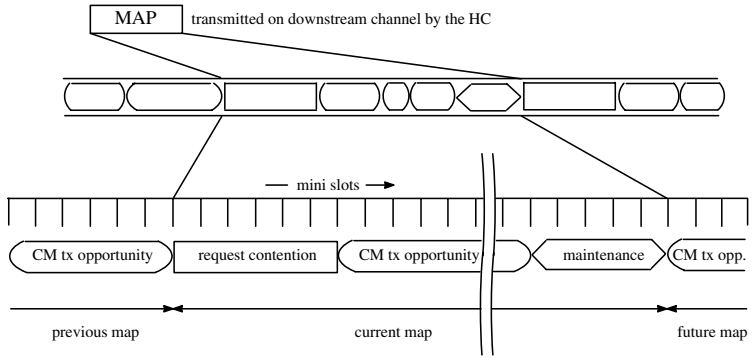


Figure 17.13. Example of a MAP message [12]. [*Reproduced with permission of Cable Television Laboratories, Inc.*]

maintenance interval is equivalent to the maximum round-trip delay plus the transmission time of a RNG-REQ message. At the instant specified in the MAP message, the station sends a first RNG-REQ message using the lowest power level of the transmitted signal, and is identified by a SID equal to zero as a station that requests to join the network.

If the station does not receive a response within a pre-established time, it means that a collision occurred between RNG-REQ messages sent by more than one station, or that the power level of the transmitted signal was too low; to reduce the probability of repeated collisions, a collision resolution protocol is used with random back-off. After the back-off time interval, of random duration, is elapsed, the station waits for a new MAP message containing an IE of initial maintenance and at the specified instant retransmits a RNG-REQ message with a higher power level of the transmitted signal. These steps are repeated until the HC detects a RNG-REQ message, from which it can determine the round-trip delay and the correction of the power level that the station must apply for future transmissions. In particular, the compensation for the round-trip delay is computed so that, once applied, the transmitted signals from the station arrive at the HC at well-defined time instants. Then the HC sends to the station, in a *ranging response* RNG-RSP message, the information on round-trip delay compensation and power level correction to be used for future transmissions; this message also includes a temporary SID.

The station waits for a MAP message containing an IE of station maintenance, individually addressed to it by its temporary SID, and in turn responds through a RNG-REQ message, signing it with its temporary SID and using the specified round-trip delay compensation and power level correction; next, the HC sends another RNG-REQ message to the station with information for a further refinement of round-trip delay compensation and power level correction. The steps of ranging request/response are repeated until the HC sends a ranging successful message; at this point the station can send a registration request

(REG-REQ) message, to which the HC responds with a registration response (REG-RSP) message confirming the registration and specifying one or more SID that the cable modem must use during the following transmissions. Finite state machines for the ranging procedure and the regulation of the power level for the cable modem and for the HC are shown in Figures 17.14 and 17.15, respectively.

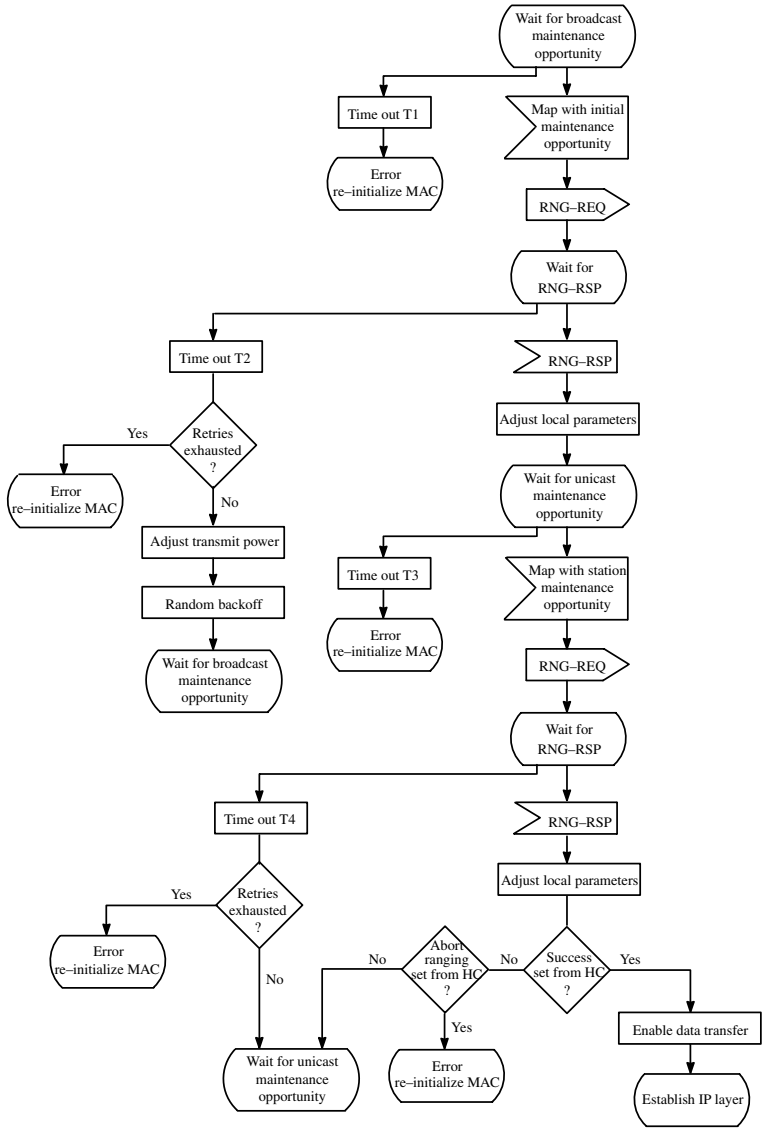


Figure 17.14. Finite state machine used by the cable modem for the registration procedure [12]. [*Reproduced with permission of Cable Television Laboratories, Inc.*]

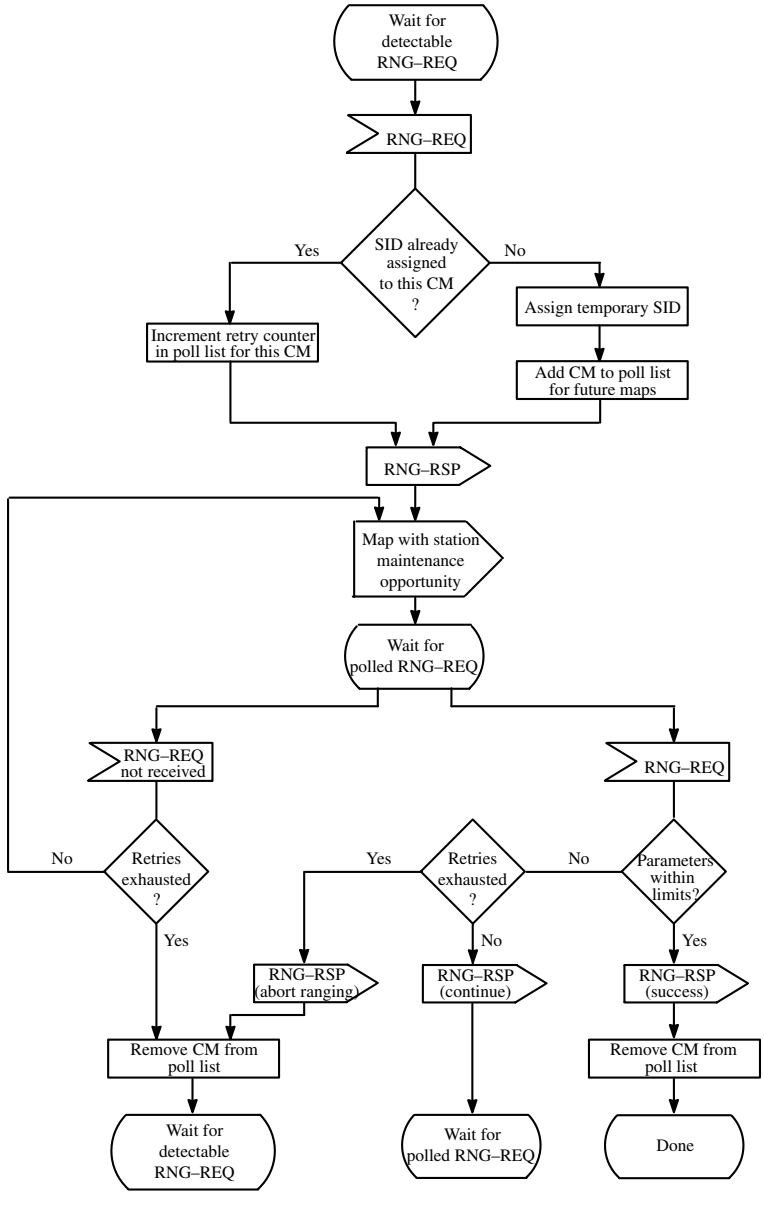


Figure 17.15. Finite state machine used by the HC for the registration procedure [12]. [*Reproduced with permission of Cable Television Laboratories, Inc.*]

17.2 Wireless network technologies

Mobile radio technologies used to provide personal communication services to the end user are commonly grouped into high tier and low tier; the main characteristics of these

Table 17.4 Scheme summarizing the main characteristics of high tier and low tier technologies.

	High tier	Low tier
Base station	Large and expensive	Small and inexpensive
Coverage/base station	Up to several km	<500 m radius
Bit rate	Low to medium	Very high
Talk time for portables	Short (∼1 hour)	Long (>4 hours)
Vehicular speed	>100 km/h	∼50 km/h
Quality	<wireline	∼wireline
Principal application	Outdoor vehicular	Indoor/outdoor pedestrian
Typical examples	GSM, IS-95, UMTS, IMT-2000	DECT, PACS, PHS, Bluetooth

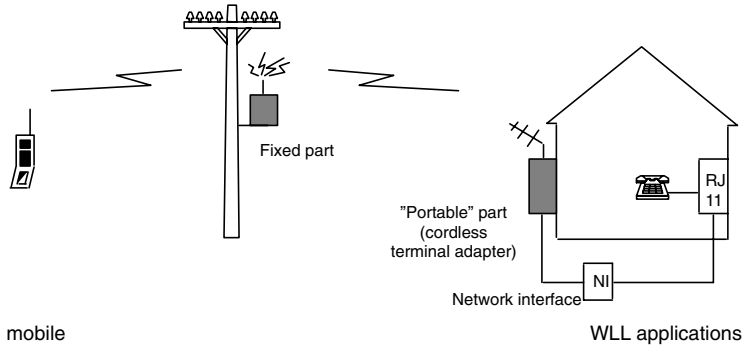


Figure 17.16. Illustration of the utilization of DECT in the wireless local loop.

technologies are summarized in Table 17.4. Appendix 17.A describes some of the widely adopted wireless technologies listed in Table 17.4.

Low tier technologies are designed to achieve a *quality of service* (QoS) similar to that offered by wired networks. DECT in Europe and PACS in the United States were developed as technologies for the *wireless local loop* (WLL) to provide communication services also to mobile users [14, 15]. DECT was originally developed for application in *wireless private branch exchanges* (wireless PBXs) with low user mobility, and later extended as a technology for the WLL, as illustrated in Figure 17.16. PACS is illustrated in Figure 17.17.

17.2.1 Wireless local area networks

For the transmission at bit rates of the order of 1–10 Mbit/s, wireless local area networks (WLANs) normally use the *industrial, scientific, and medical* (ISM) frequency bands defined by the United States *Federal Communications Commission* (FCC), that is 902–928 MHz,

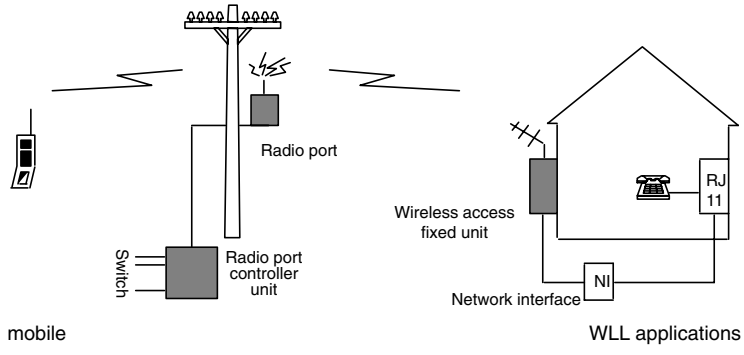


Figure 17.17. Illustration of the utilization of PACS in the wireless local loop.

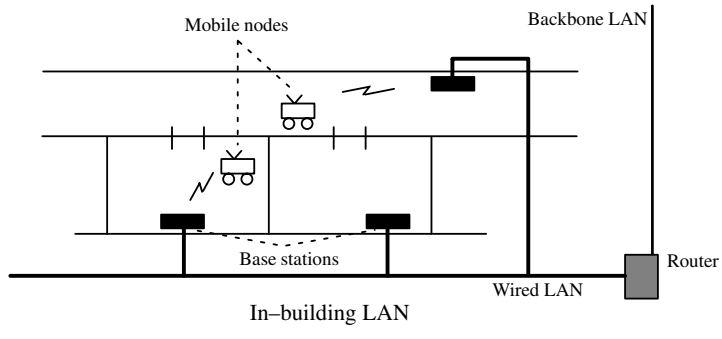


Figure 17.18. Illustration of the configuration of a wireless LAN inside a building.

2400–2483.5 MHz, and 5725–5850 MHz; these networks are also called RF WLANs to distinguish them from the IR WLANs that use the *infrared* (IR) frequency band. Specifications for PHY and MAC layers of WLANs are developed by various standardization organizations, among which we cite the IEEE 802.11 Working Group and the *European Telecommunications Standard Institute* (ETSI).

The region within which mobile stations have the possibility of exchanging information with the network is divided into cells (see Figure 17.24). To reduce interference, neighboring cells use different frequencies. Within each cell, an access station or base station allocates frequencies and guarantees to the mobile stations the possibility of accessing fixed networks over metallic cables or optical fibers (see Figure 17.18).

In the United States WLANs are allowed to operate in the IMS frequency bands without needing a license from the FCC, which, however, sets restrictions on the power of the radio signal that must be less than 1 W and specifies that spread-spectrum technology (see Chapter 10) must be used whenever the signal power is larger than 50 mW. Most RF WLANs employ direct sequence or frequency hopping spread-spectrum systems;

WLANs that use narrowband modulation systems usually operate in the band around 5.8 GHz, with a transmitted signal power lower than 50 mW, in compliance with FCC regulations. The coverage radius of RF WLANs is typically of the order of a few hundred meters.

IR WLANs usually employ PAM-DSB modulation (see Appendix 7.C) with values of the carrier frequency of the order of the frequency of the infrared radiation, as also typically adopted in a fiber optic link; this system allows a simple and low cost implementation of the PHY layer [16]. However, it is necessary to consider that non-negligible interference signals can be generated by infrared radiation sources, like the sun or lighting systems. Two methods are usually considered in designing IR WLANs. In the first method the signal is transmitted along a well-defined direction; in this case an IR system can also be used outdoor with a coverage radius of a few kilometers. In the second method, the signal is irradiated in all directions and the coverage radius is limited to about 10 m.

Medium access control protocols

Unlike cabled LANs, WLANs operate over channels with multipath fading, and channel characteristics typically vary over short distances. Channel monitoring to determine whether other stations are transmitting requires a larger time interval than that required by a similar operation in cabled LANs; this translates into an efficiency loss of the CSMA protocols, whenever they are used without modifications. The MAC layer specified by the IEEE 802.11 Working Group is based on the CSMA protocol with *collision avoidance* (CSMA/CA), in which four successive stages are foreseen for the transmission of a data packet, as illustrated in Figure 17.19 [17].

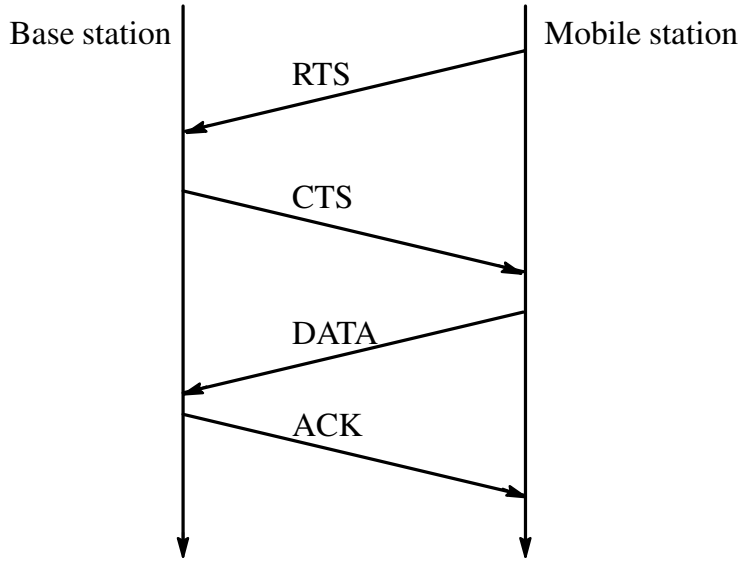


Figure 17.19. Illustration of the CSMA/CA protocol.

The CSMA/CA principle is simple. All mobile stations that have packets to transmit compete for channel access by sending *ready to transmit* (RTS) messages by a CSMA protocol. If the base stations recognize a RTS message sent by a mobile station, it sends a *clear to transmit* (CTS) message to the same mobile station and this one transmits the packet; if reception of this packet occurs correctly, then the base station sends an *acknowledgement* (ACK) message to the mobile station. With CSMA/CA, the only possibility of collision will occur during the RTS stage of the protocol; however, we note that also the efficiency of this protocol is reduced with respect to that of the simple CSMA/CD because of the presence of the RTS and CTS stages.

As mobility is allowed, the number of mobile stations that are found inside a cell can change at any instant. Therefore it is necessary that each station informs the others of its presence as it moves around. A protocol used to solve this problem is the so-called hand-off, or hand-over, protocol, which can be described as follows:

- a switching station, or all base stations with a coordinated operation, registers the information relative to the signal levels of all mobile stations inside each cell;
- if a mobile station M is serviced by base station B1, but the signal level of station M becomes larger if received by another base station B2, the switching station proceeds to a *hard hand-off* operation whose final result is that mobile station M is considered part of the cell covered by base station B2.

17.2.2 MMDS and LMDS

The multichannel multipoint distribution service is presently used in the United States to distribute TV signals for entertainment and education programming. It can offer a maximum number of 33 channels for analog TV in the band 2.150–2.686 GHz, and the cell radius is typically in the range from 40 to 60 km, according to the environment characteristics and antenna positioning. The local multipoint distribution service is also considered for wireless access. The proposed spectrum allocation for this system comprises the bands 27.5–28.35 GHz and 29.1–29.25 GHz, and the cell radius is typically limited to 8 km. The configuration of MMDS and LMDS networks is illustrated in Figure 17.20.

Assuming that most MMDS channels are dedicated to the distribution of TV signals, only a limited number of channels are available for downstream data transmission. An increase of system capacity can be obtained by methods of frequency reuse, for example "cellularization" or "sectorization". "Cellularization" consists of subdividing the area into cells and designing the system so that there is a certain spatial separation between cells using the same frequencies. On the other hand, "sectorization" consists of using multiple directional antennas at the base station, where each antenna is used to distribute information to a certain group of users; frequency reuse means sending different signals to different user groups using the same RF channels. This method requires that the various user groups are located in geographical areas characterized by a sufficient azimuth separation and that antennas have a good rejection characteristic of the secondary lobes.

An example of "sectorization" is illustrated in Figure 17.21, where six antennas are used; each antenna offers a 60-degree coverage, and to each antenna is assigned the same number of downstream channels. Moreover, two separate channel sets, labelled A and B,

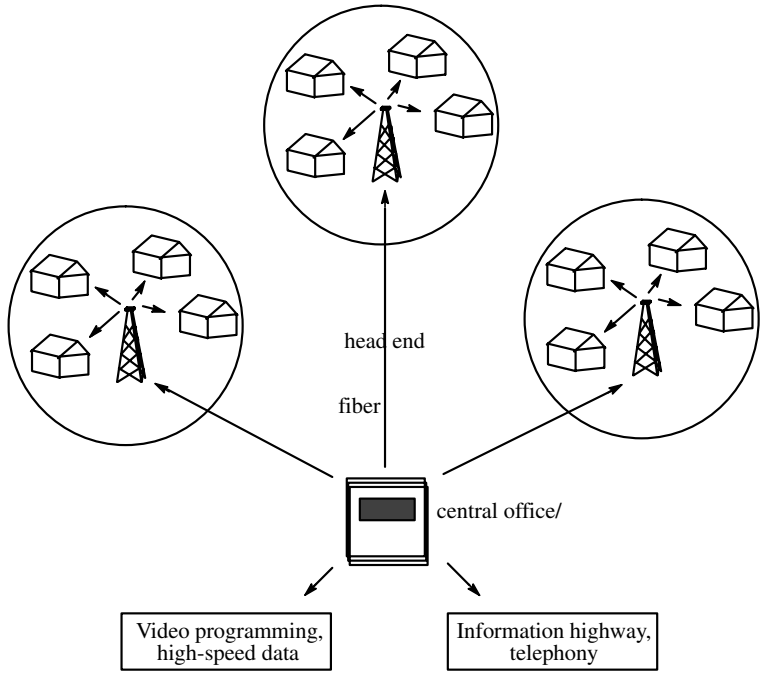


Figure 17.20. Illustration of the configuration of MMDS and LMDS networks.

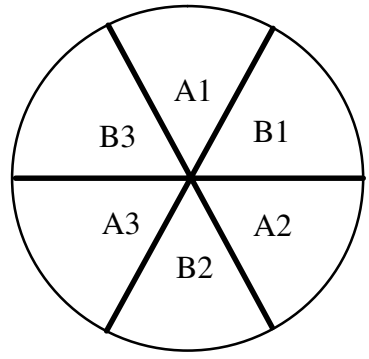


Figure 17.21. Example of sectorization using six antennas.

are considered; the two sets include the entire frequency band for downstream transmission and they are reused three times, thus tripling the system capacity with respect to the case without "sectorization". The required rejection level of secondary lobes can be reduced by the combination of "sectorization" with orthogonal polarization of adjacent sectors; we note, however, that orthogonal polarization can be effectively introduced only in the case of signal propagation in the absence of multipath, as in point-to-point links over short distances, using at each end of the link antennas placed at a large height with respect

to the ground. For residential area installations where roof top antennas are used, usually multipath components due to reflected signals from the ground and surrounding buildings are present (see Section 4.6); the reflected signals may have components with orthogonal polarity with respect to the desired signal and this phenomenon introduces interference.

In the design of transmission systems for LMDS it is necessary to consider the fact that, even if the transmitter and receiver are in line of sight, the effect of traffic and foliage movement determines a transmission channel with very hostile fading characteristics; for example, it is common to encounter in these environments a Doppler spread of 2 Hz or less, with a fade of over 40 dB. To obtain a reliable transmission for LMDS and to overcome effects due to the multipath, in addition to sectorized directional antenna systems various techniques may be considered, such as:

- signal coding and adaptive equalization,
- frequency diversity,
- combining of received signals by multiple antenna systems.

Bibliography

[1] S. V. Ahamed, P. L. Gruber, and J.-J. Werner, "Digital subscriber line (HDSL and ADSL) capacity of the outside loop plant", *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 1540–1549, Dec. 1995.

[2] W. Y. Chen and D. L. Waring, "Applicability of ADSL to support video dial tone in the copper loop", *IEEE Communications Magazine*, vol. 32, pp. 102–109, May 1994.

[3] G. T. Hawley, "Systems considerations for the use of xDSL technology for data access", *IEEE Communications Magazine*, vol. 35, pp. 56–60, Mar. 1997.

[4] W. Y. Chen and D. L. Waring, "Applicability of ADSL to support video dial tone in the copper loop", *IEEE Communications Magazine*, vol. 32, pp. 102–109, May 1994.

[5] T. Starr, J. M. Cioffi, and P. J. Silverman, *Understanding digital subscriber line technology*. Englewood Cliffs, NJ: Prentice-Hall, 2000.

[6] F. Sjoberg, M. Isaksson, R. Nilsson, P. Ödling, S. K. Wilson, and P. O. Borjesson, "Zipper: a duplex method for WDSL based on DMT", *IEEE Trans. on Communications*, vol. 47, pp. 1245–1252, Aug. 1999.

[7] G. Cherubini, E. Eleftheriou, S. Ölçer, and J. M. Cioffi, "Filter bank modulation techniques for very high-speed digital subscriber lines", *IEEE Communications Magazine*, vol. 38, pp. 98–104, May 2000.

[8] G. Cherubini, E. Eleftheriou, and S. Ölcer, "Filtered multitone modulation for very-high-speed digital subscriber lines", *IEEE Journal on Selected Areas in Communications*, June 2002.

[9] G. Cherubini, E. Eleftheriou, and S. Ölcer, "Filtered multitone modulation for VDSL", in *Proc. IEEE GLOBECOM '99*, Rio de Janeiro, Brazil, pp. 1139–1144, Dec. 1999.

[10] C. A. Eldering, N. Himayat, and F. M. Gardner, "CATV return path characterization for reliable communications", *IEEE Communications Magazine*, vol. 33, pp. 62–69, Aug. 1995.

[11] C. Bisdikian, K. Maruyama, D. I. Seidman, and D. N. Serpanos, "Cable access beyond the hype: on residential broadband data services over HFC networks", *IEEE Communications Magazine*, vol. 34, pp. 128–135, Nov. 1996.

[12] D. Fellows and D. Jones, "DOCSIS$^{(tm)}$", *IEEE Communications Magazine*, vol. 39, pp. 202–209, Mar. 2001.

[13] "Digital multi-programme systems for television sound and data services for cable distribution", ITU-T Recommendation J.83, ITU-T Study Group 9, Oct. 24 1995.

[14] W. Honcharenko, J. P. Kruys, D. Y. Lee, and N. J. Shah, "Broadband wireless access", *IEEE Communications Magazine*, vol. 35, pp. 20–27, Jan. 1997.

[15] C. C. Yu, D. Morton, C. Stumpf, R. G. White, J. E. Wilkes, and M. Ulema, "Low-tier wireless local loop radio systems. Part 1: Introduction", *IEEE Communications Magazine*, vol. 35, pp. 84–92, Mar. 1997.

[16] F. Gfeller and W. Hirt, "A robust wireless infrared system with channel reciprocity", *IEEE Communications Magazine*, vol. 36, pp. 100–106, Dec. 1998.

[17] S. Singh, "Wireless LANs", in *The Mobile Communications Handbook* (J. D. Gibson, ed.), ch. 34, pp. 540–552, New York: CRC/IEEE Press, 1996.

[18] R. O. LaMaire, A. Krishna, P. Bhagwat, and J. Panian, "Wireless LANs and mobile networking: standards and future directions", *IEEE Communications Magazine*, vol. 34, pp. 86–94, Aug. 1996.

[19] M. Rahnema, "Overview of the GSM system and protocol architecture", *IEEE Communications Magazine*, vol. 31, pp. 92–100, Apr. 1993.

[20] T. S. Rappaport, *Wireless communications: principles and practice*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[21] D. J. Goodman, *Wireless personal communication systems*. Reading, MA: Addison-Wesley, 1997.

[22] R. Pirhonen, T. Rautava, and J. Penttinen, "TDMA convergence for packet data services", *IEEE Personal Communications Magazine*, vol. 6, pp. 68–73, June 1999.

[23] A. Furuskar, S. Mazur, F. Muller, and H. Olofsson, "EDGE: enhanced data rates for GSM and TDMA/136 evolution", *IEEE Personal Communications Magazine*, vol. 6, pp. 56–66, 1999.

[24] N. R. Sollenberger, N. Seshadri, and R. Cox, "The evolution of IS-136 TDMA for third-generation wireless services", *IEEE Personal Communications Magazine*, vol. 6, pp. 8–18, 1999.

[25] M. Austin, A. Buckley, C. Coursey, P. Hartman, R. Kobylinski, M. Majmundar, and J. P. Seymour, "Service and system enhancements for TDMA digital cellular systems", *IEEE Personal Communications Magazine*, vol. 6, pp. 20–33, 1999.

[26] K. Pahlavan and A. H. Levesque, *Wireless information networks*. New York: John Wiley & Sons, 1995.

[27] H. Holma and A. Toskala, eds, *WCDMA for UMTS radio access for third generation mobile communications*. Chichester: John Wiley & Sons, 2001.

[28] B. A. Miller and C. Bisdikian, *Bluetooth revealed*. Upper Saddle River, NJ: Prentice-Hall, 2001.

[29] J. C. Haartsen and S. Mattison, "Bluetooth — a new low–power radio interface providing short–range connectivity", in *Proc. of the IEEE*, vol. 88, pp. 1651–1661, Oct. 2000.

[30] T. Wilkinson, T. G. C. Phipps, and S. K. Barton, "A report on HIPERLAN standardization", *Int. Journal of Wireless Information Networks*, vol. 2, pp. 99–120, Apr. 1995.

[31] R. van Nee, G. Awater, M. Morikura, and H. Takanashi, "New high-rate wireless LAN standards", *IEEE Communications Magazine*, vol. 37, pp. 82–88, Dec. 1999.

Appendix 17.A *Standards for wireless systems*

17.A.1 General observations

Before presenting some of the consolidated standards for wireless systems, we recall a few concepts that are of great interest in this field.

Wireless systems

Although the trend, through third generation *universal mobile telecommunications service* (UMTS) systems, is to "merge" the different type of services into one large structure, a distinction must be made between two classes of wireless systems: *voice-oriented* systems and *data-oriented* systems. Figure 17.22 illustrates this aspect.

From this scheme we note that both classes of systems can be further subdivided into two categories. Systems for voice transmission can be *cordless*, that is with low transmitted power and local area services [18], or *cellular*, that is with high transmitted power and wide area services. A similar distinction can be made for data transmission services: the *wireless local area networks* (WLANs) are low power systems with short range coverage, whereas the *mobile-data networks* are high power systems with long range coverage (see Sections 17.2 and 17.2.1).

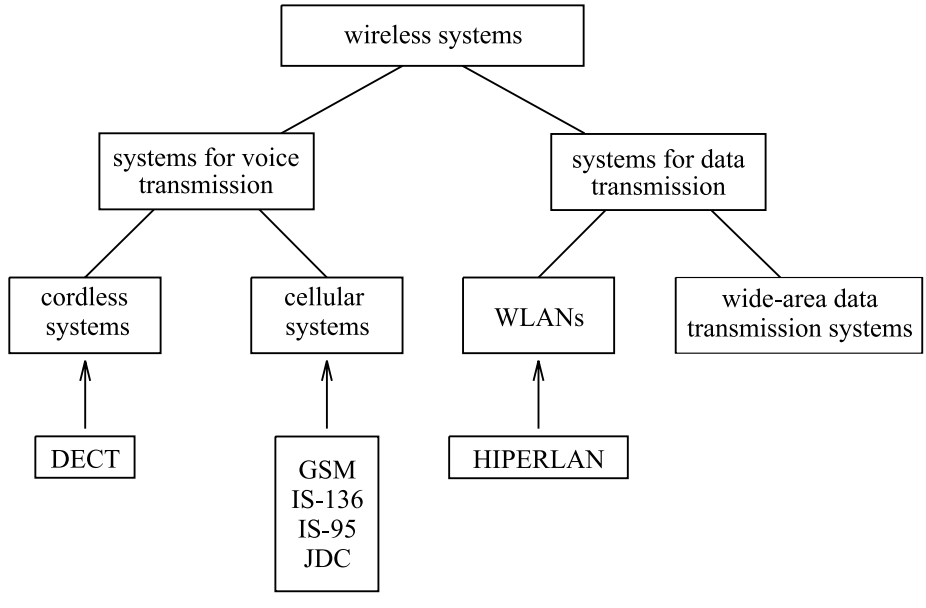


Figure 17.22. Types of services offered by wireless systems.

We will discuss here cellular systems, specifically the standards GSM, IS-136, JDC, and IS-95; we will also address the standard DECT for cordless systems and HIPERLAN for WLANs.

Modulation techniques

The current second generation wireless systems are digital and use different types of modulation according to the required performance. For a discussion of modulation techniques for radio transmission systems, see Chapter 18.

Using a QPSK (or a BPSK) system with a square root raised cosine transmit pulse yields a good trade-off between level of ISI and required bandwidth; in this case, however, the modulated signals do not have a constant envelope, which may represent a problem in the presence of a high power amplifier. An alternative is given by GMSK, whose basic scheme is given in Figure 18.45; in this case the required bandwidth increases for higher values of $B_t T$, where B_t is the bandwidth of the Gaussian filter; the choice of the parameter $B_t T$ is a compromise between pulse duration and system bandwidth.

Figure 17.23 illustrates estimated power spectra of baseband QPSK and GMSK signals for two values of $B_t T$. The frequency is normalized by bit rate $1/T_b$ of the system; in this way it is easy to realize that the bandwidth efficiency of QPSK is larger than that of GMSK.

Parameters of the modulator

There are some essential parameters that must be taken into account for the design and definition of a wireless system.

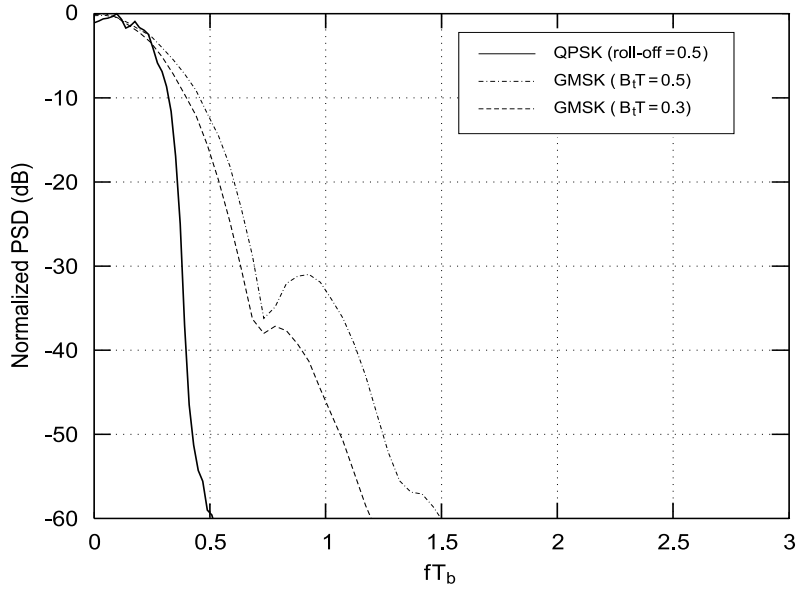


Figure 17.23. Comparison between the PSD of QPSK and GMSK signals.

1. *Bandwidth efficiency or spectral efficiency*: it is the ratio between the bit rate $1/T_b$ and the utilized bandwidth.
2. *Power efficiency*: for the same $1/T_b$ and receiver complexity, the lower the transmitted power for a certain bit error probability, the greater the efficiency.
3. *Out-of-band radiation*: it is the power outside of the main lobe of the power spectrum.[1]
4. *Robustness to multipath*: it expresses the insensitivity of the system to multipath channels.
5. *Constant envelope*: the use of power-efficient class C amplifiers, because of their strong non-linearity, requires input signals with a constant envelope.

Cells in a wireless system

A fundamental concept when we speak of wireless systems is that of a *cell*. The coverage area of a certain system is not "served" by a single transmitter but by numerous transmitters, called *base stations* (BSs) or *base station transceivers* (BTSs): each one of them can "cover" only a small part of this area, called a cell. To each cell a set of carrier frequencies completely different from those of the neighboring cells is assigned, so that *co-channel interference* is as low as possible.

The method of frequency reuse, illustrated in Figure 17.24, is related to the concept of a cell. To two cells separated by an adequate distance, the same set of carrier frequencies can be assigned with minimum co-channel interference; then there is a periodic repetition in assigning the frequencies, as seen in Figure 17.24. *Cluster* is a set of cells that subdivides the available bandwidth of the system. Obviously, each station will interfere with the other stations, especially with those in the same cell and in adjacent cells, as there is always a certain percentage of irradiated power outside of the nominal bandwidth of the assigned channel; hence the strategy in the choice of carrier frequencies assigned to the various cells is of fundamental importance.

Related to the concept of cell is also the concept of *hand-off* or *hand-over*. When a user moves from one cell to another, the communication cannot be interrupted; hence, a mechanism must be present that, at every instant, keeps track of which BTS sends the "strongest" signal and possibly is capable of changing the BTS to which the mobile terminal is "linked".

Other specific aspects will be dealt with in the description of each standard.

17.A.2 GSM standard

System characteristics

The global system for mobile communications (GSM) was started in the early 1990s with the aim of providing one standard within Europe. The services that this system offers are [19]:

[1] For example, MSK has low bandwidth efficiency and high out-of-band radiation; the Gaussian filter of a GMSK system attempts to mitigate both these problems.

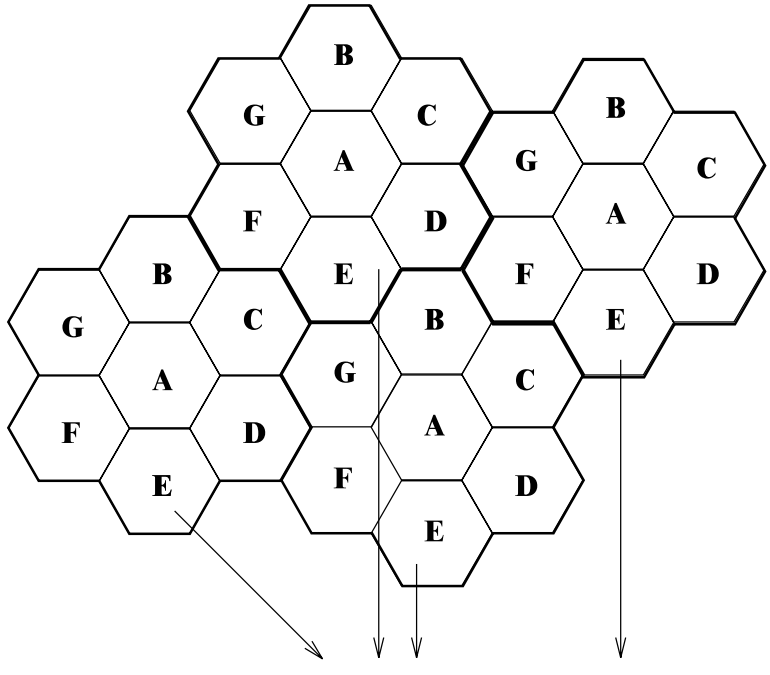


Figure 17.24. Illustration of cell and frequency reuse concepts; cells with the same letter are assigned the same carrier frequencies.

- *Telephony service*: digital telephone service with guarantee of service to users that moves at a speed of up to 250 km/h;
- *Data service*: can realize the transfer of data packets with bit rates in the range from 300 to 9600 bit/s.
- *ISDN service*: some services, as the identification of a user that sends a call and the possibility of sending *short messages* (SMS), are realized by taking advantage of the *integrated services digital network* (ISDN), whose description can be found in [20].

A characteristic of the GSM system is the use of the *subscriber identity module* (SIM) card together with a four-digit *number* (ID); inserting the card in any mobile terminal, it identifies the subscriber who wants to use the service. Important is also the protection of privacy offered to the subscribers of the system.

Figure 17.25 represents the structure of a GSM system, that can be subdivided into three subsystems. The first subsystem, composed of the set of BTSs and mobile terminals or *mobile stations* (MSs), is called a *radio subsystem*; it allows communication between the MSs and the *mobile switching center* (MSC), that coordinates the calls and

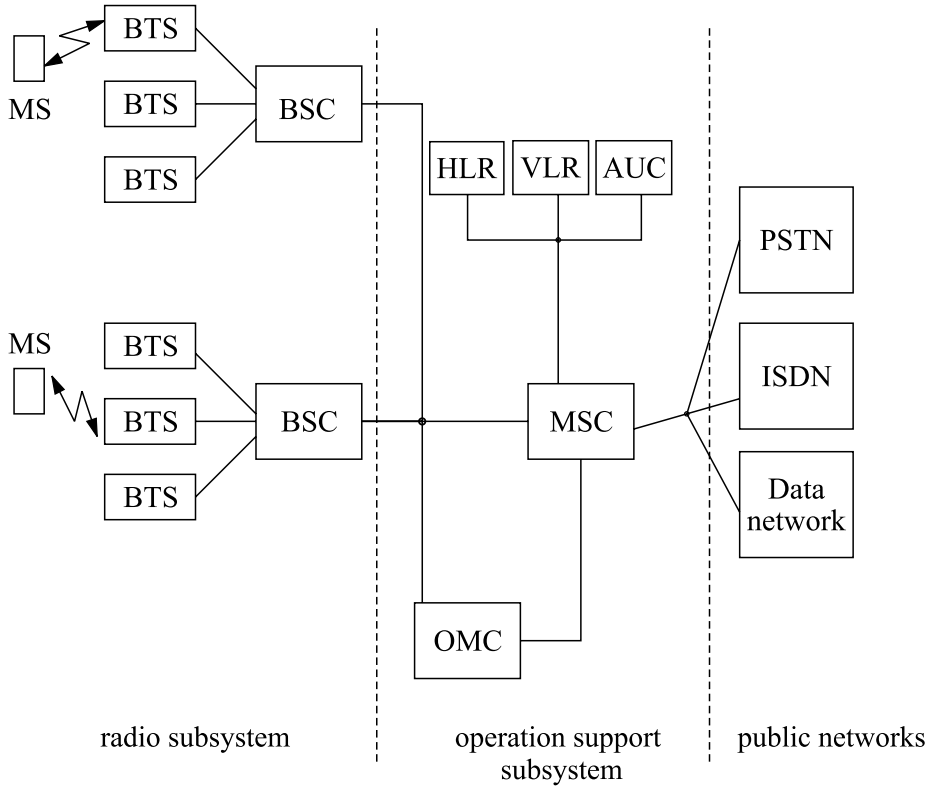


Figure 17.25. GSM system structure.

also other system control operations. To a MSC are linked many *base station controllers* (BSCs); each BSC is linked up to several hundreds BTSs, each of which identifies a cell and directly realizes the link with the mobile terminal. The hand-off procedure between two BTSs is assisted by the mobile terminal in the sense that it is a task of the MS to establish at any instant which BTS is sending the "strongest" signal. In the case of the hand-over between two BTS linked to the same BSC, the entire procedure is handled by the BSC itself and not by the MSC; in this way the MSC can save many operations.

The second subsystem is the *network switching subsystem* (NSS), that in addition to the MSC includes:

- *the home location register* (HLR): this is a database that contains information regarding subscribers who reside in the same geographical area as the MSC;

- *the visitor location register* (VLR): this is a database that contains information regarding subscribers that are temporarily under the control of the MSC, but do not reside in the same geographical area;

- *the authentication center* (AUC): this controls codes and other information for correct communications management;
- *the operation maintenance centers* (OMC): they take care of the proper functioning of the various blocks of the structure.

Finally, the MSC is directly linked to the public networks: PSTN for telephone services, ISDN for particular services as SMS, and data network for the transmission of data packets.

Radio subsystem

We now give some details with regard to the radio subsystem. The total bandwidth allocated for the system is 50 MHz; frequencies that go from 890 to 915 MHz are reserved for MS-BTS communications, whereas the bandwidth 935–960 MHz is for communications in the opposite direction.[2] In this way a full-duplex communication by *frequency division duplexing* (FDD) is realized. Within the total bandwidth there are 248 carriers allocated, that identify as many frequency channels called ARFCN; of these 124 are for uplink communications and 124 for downlink communications. The separation between two adjacent carriers is 200 kHz; the bandwidth subdivision is illustrated in Figure 17.26. Full-duplex communication is achieved by assigning two carriers to the user, one for transmission and one for reception, such that they are about 45 MHz apart.

Each carrier is used for the transmission of an overall bit rate R_b of 270.833 kbit/s, corresponding to a bit period $T_b = 3.692$ μs. The system employs GMSK modulation with parameter $B_t T$ equal to 0.3; the aim is to have a power-efficient system. However, the bandwidth efficiency is not very high; in fact we have

$$\nu = \frac{270.833}{200} \simeq 1.354 \text{ bit/s/Hz} \tag{17.1}$$

which is smaller than that of other systems.

Besides this FDM structure, there is also a TDMA structure; each transmission is divided into eight time intervals, or *time slots*, that identify the TDM frame. Figure 17.27 shows the structure of a frame as well as that of a single time slot.

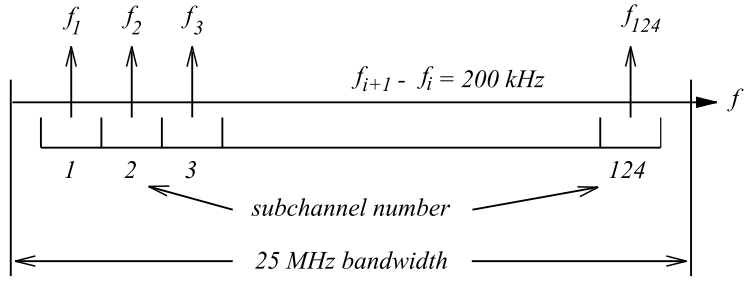


Figure 17.26. Bandwidth allocation of the GSM system.

[2] There also exists a version of the same system that operates at around the frequency of 1.8 GHz (in the USA the frequency is 1.9 GHz).

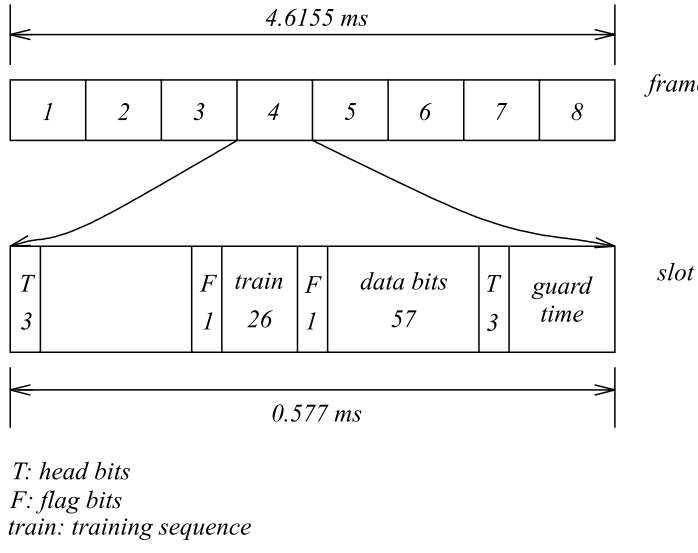


Figure 17.27. TDM frame structure and slot structure of the GSM system.

As a slot is composed of 156.25 bits (not an integer number because of the guard time equal to 8.25 bits), its duration is about 576.92 μs; therefore the frame duration is about 4.615 ms. In Figure 17.27 it is important to note the training sequence of 26 bits, used to analyze the channel by the MS or BS. The *flag bits* signal if the 114 information bits are for voice transmission or for control of the system. Finally, the *tail bits* indicate the beginning and the end of the frame bits.

Although transmissions in the two directions occur over different carriers, to each communication is dedicated a pair of time slots spaced 4 slots apart (one for the transmit station and one for the receive station); for example, the first and fifth or the second and sixth, etc. Considering the sequence of 26 consecutive frames, that have a duration of about 120 ms, the 13th and 26th frames are used for control; then in 120 ms a subscriber can transmit (or receive) $114 \cdot 24 = 2736$ bits, which corresponds to a bit rate of 22.8 kbit/s. Indeed, the net bit rate of the message can be 2.4, 4.8, 9.6, or 13 kbit/s. Redundancy bits are introduced by the channel encoder for protection against errors, so that we get a bit rate of 22.8 kbit/s in any case.

The original speech encoder chosen for the system was a RELP (see Chapter 5), improved by a *long-term predictor* (LTP), with a bit rate of 13 kbit/s. The use of a *voice activity detector* (VAD) allows an improvement in system capacity by reducing the bit rate to a minimum value during silence intervals in the speech signal. For channel coding, a convolutional encoder with code rate 1/2 is used. In [20] the most widely used speech and channel encoders are described, together with the data interleavers.

In Figure 17.28 a scheme is given that summarizes the protection mechanism against errors used by the GSM system. The speech encoder generates, in 20 ms, 260 bits; as the

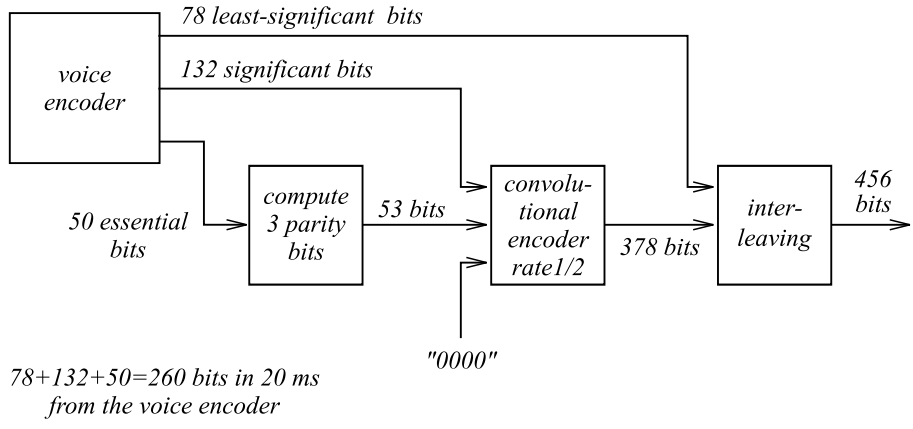


Figure 17.28. Channel coding for the GSM system.

bit rate per subscriber is 22.8 kbit/s, in 20 ms 456 bits must then be generated, introducing a few redundancy bits.[3]

To achieve reliable communications in the presence of multipath channels with delay spread up to 16 μs, at the receiver equalization by a DFE and/or detection by the Viterbi algorithm are implemented.

In [20, 21] the calling procedure followed by the GSM system and other specific details are fully described.

GSM-EDGE

Recently [22, 23], to support services with higher bit rates, the *enhanced data for GSM evolution* (EDGE) system was introduced, which employs 8-PSK, with a pulse represented in Figure 18.57, in place of GMSK. The bit rate is almost tripled to 69.2 kbit/s per subscriber; the channel encoder is also modified.

17.A.3 IS-136 standard

The standard IS-136 is an extension of the digital cellular IS-54 system; in turn IS-54 (called *United States digital cellular* (USDC)), was developed in the early 1990s as an extension of the analog standard AMPS, existing in the United States since the beginning of the 1980s. In many respects IS-136 is similar to GSM, in others is completely different.

[3] The described slot structure and coding scheme refer to the transmission of user information, namely speech. Other types of communications, as for the control and management of the system, use different coding schemes and different time slot structures (see [21]).

The structure of the system (see Figure 17.25) and types of services provided are similar to those of GSM; even IS-136 uses a combination of TDMA and FDMA, although with different specifications.

A first substantial difference between the two systems is given by the modulation adopted: IS-136 uses a $\pi/4$-DQPSK with a roll-off factor of 0.35. A considerable improvement in the bandwidth efficiency is thus obtained. The bandwidth efficiency is given by

$$\nu = \frac{48.6}{30} \simeq 1.62 \text{ bit/s/Hz} \tag{17.2}$$

where the numerator is given by the bit rate per carrier and the denominator by the spacing between carriers. The price for a better bandwidth efficiency is a decrease of the power efficiency with respect to the GSM.

We examine some parameters used by the system. IS-136 has also a bandwidth of 50 MHz; the band 824–849 MHz (or 1850–1865 MHz) is used for the MS-BS transmissions, whereas the band 869–894 MHz (or 1930–1945 MHz) is reserved for communications in the reverse direction; again, the FDD technique is used to achieve full-duplex communication. Moreover, as in GSM, a combination of FDMA and TDMA is used to realize multiple access; the overall bandwidth is divided into sub-bands of 30 kHz, each carrying an overall bit rate of 48.6 kbit/s, that corresponds to a bit period of $T_b \simeq 20.576$ μs.

The structure of the TDM frame is illustrated in Figure 17.29, together with that of a single slot, which can be of two types, depending on whether it is used for communication from the mobile station to the base or vice versa.

The frame is divided into 6 time slots, and each slot is composed of 324 bits, for a duration of about 6.667 ms, which means that the frame duration is of about 40 ms. As for GSM, for full-duplex communication two carriers, separated by about 45 MHz, are assigned: one for the MS-BS communication and the other for the reverse. This time, however, 2 time slots are assigned to the user, such that at most 3 users[4] can use the same frame.

We see from Figure 17.29 that, of the overall 324 bits per slot, only 260 bits are effective information, and the remaining serve as control and signaling for the system. Then in 40 ms there are only $260 \cdot 2 = 520$ information bits (transmitted or received), which corresponds to a bit rate of 13 kbit/s. The voice encoder is a *vector-sum excitation linear predictive* (VSELP) at 7.95 kbit/s, that generates only 159 bits in 20 ms (half a frame period). However, we have seen that in 20 ms there are $520/2 = 260$ transmitted bits, hence the difference, that is 101 bits, is constituted by redundancy bits added by the channel encoder.

Figure 17.30 illustrates channel coding for voice messages; for control messages the organization is different. The mechanism is similar to that of GSM; the 159 bits, generated by the voice encoder in 20 ms, are divided into two classes: the first comprises the 77 most significant bits (divided into a group of 65 bits and one of 12 bits), the second, the remaining 82 bits. After the procedure illustrated in the figure, the bits become 260, for a bit rate of 13 kbit/s.

The type of demodulator/equalizer is not specified. For channels with a rms delay spread greater than 4.12 μs, a RLS adaptive DFE was proposed [20].

For information on other characteristics of the IS-136 system and on its evolution, as well as on convergence with GSM-EDGE, we recommend references [24, 25].

[4] In the GSM, 8 users share the same frame; in fact, to each user is assigned only one slot.

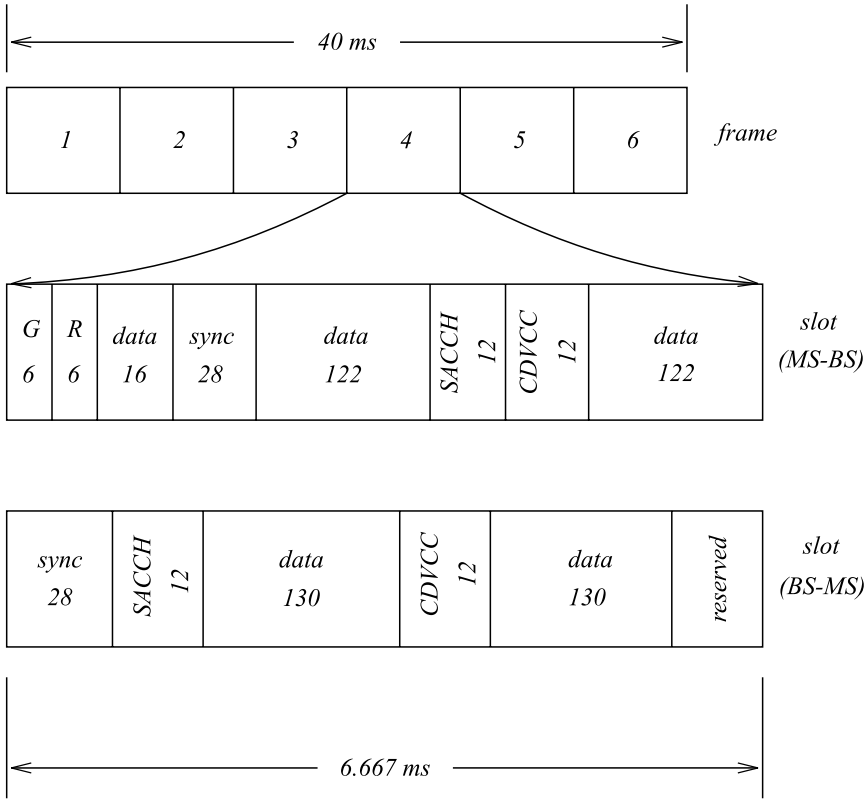


Figure 17.29. TDM frame structure and slot structure for the IS-136 system.

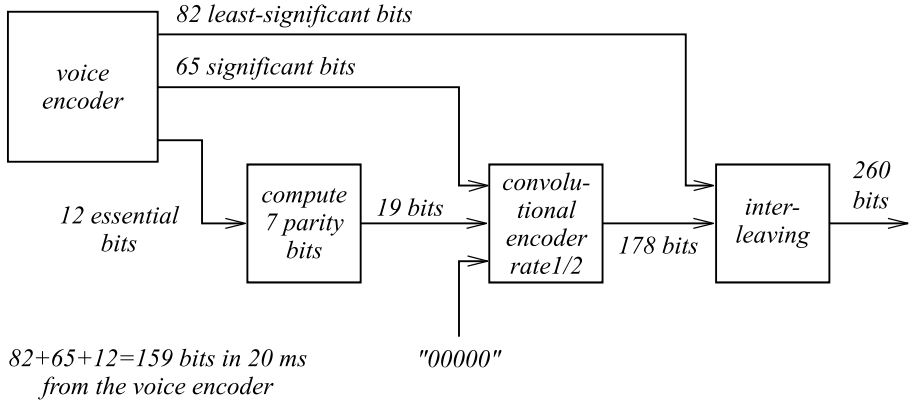


Figure 17.30. Channel coding for voice messages of the IS-136 system.

17.A.4 JDC standard

We discuss the *Japanese digital cellular* (JDC), which is very similar to the IS-136 standard. The JDC standard differs from the previous two standards mainly due to the fact that the total bandwidth of 80 MHz allocated for this system is much larger than that of GSM and IS-136. The frequency interval 940–956 MHz is reserved for base-mobile station communications, whereas the band 810–826 MHz is for communications in the opposite direction. To these two sub-bands two other sub-bands are added: the interval 1477–1501 MHz is for BS-MS communication, and the interval 1429–1453 MHz for MS-BS communications. The first pair of sub-bands is called *low band* (LB), the second *high band* (HB). The FDD technique is used to achieve full-duplex communication; if the LB is used, the transmit carrier and the receive carrier are separated by 130 MHz; if we use the HB, the separation is only 48 MHz. Channels have a bandwidth of 25 MHz, both in the LB and in the HB; each channel is used for transmission of an overall bit rate $R_b = 42$ kbit/s, that corresponds to a bit period of about 23.81 μs.

Similar to the other two systems, besides the described FDMA structure, a TDMA structure is also used; a frame consists of 3 time slots and each one is composed of 280 bits (224 of information, 56 of control and signaling), which corresponds to a frame period of $3 \cdot 280 \cdot 23.81 = 20$ ms. As one slot per frame is utilized, 280 bits (of which only 224 are of information) the user can transmit (or receive) information in 20 ms, which corresponds to a bit rate of 11.2 kbit/s. The speech encoder is a VSELP operating at 6.7 kbit/s and, after channel coding, the bit rate increases to 11.2 kbit/s. Channel coding is performed by a convolutional encoder with code rate 9/17.

$\pi/4$-DQPSK with a roll-off factor of 0.5 is adopted. As for the other two standards, the hand-off is assisted by the mobile terminal to reduce the MSC load.

17.A.5 IS-95 standard

As IS-136, the IS-95 system is a digital cellular system for North American geographical areas; it was also developed to be compatible with the AMPS standard, and is based on the CDMA spread spectrum technique, rather than on TDMA as the previously described systems [20, 26].

FDMA is still used: the total bandwidth allocated to the system is identical to that of IS-136 and has the same organization,[5] but the channel bandwidth is 1.25 MHz. Also FDD is still used with a separation of 45 MHz between transmit and receive carriers.

An important characteristic of this system is that the bit rate of the speech encoder can vary to lower power consumption, and is reduced to a minimum during silence intervals. The voice encoder is a *Qualcomm codebook excited linear prediction* (QCELP) with variable bit rate: 1.2, 2.4, 4.8, or 9.6 kbit/s.[6]

Interesting is the fact that the MS-BS communication greatly differs from the reverse: in particular, different sequences are employed to perform spreading. When the BS is

[5] Within the total bandwidth, each of the two systems should use only a portion of the spectrum so that they do not interfere with each other.

[6] In fact, the effective rate is from 1 to 8 kbit/s and the higher rates are due to the fact that additional redundancy bits have already been included.

transmitting, data to all mobile terminals within the same cell (identified by a carrier) are sent using a different spreading sequence for each user. Regarding the MS, in transmission the communication occurs asynchronously, with the requirement that the power level of the received signals must be as uniform as possible.

Whatever the bit rate used by the voice encoder, the channel encoder brings the bit rate to 19.2 kbit/s. The spreading procedure used for the BS-MS transmission is illustrated in Figure 17.31.

Note that, after channel coding, interleaving is performed. Then the bit rate is increased to 1.2288 Mbit/s using one of 64 Walsh codes; to each user in a cell is assigned a different code, in order to maintain separation among signals of different users in the same cell and eliminate interference, at least in the absence of multipath. Moreover, to reduce the level of interference between two users that are in different cells, and to which the same Walsh code is assigned, scrambling is used; without going into details (see Chapter 10), the function of scrambling is to achieve quasi-orthogonality between the signals of the two users, also for different lags, so that the interference is kept at a low level [20]. Before the two modulation filters, a short pilot PN sequence is also inserted, that is used at the receiver for various purposes, e.g., channel identification.

The system for the MS-BS communication has a different structure: channel coding is performed by a convolutional encoder with code rate 1/3, that yields a bit rate of 28.8 kbit/s. Spreading is still done by selecting one of 64 Walsh codes, but in a different way as compared to the BS-MS transmission [20].

QPSK is adopted; in particular, the modulation used by the mobile station is OQPSK. For detection, a RAKE receiver is used.

We conclude this section by mentioning the third generation (3G) mobile radio system denoted *universal mobile telecommunications system* (UMTS), in the process of being standardized. 3G systems are intended for the transmission at rates from 16 kbit/s up to

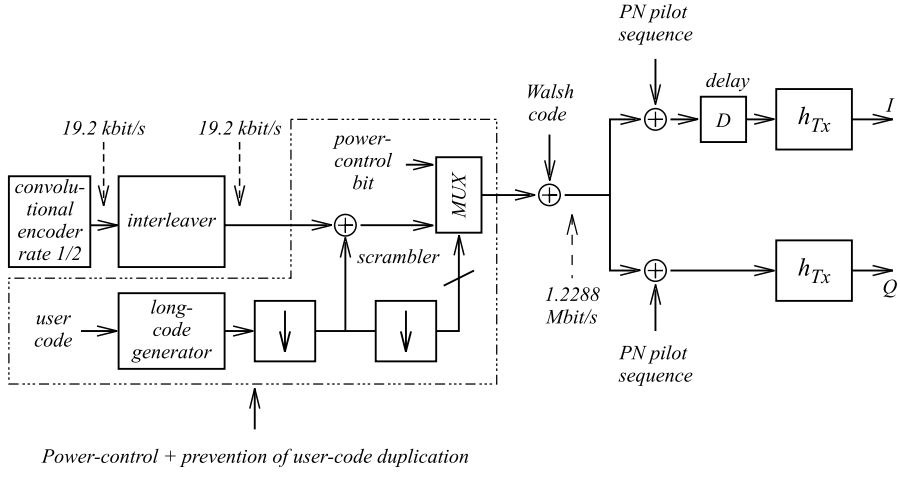


Figure 17.31. Spreading procedure used by the IS-95 system for transmission from the base station to the mobile station.

2 Mbit/s. Various approaches have been identified for the definition of the link layer; the most important is the *wideband* CDMA (WCDMA) system [27].

17.A.6 DECT standard

The *digital European cordless telecommunications* (DECT) system is a cordless digital system used in Europe. Unlike cellular systems that use cells with radius of the order of a few kilometers, the DECT system is mainly employed indoor and the cell radius is at most of a few tens of meters (typically 100 m). Services provided by the system are:

- speech transmission,
- data transmission,
- support to the ISDN network.

The main characteristics of DECT are summarized in Table 17.5. For comparison, the main characteristics of PACS are summarized in Table 17.6.

While originally the hand-off problem was not considered, as each MS corresponded to one BS only, now even for DECT we speak of hand-off assisted by the mobile terminal, such that the system configuration is similar[7] to that of a cellular system (see Figure 17.25). An interesting characteristic is the use of the *dynamic channel selection* (DCS) algorithm, that allows the portable to know at every moment which channel (frequency) is the best (with the lowest level of interference) for communication and select it. We briefly illustrate the calling procedure followed by the system:

1. When a mobile terminal wishes to make a call, it first measures the received signals from the various BS[8] and selects the one which yields the best signal level.
2. By the DCS algorithm, the mobile terminal selects the best free channels of the selected BS.

Table 17.5 Table summarizing the main characteristics of DECT.

Frequency range	1880÷1900 MHz
RF channel spacing	1728 kHz
Modulation	GMSK
Transmission bit rate	1152 kbit/s
Voice encoding method	32 kbit/s ADPCM
Access method	FDMA/TDMA/TDD
Frame duration	10 ms (24 time slots)
Subscriber TX peak power	250 mW
Radius of service	100÷150 m
Frequency planning	Dynamic channel allocation

[7] For a discussion of the differences between a cellular system and a cordless system, see Section 17.A.1.

[8] For DECT, the acronyms *portable handset* (PH) in place of MS and *radio fixed part* (RFP) in place of BS, are often used.

Table 17.6 Table summarizing the main characteristics of PACS.

Frequency range	1850÷1910 MHz uplink 1930÷1990 MHz downlink
RF channel spacing	300 kHz
Modulation	$\pi/4$ QPSK
Transmission bit rate	384 kbit/s
Voice encoding method	32 kbit/s ADPCM
Access method	TDD/TDMA
Frame duration	2.5 ms (8 time slots)
Subscriber TX peak power	200 mW
Radius of service	300÷500 m
Frequency planning	Quasi-static automatic frequency assignment

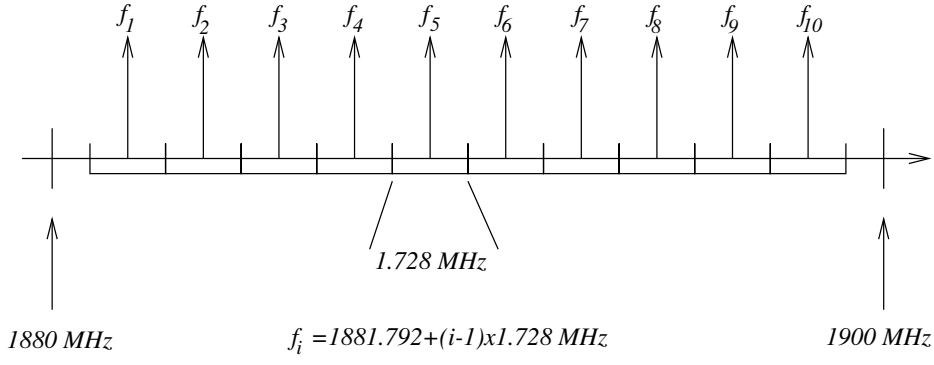


Figure 17.32. FDMA structure of the DECT system.

3. MS sends a message, called *access request*, over the least interfered channel.
4. BS sends (or not) an answer: *access granted*.
5. If the MS receives this message, in turn it transmits the *access confirm* message and the communication starts.
6. If the MS does not receive the *access granted* signal on the selected channel, it abandons this channel and selects the second least interfered channel, repeating the procedure; after failing on 5 channels, the MS selects another BS and repeats all operations.

The total band allocated to the system goes from 1880 to 1900 MHz and is subdivided into ten *sub-bands*, each with a width of 1.728 MHz; this is the FDMA structure of DECT represented in Figure 17.32. Each channel has an overall bit rate of 1.152 Mbit/s, that corresponds to a bit period of about 868 ns.

Similar to the other systems, TDMA is used. The TDM frame, given in Figure 17.33, is composed of 24 slots: the first 12 are used for the communication BS-MS, the other

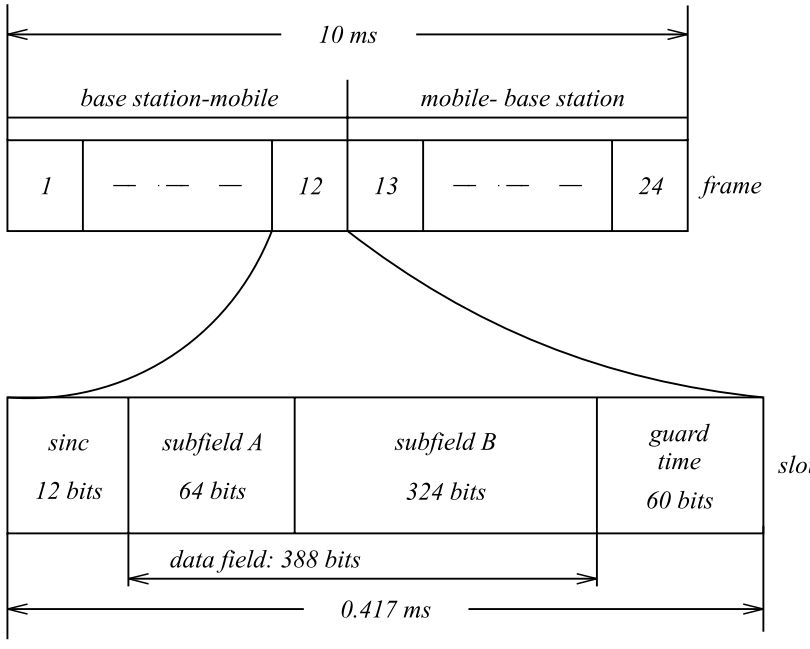


Figure 17.33. TDM frame structure and slot structure for the DECT system.

12 for the reverse communication; thus we realize a full-duplex communication by time division duplexing (TDD). In this DECT differs from all above considered wireless systems which use FDD; DECT allocates half of the frame for transmission and the other half for reception.

In Figure 17.33 the slot structure is also shown; it is composed of 480 bits of which the first 32 are fixed and correspond to a synchronization word, the successive 388 are information bits, and the remaining 60 constitute the guard time. The frame has a duration of $480 \cdot 24/1152 = 10$ ms. The field of 388 bits reserved for information bits is subdivided into two subfields A and B: the first (64 bits) contains information for signaling and control of the system, the second (324 bits) contains user data. If the signal is speech, 4 of these 324 bits are parity bits, which translates into a net bit rate of 320 bits in a frame interval of 10 ms, and therefore in a bit rate of 32 kbit/s; an ADPCM voice encoder at 32 kbit/s is used and no channel coding is provided.

For transmission, GMSK with parameter $B_t T$ of 0.5 is adopted. At the MS receiver a pair of antennas are often used to realize switched antenna diversity. With this mechanism, fading and interference are mitigated.

We summarize in Table 17.7 the characteristics of the various standards so far discussed. A standard that has in part overcome DECT is Bluetooth [28, 29], that operates around the frequency of 2.45 GHz, in the unlicensed and open industrial-scientific-medical (ISM) band. Bluetooth uses a FH/TDD scheme with Gaussian-shaped FSK ($h \leq 0.25$). The modulation rate is 1 MBaud with a slot of 625 μs; a different hop channel is used for each slot. This gives a nominal hop rate of 1600 hops/s. There are 79 hop carriers spaced by 1 MHz.

Table 17.7 Summary of characteristics of the standards GSM, IS-136, JDC, IS-95, and DECT.

System	GSM	IS-136	JDC	IS-95	DECT
Multiple access	TDMA/ FDMA	TDMA/ FDMA	TDMA/ FDMA	FDMA/ CDMA	TDMA/ FDMA
Band (MHz)					
Forward (BS→MS)	935÷960 1805÷1880	869÷894 1930÷1945	940÷960 1477÷1501	935÷960	1880÷1900
Reverse (MS→BS)	890÷915 1710÷1785	824÷849 1850÷1865	810÷830 1429÷1453	824÷849	1880÷1900
Duplexing	FDD	FDD	FDD	FDD	TDD
Spacing between carriers	200 kHz	30 kHz	25 kHZ	1250 kHz	1728 kHz
Modulation	GMSK 0.3	$\pi/4$-DQPSK	$\pi/4$-DQPSK	QPSK	GMSK 0.5
Bit rate per carrier (kbit/s)	270.833	48.6	42	1228.8	1152
Speech encoder	RPE-LTP	VSELP	VSELP	QCELP	ADPCM
bit rate (kbit/s)	1 3	7.95	6.7	1.2÷9.6	32
Convolutional encoder* code rate	1/2	1/2		1/2 (F) 1/3 (R)	none
Frame duration (ms)	4.615	40	20	20	10

*All these standards use a CRC code, possibly together with a convolutional code.

The maximum bit rate is of 723.2 kbit/s in one direction and 57.6 kbit/s in the reverse direction.

17.A.7 HIPERLAN standard

In 1991 ETSI started the standardization process for the realization of an European high speed and low power data transmission system for WLAN applications. As cordless systems, also HIPERLAN supports mobile users, up to a maximum speed of about 35 km/h.

HIPERLAN has a double application: the first is that of replacing (see Figure 17.34) the last network link to the various users, in order to reduce costs due to the reconfiguration of the position of the terminal, the second is that of constituting, in indoor environments, a network among mobile (and fixed) terminals. There are two versions of the HIPERLAN standard.

HIPERLAN type 1. The allocated band for the system is from 5.15 to 5.30 GHz and from 17.1 to 17.3 GHz. The first bandwidth of 150 MHz is still the most used and is further divided into 5 sub-bands; the first carrier is at 5.176468 GHz, with a channel bandwidth of 23.5294 MHz. Figure 17.35 shows the band allocation.

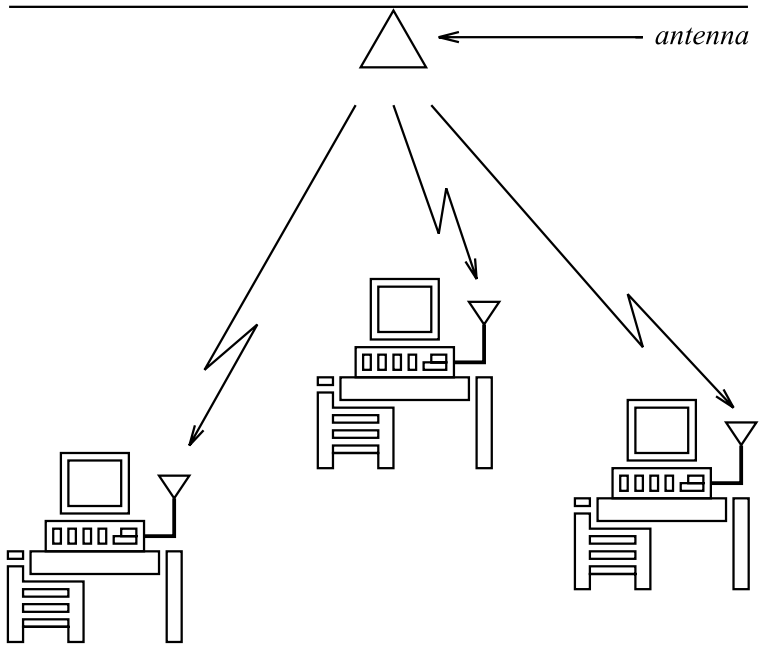


Figure 17.34. Configuration of a HIPERLAN network.

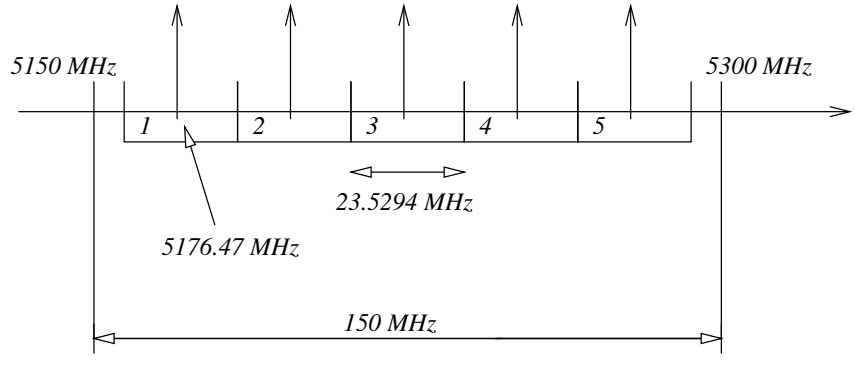


Figure 17.35. Band allocation of the HIPERLAN type 1 system.

The maximum overall (raw) bit rate is of 23.5294 Mbit/s with a corresponding bit interval of about 42.5 ns. The system is also able to work at lower bit rates (the so-called low rate of 1.4706 Mbit/s), in order to reduce power consumption. The transmission takes place by data packets: m blocks of 496 bits, with m that can vary from 1 to 47, are transmitted. Before the transmission of these blocks, there is the transmission of a training sequence of 450 bits, which is needed for channel identification and synchronization. Figure 17.36 shows the block diagram of the transmitter.

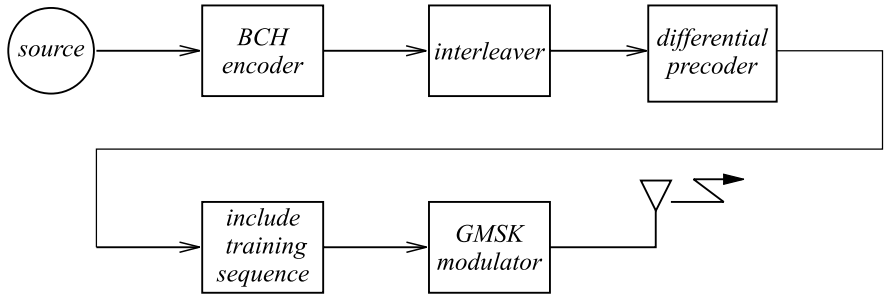


Figure 17.36. Block diagram of the transmitter for the HIPERLAN type 1 system.

The source generates a sequence of m blocks, each one of 416 bits; at the output of the channel encoder the block length is of 496 bits. After interleaving, differential precoding is performed and successively a training sequence is inserted; the generated symbols are then input to the modulator. For HIPERLAN type 1, GMSK with parameter $B_t T$ equal to 0.3 is adopted.[9]

At the receiver, adaptive equalization is performed before decoding [30].

HIPERLAN type 2. A more recent version of HIPERLAN allows transmission at a raw bit rate of 54 Mbit/s, for a net bit rate of up to 45 Mbit/s. The modulation is DMT-OFDM, with 48 subchannels.

Other WLAN standards are described in [31], among which we mention the IEEE 802.11 standard.

[9] For low rate communication, the modulation is simple FSK.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 18

Modulation techniques for wireless systems

In this chapter, after an overview of front-end architectures for mobile radio receivers, we discuss modulation and demodulation schemes that are well suited for application to mobile radio systems because of their simplicity and robustness against disturbances introduced by the transmission channel. Appendix 17.A describes some of the standards where these schemes are adopted.

18.1 Analog front-end architectures

Conventional superheterodyne receiver

Many receivers for radio frequency (RF) communications use the conventional superheterodyne scheme illustrated in Figure 18.1 [1]. A first *mixer* employs a *local oscillator* (LO), whose frequency f_1 is variable, to shift the signal to around a fixed *intermediate frequency* (IF) f_{IF}. The output signal of the first mixer is filtered by a passband filter centered around the frequency f_{IF} to eliminate out of band components. The cascade of RF filter, *linear amplifier* (LNA), and *image rejection* (IR) filter performs the tasks of amplifying the desired signal as well as eliminating the noise outside of the desired band and rejecting the spectral image components introduced by the LO. The IF filter output signal is shifted into baseband through a second mixer, which makes use of an oscillator with a fixed frequency f_2, whose output is filtered with a lowpass filter to eliminate high frequency components. The signal is then sent to the analog-to-digital converter.

One drawback of the superheterodyne receiver is the high selectivity in frequency (high Q) that the various elements must exhibit, which makes an integrated implementation at high frequency a difficult problem.

Problems related to a full integration can be divided into two categories.

1. Integration of the acquisition structure and signal processing requires the elimination of both IR and IF filters.

2. For channel selection, a synthesizer can be integrated using an on-chip VCO with low Q, which, however, yields poor performance in terms of phase noise.

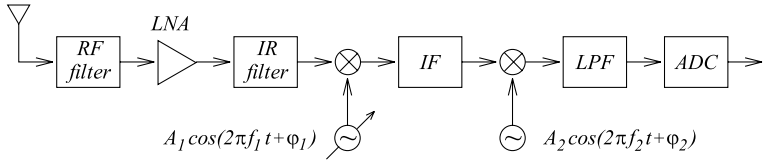


Figure 18.1. Conventional superheterodyne receiver.

Alternative architectures

In this section three receiver architectures are considered that attempt to integrate the largest number of receiver elements. To reach this objective, the A/D conversion of the scheme of Figure 18.1 must be shifted from baseband to IF or RF. In this case implementation of the channel selection by an analog filter bank is not efficient; indeed, it is more convenient to use a wideband analog filter followed by a *digital filter bank*. Reference is also made to multi-standard *software-defined radio* (SDR) receivers, or to the possibility of receiving signals that have different bandwidths and carriers, that are defined according to different standard systems.

We can identify two approaches to the A/D conversion.

1. *Full bandwidth digital conversion*: by using a very high sampling frequency, the whole bandwidth of the SDR system is available in the digital domain, i.e. all desired channels are considered. Considering that this bandwidth can easily achieve 100 MHz, and taking into account the characteristics of interference signals, the dynamic of the ADC should exceed 100 dB. This solution, even though it is the most elegant, cannot easily be implemented as it presents a high complexity and high power consumption.
2. *Partial bandwidth digital conversion*: this approach uses a sampling frequency that is determined by the radio channel that presents the most extended bandwidth within the different systems we want to implement.

The second approach will be considered, as it leads to an SDR system architecture that can be implemented with moderate complexity.

Direct conversion receiver

This architecture eliminates many off-chip components. In this approach, illustrated in Figure 18.2, all desired channels are shifted to baseband through a mixer that uses an oscillator with a varying frequency. Unwanted spectral components are removed by an on-chip baseband filter. Clearly this structure allows a higher integration level with respect to the superheterodyne scheme; in fact this architecture offers two important advantages:

1. the problem of image components is bypassed and the IR filter is not needed;
2. the IF filter and following operations were substituted with a lowpass filter and a baseband amplifier, which are suitable for monolitic integration.

However, there are numerous problems with this structure; in fact, the oscillator is at the same frequency as the RF carrier and there is a possibility of leakage of the oscillator signal

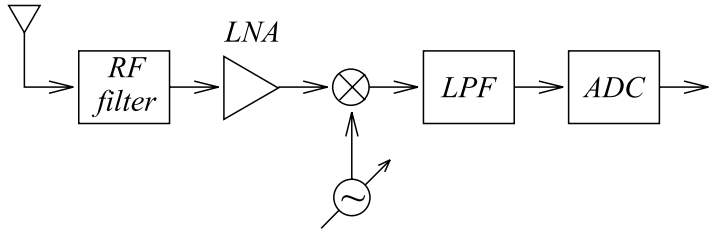


Figure 18.2. Direct conversion receiver.

towards the mixer input as well as towards the antenna, with consequent radiation. The interference signal generated by the oscillator could be reflected on surrounding objects and be "re-received": therefore this spurious signal would produce a time variant DC offset [1] at the mixer output.

To understand the origin and consequences of this offset, we can make the following two observations.

1. The isolation between the oscillator input and the inputs of the mixer and linear amplifier is not infinite; in fact an LO leakage is determined by both capacitive coupling and device substrate. The spurious signal appears at the linear amplifier and mixer inputs and is then mixed with the signal generated by the oscillator, creating a DC component; this phenomenon is called *self-mixing*. A similar effect is present when there is a strong signal coming from the linear amplifier or from the mixer that couples with the LO input: this signal would then be multiplied by itself.
2. In Figure 18.2 to amplify the input signal, that is of the order of microvolts, to a level such that it can be digitized by a low cost and low power ADC, the total gain, from the antenna to the LPF output, is about 100 dB. Of this gain, 30 dB are usually provided by the linear amplifier/mixer combination. With these data we can make a first computation of the offset due to self-mixing. We assume that the oscillator generates a signal with a peak-to-peak value of 0.63 V and undergoes an attenuation of 60 dB when it couples with the LNA input. If the gain of the linear amplifier/mixer combination is 30 dB, then the offset produced at the mixer output is of the order of 10 mV; if directly amplified with a gain of 70 dB, the voltage offset would saturate the circuits that follow.

The problem is even worse if self-mixing is time variant. This event, as previously mentioned, occurs if the oscillator signal leaks to the antenna, thus being irradiated and reflected back to the receiver from moving objects.

Finally, the direct conversion receiver needs a tuner for the channel frequency selection that works at high frequency and with low phase noise; this is hardly obtainable with an integrated VCO with low Q.

Single conversion to low-IF

The low-IF architecture reduces the problem of DC offset present in the direct conversion receiver. This system has the same structure as Figure 18.2 and, similarly to direct

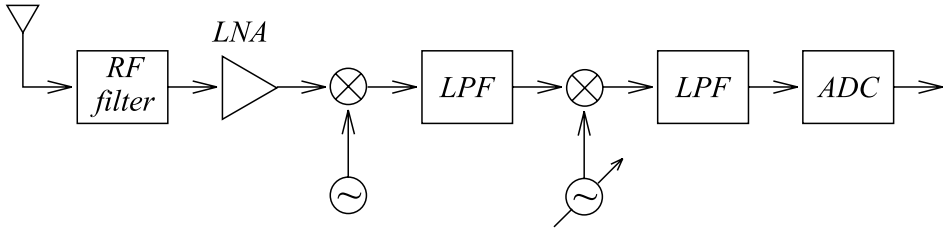


Figure 18.3. Double conversion with wideband IF.

conversion, it utilizes a single mixer stage; the principal difference is that the frequency shift is not made to DC but to a small intermediate frequency. The main advantage of the low-IF system is that the desired channel has no DC components; therefore, the usual problems that emerge from DC offset present in the direct conversion are avoided.

Double conversion and wideband IF

This architecture, illustrated in Figure 18.3, takes all desired channels and shifts them from RF to IF using the first mixer by a fixed frequency oscillator; a simple lowpass filter is used to eliminate high frequency image components. All channels are then shifted down to baseband by a second mixer, this time with a variable frequency oscillator. The baseband filter that follows has a variable gain and eliminates spectral components not belonging to the desired signal band. Channel tuning is obtained by the second low frequency LO; in this case the first RF oscillator can be implemented by a quartz oscillator with a fixed frequency, while the second can be implemented by on-chip techniques that provide low-frequency oscillators with low phase noise and low Q.

It is important to emphasize the absence of oscillators that operate at the same frequency of the RF carrier; this eliminates the potential problem of "re-radiation" of the oscillator. We can conclude by affirming that the double conversion architecture is the most adequate for the analog front-end.

18.2 Three non-coherent receivers for phase modulation systems

We introduce three non-coherent receivers to demodulate phase-modulated signals.

18.2.1 Baseband differential detector

In M-PSK, the transmitted signal (isolated pulse) is given by (6.128). For a continuous M-DPSK transmission with symbol period T, the transmitted signal is given by

$$s(t) = \sum_{k=-\infty}^{+\infty} Re[e^{j\psi_k} h_{Tx}(t - kT) e^{j2\pi f_0 t}] \tag{18.1}$$

where ψ_k is the phase associated with the transmitted symbol at instant kT given by the recursive equation (6.157).

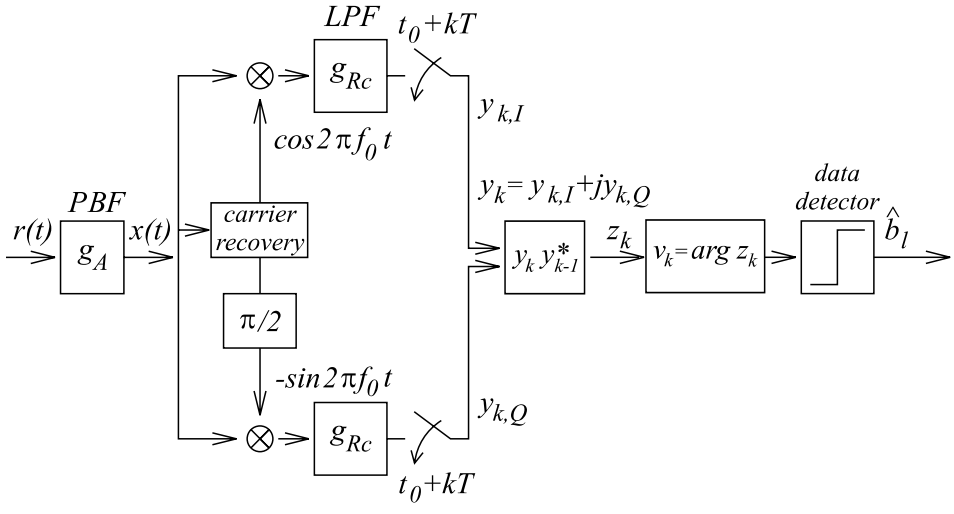


Figure 18.4. Non-coherent baseband differential receiver. Thresholds are set at $(2n-1)\pi/M$, $n = 1, 2, \ldots, M$.

At the receiver the signal r is a version of s, filtered by the transmission channel and corrupted by additive noise. We denote as g_A the cascade of passband filters used to amplify the desired signal and partially remove noise. As shown in Figure 18.4, let x be the passband received signal, centered around the frequency f_0, equal to the carrier of the transmitted signal

$$x(t) = Re[x^{(bb)}(t)e^{j2\pi f_0 t}] \tag{18.2}$$

where $x^{(bb)}$ is the complex envelope of x with respect to f_0. Using the polar notation $x^{(bb)}(t) = M_x(t)e^{j\Delta\varphi_x(t)}$, (18.2) can be written as

$$x(t) = M_x(t)\cos(2\pi f_0 t + \Delta\varphi_x(t)) \tag{18.3}$$

where $\Delta\varphi_x(t)$ is the instantaneous *phase deviation* of x with respect to the carrier phase (see Chapter 1, equation (1.207)).

In the ideal case of absence of distortion and noise, sampling at suitable instants yields $\Delta\varphi_x(t_0 + kT) = \psi_k$. Then for the recovery of the phase θ_k we can use the receiver scheme of Figure 18.4, which, based on signal x, determines the baseband component y as

$$y(t) = y_I(t) + jy_Q(t) = \tfrac{1}{2}x^{(bb)} * g_{Rc}(t) \tag{18.4}$$

The phase variation of the sampled signal $y_k = y(t_0 + kT)$ between two consecutive symbol instants is obtained by means of the signal

$$z_k = y_k y_{k-1}^* \tag{18.5}$$

Always in the ideal case and assuming that g_{Rc} does not distort the phase of $x^{(bb)}$, z_k turns out to be proportional to $e^{j\theta_k}$. The simplest data detector is the threshold detector based on

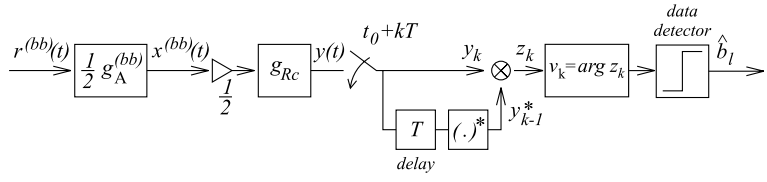


Figure 18.5. Baseband equivalent scheme of Figure 18.4.

the value of

$$v_k = \arg z_k = \Delta\varphi_x(t_0 + kT) - \Delta\varphi_x(t_0 + (k-1)T) \tag{18.6}$$

Note that a possible phase offset $\Delta\varphi_0$ and a frequency offset Δf_0, introduced by the receive mixer, yields a signal y given by

$$y(t) = [\tfrac{1}{2}x^{(bb)}(t)e^{-j(\Delta\varphi_0 + 2\pi\,\Delta f_0 t)}] * g_{Rc}(t) \tag{18.7}$$

Assuming that g_{Rc} does not distort the phase of $x^{(bb)}$, the signal v_k becomes

$$v_k = \Delta\varphi_x(t_0 + kT) - \Delta\varphi_x(t_0 + (k-1)T) - 2\pi\,\Delta\, f_0 T \tag{18.8}$$

which shows that the phase offset $\Delta\varphi_0$ does not influence v_k, while a frequency offset must be compensated by the data detector, summing the constant phase $2\pi\,\Delta f_0 T$.

The baseband equivalent scheme of the baseband differential receiver is given in Figure 18.5.

The choice of h_{Tx}, $g_A^{(bb)}$, and g_{Rc} is governed by the same considerations as in the case of a QAM system; for an ideal channel, the convolution of these elements must be a Nyquist pulse.

18.2.2 IF-band (1 Bit) differential detector (1BDD)

The scheme of Figure 18.4 is introduced only to illustrate the basic principle, as its implementation complexity is similar to that of a coherent scheme. Indeed, this scheme has a reduced complexity because specifications on the carrier recovery can be less stringent; moreover, it does not need phase recovery.

An alternative scheme that does not use carrier recovery is illustrated in Figure 18.6. In this case the signal is first delayed by a symbol period T and then multiplied by itself (I branch) and with its $-\pi/2$ phase-shifted version (Q branch) by a Hilbert filter. On the two branches the signals are given by

$$\begin{aligned} I &: x(t)x(t-T) \\ &= M_x(t)\cos[2\pi f_0 t + \Delta\varphi_x(t)]M_x(t-T)\cos[2\pi f_0(t-T) + \Delta\varphi_x(t-T)] \\ Q &: x^{(h)}(t)x(t-T) \\ &= M_x(t)\sin[2\pi f_0 t + \Delta\varphi_x(t)]M_x(t-T)\cos[2\pi f_0(t-T) + \Delta\varphi_x(t-T)] \end{aligned} \tag{18.9}$$

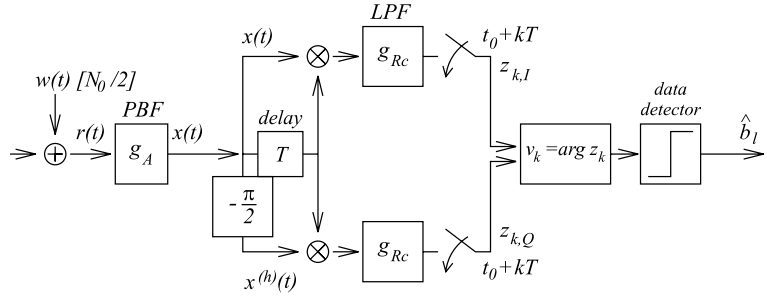


Figure 18.6. Non-coherent 1 bit differential detector.

The filter g_{Rc} removes the components around $2f_0$; the sampled filter outputs are then given by

$$\begin{aligned} I : z_{k,I} &= M_x(t_0+kT)M_x(t_0+(k-1)T) \\ &\quad \tfrac{1}{2}\cos[2\pi f_0 T + \Delta\varphi_x(t_0+kT) - \Delta\varphi_x(t_0+(k-1)T)] \\ Q : z_{k,Q} &= M_x(t_0+kT)M_x(t_0+(k-1)T) \\ &\quad \tfrac{1}{2}\sin[2\pi f_0 T + \Delta\varphi_x(t_0+kT) - \Delta\varphi_x(t_0+(k-1)T)] \end{aligned} \tag{18.10}$$

If $f_0 T = n$, n an integer, or by removing this phase offset by phase shifting x or z_k, it results

$$v_k = \tan^{-1}\frac{z_{k,Q}}{z_{k,I}} = \Delta\varphi_x(t_0+kT) - \Delta\varphi_x(t_0+(k-1)T) \tag{18.11}$$

as in (18.6).

The baseband equivalent scheme is shown in Figure 18.7, where, assuming that g_{Rc} does not distort the desired signal,

$$z_k = \tfrac{1}{2}x^{(bb)}(t_0+kT)x^{(bb)*}(t_0+(k-1)T) \tag{18.12}$$

Typically, in this modulation system, the transmit filter h_{Tx} is a rectangular pulse or a Nyquist pulse; instead, g_A is a narrow band filter to eliminate out of band noise.

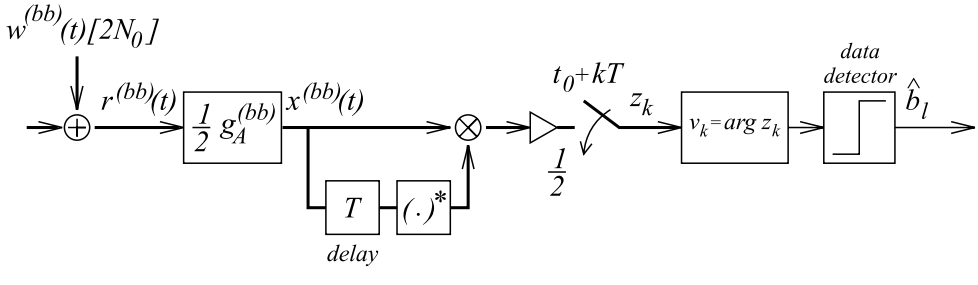


Figure 18.7. Baseband equivalent scheme of Figure 18.6.

For a simple DBPSK, with $\theta_k \in \{0, \pi\}$, we only consider the I branch, and $v_k = z_{k,I}$ is compared with a threshold set to 0.

Performance of M-DPSK

With reference to the transmitted signal (18.1), we consider the isolated pulse k

$$s(t) = Re[e^{j\psi_k} h_{Tx}(t - kT) e^{j2\pi f_0 t}] \tag{18.13}$$

where

$$\psi_k = \psi_{k-1} + \theta_k \qquad \theta_k \in \left\{0, \frac{2\pi}{M}, \ldots, \frac{2\pi(M-1)}{M}\right\} \tag{18.14}$$

In general, referring to the scheme of Figure 18.7, the filtered signal is given by

$$x^{(bb)}(t) = u^{(bb)}(t) + w_R^{(bb)}(t) \tag{18.15}$$

where $u^{(bb)}$ is the desired signal at the demodulator input (isolated pulse k)

$$u^{(bb)}(t) = e^{j\psi_k} h_{Tx} * \tfrac{1}{2} g_{Ch}^{(bb)} * \tfrac{1}{2} g_A^{(bb)}(t - kT) \tag{18.16}$$

as $(1/2)g_{Ch}^{(bb)}$ is the complex envelope of the impulse response of the transmission channel, and $w_R^{(bb)}(t)$ is zero mean additive complex Gaussian noise with variance $\sigma^2 = 2N_0 B_{rn}$, where

$$\begin{aligned} B_{rn} &= \int_{-\infty}^{+\infty} |\mathcal{G}_A(f)|^2 \, df = \int_{-\infty}^{+\infty} |g_A(t)|^2 \, dt \\ &= \int_{-\infty}^{+\infty} \frac{1}{4} \left| \mathcal{G}_A^{(bb)}(f) \right|^2 \, df = \int_{-\infty}^{+\infty} \frac{1}{4} \left| g_A^{(bb)}(t) \right|^2 \, dt \end{aligned} \tag{18.17}$$

is the equivalent bilateral noise bandwidth.

We assume the following configuration: the transmit filter impulse response is given by

$$h_{Tx}(t) = \sqrt{\frac{2E_s}{T}} \operatorname{rect} \frac{t - T/2}{T} \tag{18.18}$$

the channel introduces only a phase offset

$$g_{Ch}^{(bb)}(t) = 2e^{j\varphi_a} \delta(t) \tag{18.19}$$

and the receive filter is matched to the transmit filter,

$$g_A^{(bb)}(t) = \sqrt{\frac{2}{T}} \operatorname{rect} \frac{t - T/2}{T} \tag{18.20}$$

At the sampling intervals $t_0 + kT = T + kT$, let

$$w_k = w_R^{(bb)}(t_0 + kT) \tag{18.21}$$

and

$$A = \sqrt{\frac{2E_s}{T}}\frac{1}{2}\sqrt{\frac{2}{T}}T = \sqrt{E_s} \tag{18.22}$$

then

$$x^{(bb)}(t_0 + kT) = Ae^{j(\psi_k+\varphi_a)} + w_k \tag{18.23}$$

with

$$E[|w_k|^2] = \sigma^2 = 2N_0B_{rn} = N_0 \tag{18.24}$$

since $B_{rn} = 1/2$. Moreover, it results in

$$\begin{aligned} z_k &= \tfrac{1}{2}x^{(bb)}(T + kT)x^{(bb)*}(T + (k-1)T) \\ &= \tfrac{1}{2}[Ae^{j(\psi_k+\varphi_a)} + w_k]\,[Ae^{-j(\psi_{k-1}+\varphi_a)} + w^*_{k-1}] \\ &= \frac{1}{2}A\Bigg[\underbrace{Ae^{j\theta_k}}_{\text{desired term}} + \underbrace{e^{j(\psi_k+\varphi_a)}w^*_{k-1} + e^{-j(\psi_{k-1}+\varphi_a)}w_k + \frac{w_kw^*_{k-1}}{A}}_{\text{disturbance}}\Bigg] \end{aligned} \tag{18.25}$$

The desired term is similar to that obtained in the coherent case, M-phases on a circle of radius $A = \sqrt{E_s}$. The variance of w_k is equal to N_0, and $\sigma_I^2 = N_0/2$. However, even ignoring the term $w_kw^*_{k-1}/A$, if w_k and w_{k-1} are statistically independent it results in

$$E[|e^{j(\psi_k+\varphi_a)}w^*_{k-1} + e^{-j(\psi_{k-1}+\varphi_a)}w_k|^2] = 2\sigma^2 = 2N_0 \tag{18.26}$$

There is an asymptotic penalty, that is for $E_s/N_0 \to \infty$, of 3 dB with respect to the coherent receiver case. Indeed, for a 4-DPSK, a more accurate analysis demonstrates that the penalty is only 2.3 dB for higher values of E_s/N_0 (see Section 6.5.1).

18.2.3 FM discriminator with integrate and dump filter (LDI)

The scheme of Figure 18.8 makes use of a limiter discriminator (LD), as for *frequency modulated* (FM) signals, followed by an integrator filter over a symbol period, or *integrate and dump* (I&D) filter. Ideally, the discriminator output provides the instantaneous frequency deviation of x, i.e.

$$\Delta f(t) = \frac{d}{dt}\Delta\varphi_x(t) = \Delta\dot{\varphi}_x(t) \tag{18.27}$$

Then, integrating (18.27) over a symbol period, we have

$$\int_{t_0+(k-1)T}^{t_0+kT} \Delta f(\tau)\,d\tau = \Delta\varphi_x(t_0 + kT) - \Delta\varphi_x(t_0 + (k-1)T) + 2n\pi \tag{18.28}$$

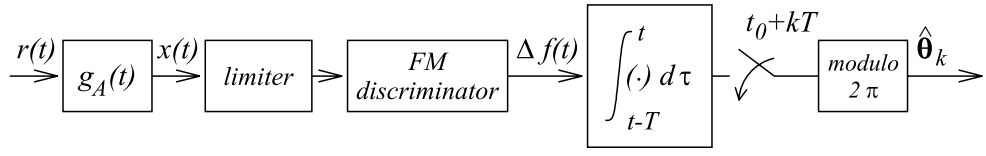


Figure 18.8. FM discriminator and integrate & dump filter.

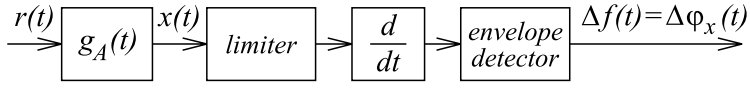


Figure 18.9. Implementation of a limiter-discriminator.

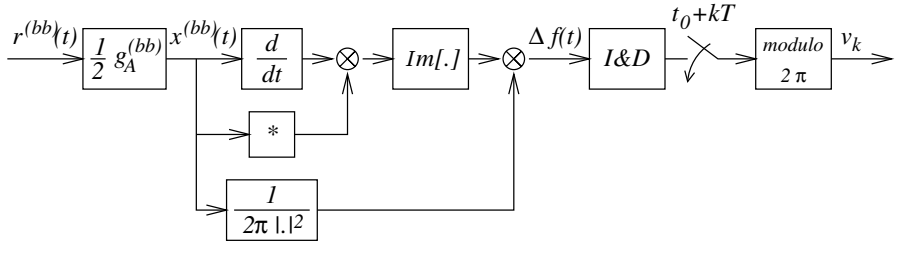


Figure 18.10. Baseband equivalent scheme of a FM discriminator followed by an integrate & dump filter.

that coincides with (18.11) taking mod 2π. An implementation of the limiter-discriminator is given in Figure 18.9, while the baseband equivalent scheme is given in Figure 18.10, which employs the general relation

$$\Delta f(t) = \frac{Im\left[\dot{x}^{(bb)}(t)x^{(bb)*}(t)\right]}{2\pi |x^{(bb)}(t)|^2} \tag{18.29}$$

In conclusion, all three schemes (baseband differential detector, differential detector and FM discriminator) yield the same output.

18.3 Variants of QPSK

18.3.1 Basic schemes

QPSK

From (18.1), we write the expression of the QPSK modulated signal

$$s(t) = \sum_{k=-\infty}^{+\infty} Re\left[e^{j\theta_k} h_{Tx}(t-kT) e^{j2\pi f_0 t} e^{j\varphi_0}\right] \tag{18.30}$$

where θ_k is the phase associated with the transmitted symbol at instant kT, and φ_0 is the carrier phase.

We denote as a_k the generic symbol at instant kT

$$a_k = e^{j\theta_k} \tag{18.31}$$

As $\theta_k \in \{\pm\pi/4, \pm 3\pi/4\}$, we get

$$a_k \in \frac{1}{\sqrt{2}}\{\pm 1, \pm j\} \tag{18.32}$$

For a modulation interval $T = 2T_b$, the map between bits and symbols is reproduced in Figure 18.11, with balanced symbols, $b_\ell \in \{-1, 1\}$. In particular, the following bit map is adopted,

$$Re\,[a_k] = b_{2k-1} \in \{-1, 1\} \quad \text{odd bits} \tag{18.33}$$

$$Im\,[a_k] = b_{2k} \in \{-1, 1\} \quad \text{even bits} \tag{18.34}$$

Let

$$I(t) = \sum_{k=-\infty}^{+\infty} b_{2k-1} h_{Tx}(t - kT) \tag{18.35}$$

$$Q(t) = \sum_{k=-\infty}^{+\infty} b_{2k} h_{Tx}(t - kT) \tag{18.36}$$

then s is given by

$$\begin{aligned} s(t) &= I(t)\cos(2\pi f_0 t + \varphi_0) - Q(t)\sin(2\pi f_0 t + \varphi_0) \\ &= Re\left[\sum_{k=-\infty}^{+\infty} a_k h_{Tx}(t - kT) e^{j(2\pi f_0 t + \varphi_0)}\right] \end{aligned} \tag{18.37}$$

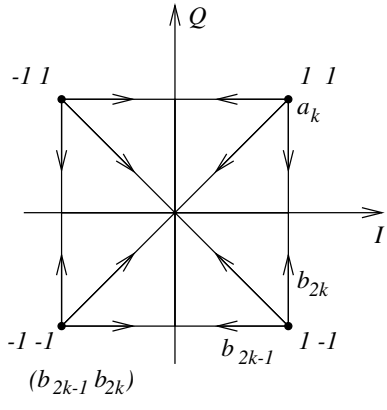


Figure 18.11. QPSK constellation with corresponding bit map and possible transitions of phase θ_k at instants kT.

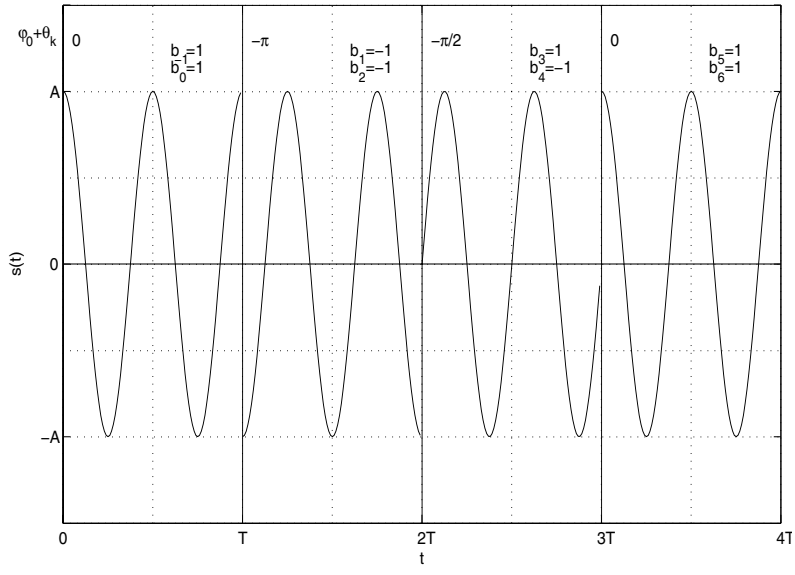


Figure 18.12. Realization of a QPSK signal with $f_0 = 2/T$ and $\varphi_0 = -\pi/4$.

From Figure 18.11 we note that at successive instants, the phase $\{\theta_k\}$ can undergo variations even equal to π; this implies large discontinuities in s with a consequent very high *peak-power/average-power* ratio of s if h_{Tx} is narrow band. In radio systems this can create saturation problems of the transmit amplifier (see Section 4.8). Figure 18.12 shows the possible behavior of s for a wideband modulation pulse

$$h_{Tx}(t) = A \,\mathrm{rect}\,\frac{t - T/2}{T} \tag{18.38}$$

for which

$$s(t) = A\cos(2\pi f_0 t + \varphi_0 + \theta_k) \qquad kT \le t < (k+1)T \tag{18.39}$$

In the figure we note phase jumps in s that are equal to π.

Offset QPSK or staggered QPSK

The in-phase signal is anticipated of $T_b = T/2$ with respect to the quadrature signal

$$I(t) = \sum_{k=-\infty}^{+\infty} b_{2k-1} h_{Tx}\left(t - \left(k - \frac{1}{2}\right)T\right) \tag{18.40}$$

$$Q(t) = \sum_{k=-\infty}^{+\infty} b_{2k} h_{Tx}(t - kT) \tag{18.41}$$

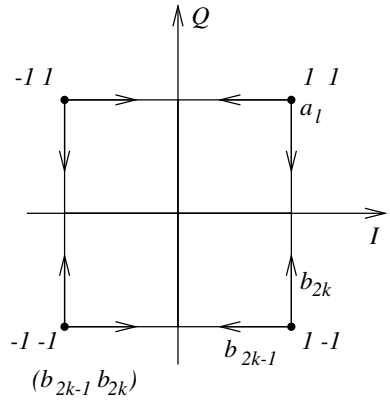


Figure 18.13. OQPSK constellation with corresponding bit map and possible transitions of phase θ_ℓ at instants ℓT_b.

and as usual we have

$$s(t) = I(t)\cos(2\pi f_0 t + \varphi_0) - Q(t)\sin(2\pi f_0 t + \varphi_0) \tag{18.42}$$

In other words, the signal phase is varied alternatively, over the two branches, every T_b seconds; therefore the phase variation is now at most $\pi/2$, as illustrated in Figure 18.13. For h_{Tx} as given by (18.38), we now have

$$s(t) = A\cos(2\pi f_0 t + \varphi_0 + \theta_\ell) \qquad \text{for } \ell T_b \le t < (\ell+1)T_b \tag{18.43}$$

For the same binary sequence of Figure 18.12, we show in Figure 18.14 the behavior of s given by (18.43). The reduced phase variation in s implies a greater compatibility with non-linear amplifiers. On the other hand, an OQPSK system requires a coherent demodulator.

Differential QPSK (DQPSK)

Recall that DQPSK was introduced in (6.157). The transmitted symbol at instant kT is given by

$$a_k = e^{j\psi_k} \tag{18.44}$$

where

$$\psi_k = \psi_{k-1} + \theta_k \qquad \theta_k \in \left\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\right\} \tag{18.45}$$

The information is mapped in the phase variation between two successive symbols. At the receiver a non-coherent differential demodulator (see Section 18.2) is commonly employed.

If we denote as $e^{j\theta_k}$ the information at instant kT, for the transmitted symbol a_k the recursive relation $a_k = a_{k-1}e^{j\theta_k}$ holds, from which we again obtain $e^{j\theta_k}$ by the equation $e^{j\theta_k} = a_k a_{k-1}^*$.

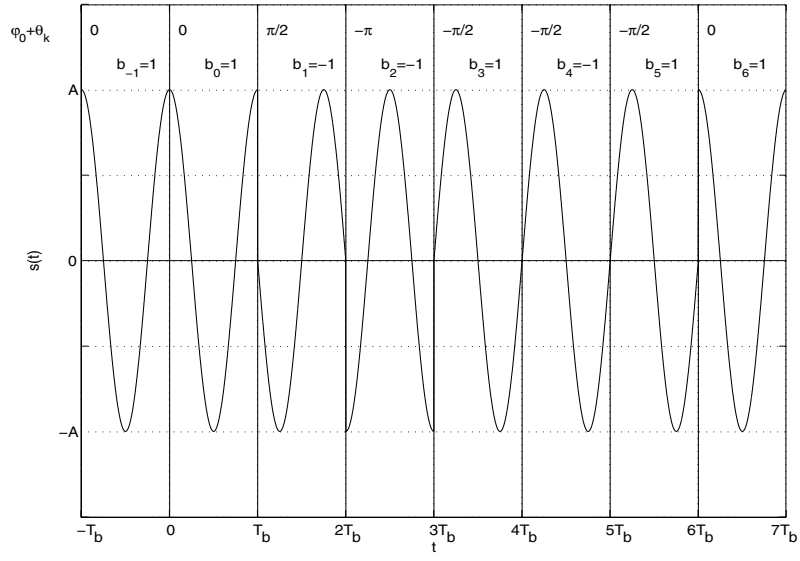


Figure 18.14. Realization of an OQPSK signal with $f_0 = 2/T$ and $\varphi_0 = -\pi/4$.

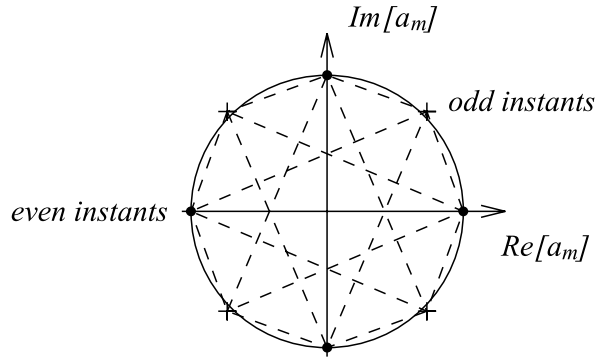


Figure 18.15. Possible values of the phase ψ_k for a $\pi/4$-DQPSK.

$\pi/4$-*DQPSK*

The transmitted symbol is $a_k = e^{j\psi_k}$, with

$$\psi_k = \psi_{k-1} + \theta_k \tag{18.46}$$

where now $\theta_k \in \{\pi/4, 3\pi/4, 5\pi/4, 7\pi/4\}$. Possible values of phase ψ_k are given in Figure 18.15.

Note that for k even, ψ_k assumes values in $\{0, \pi/2, \pi, 3\pi/2\}$, and for k odd $\psi_k \in \{\pi/4, 3\pi/4, 5\pi/4, 7\pi/4\}$. Phase variations between two consecutive instants are $\pm\pi/4$ and $\pm 3\pi/4$, with a good peak/average power ratio of the modulated signal.

18.3.2 Implementations

QPSK, OQPSK, and DQPSK modulators

Extending the scheme of Figure 6.34, we obtain the general modulator scheme for QPSK, OQPSK, and DQPSK illustrated in Figure 18.16. To implement each of these modulators it is sufficient to change the encoder and possibly add a delay of T_b on the Q branch for OQPSK [2]. Starting from the scheme of Figure 6.35, a coherent demodulator for QPSK and OQPSK is shown in Figure 18.17. In this scheme the two functions of carrier recovery (CR) and symbol timing recovery (STR) have been added, that determine the sampling instants $t_0 + k2T_b$ (+ possible T_b) for data detection. For OQPSK, on the I branch the delay T_b matches the delay introduced on the Q branch of the modulator.

Finally, with reference to the demodulator for DQPSK, the 1BDD non-coherent scheme of Figure 18.6 is usually employed.

$\pi/4$-DQPSK modulators

Due to its wide use, we examine in detail modulation and demodulation schemes for $\pi/4$-DQPSK. The modulator is similar to that for QPSK of Figure 18.16, differing only in the bit map. Now, for

$$a_k = a_{k,I} + j a_{k,Q} \tag{18.47}$$

from (18.46) a recursive equation for $a_{k,I}$ and $a_{k,Q}$ is given by

$$a_{k,I} = a_{k-1,I} \cos\theta_k - a_{k-1,Q} \sin\theta_k \tag{18.48}$$

$$a_{k,Q} = a_{k-1,Q} \cos\theta_k + a_{k-1,I} \sin\theta_k \tag{18.49}$$

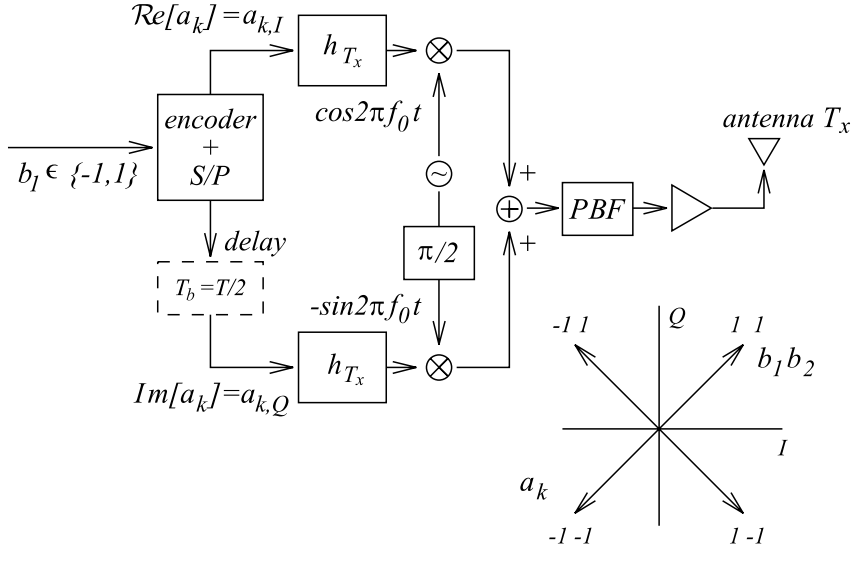


Figure 18.16. Modulator for QPSK, OQPSK, and DQPSK.

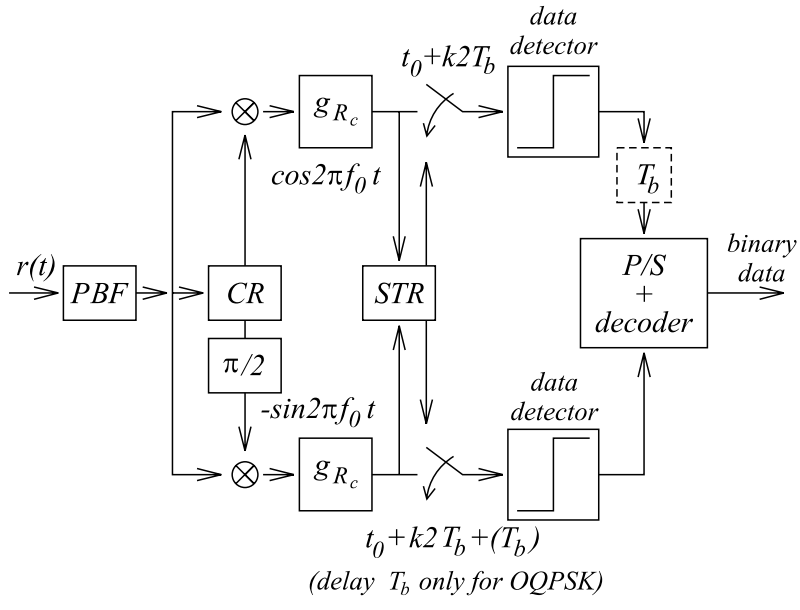


Figure 18.17. Coherent demodulator for QPSK and OQPSK (CR = carrier recovery, STR = symbol timing recovery).

As $\theta_k \in \{\pi/4, 3\pi/4, 5\pi/4, 7\pi/4\}$, then

$$e^{j\theta_k} \in \left\{ \frac{1+j}{\sqrt{2}}, \frac{-1+j}{\sqrt{2}}, \frac{-1-j}{\sqrt{2}}, \frac{1-j}{\sqrt{2}} \right\} \tag{18.50}$$

and

$$\begin{cases} a_{k,I}, a_{k,Q} \in \{0, \pm 1\} & k \text{ even} \\ a_{k,I}, a_{k,Q} \in \left\{ \pm \dfrac{1}{\sqrt{2}} \right\} & k \text{ odd} \end{cases} \tag{18.51}$$

To demodulate a $\pi/4$-DQPSK signal we can use one of the three differential schemes described in Section 18.2 or the coherent scheme illustrated in Figure 18.17. In any case, for an ideal AWGN channel performance of a $\pi/4$-DQPSK is the same as for QPSK using a coherent receiver (see (6.153)) with differential encoding or a differential receiver (see (6.164)).

For a 1BDD, for an ideal channel and using as transmit filter h_{Tx} and as receive filter $g_A^{(bb)}$ a square root raised cosine pulse with roll-off $\rho = 0.3$ ($\rho = 1$), Figure 18.18 (Figure 18.19) illustrates the eye diagram at the decision point.

We note that at the instant of maximum eye aperture the signal assumes the four possible values of $e^{j\theta_k}$ given by (18.50).

Figures 18.20 and 18.21 illustrate eye diagrams for a LDI demodulator that are obtained using the same filters as in the previous case. As indicated by (18.28), at the instant of maximum eye aperture, the signal assumes the four possible values of θ_k given by (18.46).

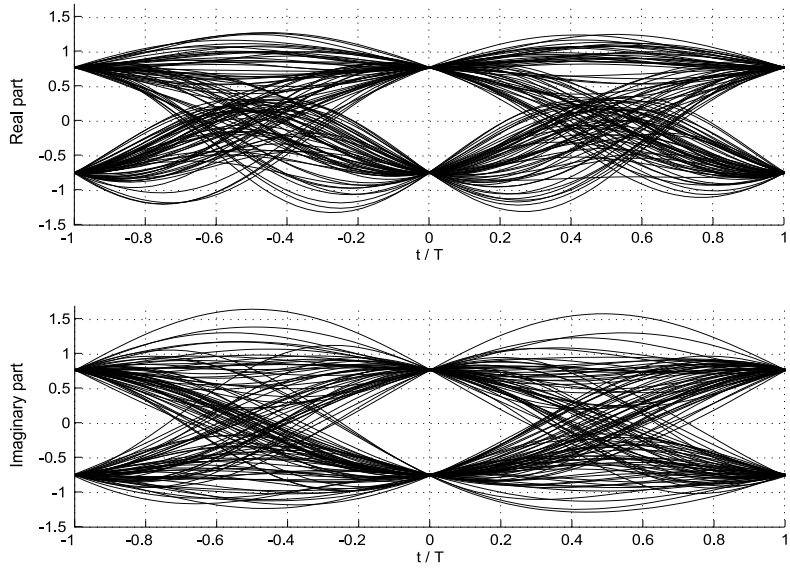


Figure 18.18. Eye diagram of the 1BDD for $\pi/4$-DQPSK, for an ideal channel and square root raised cosine receive filter with $\rho = 0.3$.

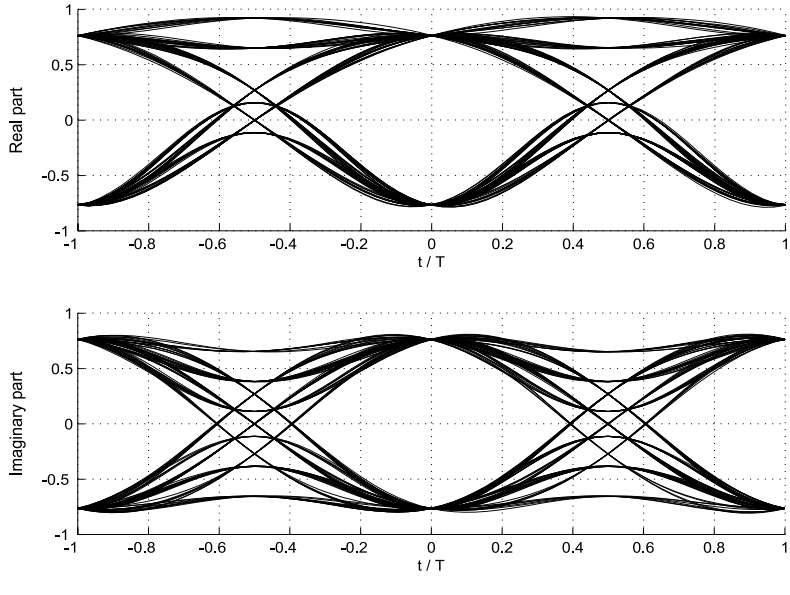


Figure 18.19. Eye diagram of the 1BDD for $\pi/4$-DQPSK, for an ideal channel and square root raised cosine receive filter with $\rho = 1$.

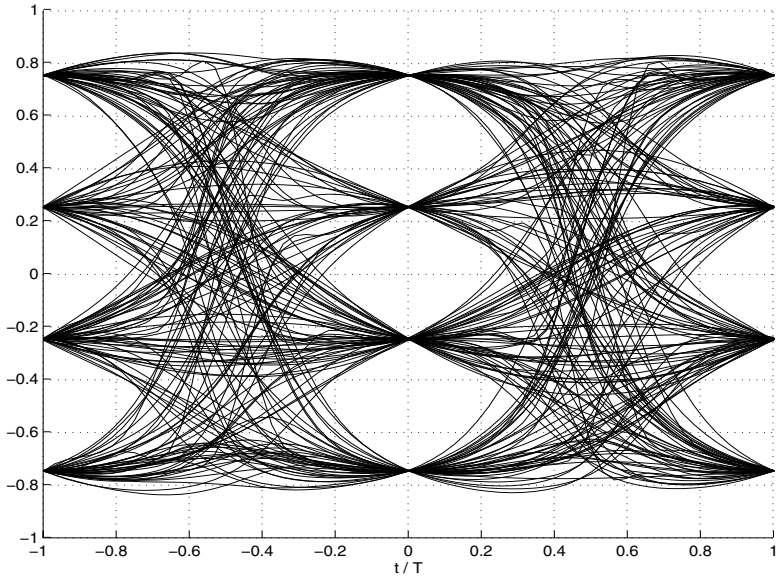


Figure 18.20. Eye diagram of the LDI demodulator for $\pi/4$-DQPSK, for an ideal channel and square root raised cosine receive filter with $\rho = 0.3$.

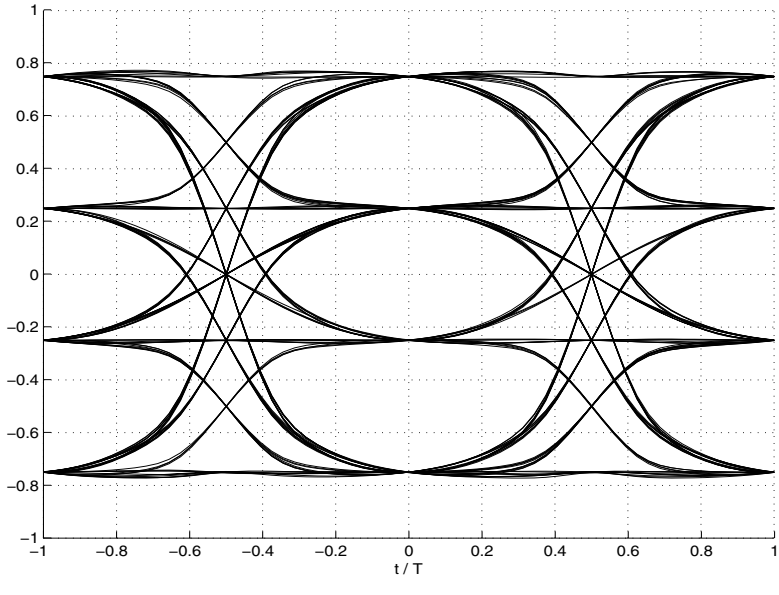


Figure 18.21. Eye diagram of the LDI demodulator for $\pi/4$-DQPSK, for an ideal channel and square root raised cosine receive filter with $\rho = 1$.

18.4 Frequency shift keying (FSK)

The main advantage of FSK modulators consists in generating a signal having a constant envelope, therefore the distortion introduced by a HPA is usually negligible. However, they present two drawbacks:

- wider spectrum of the modulated signal as compared to amplitude and/or phase modulated systems,
- complexity of the optimum receiver in the presence of non-ideal channels.

As already expressed in (6.72), a binary FSK modulator maps the information bits in frequency deviations ($\pm f_d$) around a carrier with frequency f_0; the possible transmitted waveforms are then given by

$$\begin{aligned} s_1(t) &= A\cos(2\pi(f_0 - f_d)t + \varphi_1) \\ s_2(t) &= A\cos(2\pi(f_0 + f_d)t + \varphi_2) \qquad kT < t < (k+1)T \end{aligned} \tag{18.52}$$

where, if we denote by E_s the average energy of an isolated pulse, $A = \sqrt{2E_s/T}$.

Figure 18.22 illustrates the generation of the above signals by two oscillators, with frequency $f_1 = f_0 - f_d$ and $f_2 = f_0 + f_d$, selected at instants kT by the variable $a_k \in \{1, 2\}$ related to the information bits. A particular realization of s_1 and s_2 is shown in Figure 18.23a. The resultant signal is given by

$$s(t) = \sum_{k=-\infty}^{+\infty} s_{a_k} \mathrm{w}_T(t - kT) \qquad a_k \in \{1, 2\} \tag{18.53}$$

A realization of s is shown in Figure 18.23b.

18.4.1 Power spectrum of M-FSK

Following the derivation in [3], we consider the two cases of non-coherent and coherent FSK.

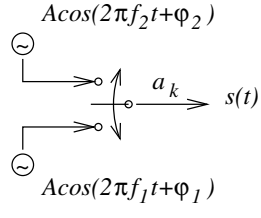


Figure 18.22. Generation of a binary (non-coherent) FSK signal by two oscillators.

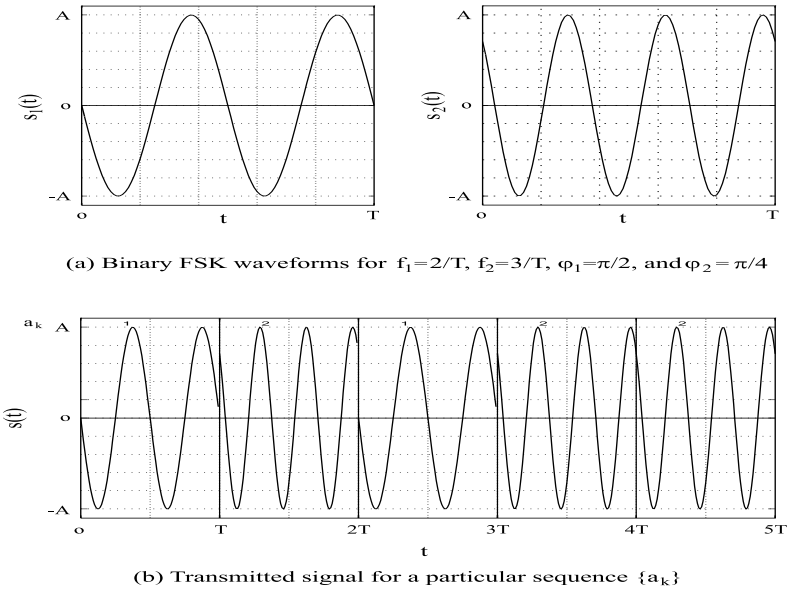


Figure 18.23. Binary FSK waveforms and transmitted signal for a particular sequence $\{a_k\}$.

Power spectrum of non-coherent binary FSK

In case the two oscillators maintain their original phases φ_1 and φ_2, we can express the modulated signal as

$$s(t) = Re\left[\sum_{k=-\infty}^{+\infty}\left(\frac{1+a_k}{2}A\,e^{j(2\pi f_1 t+\varphi_1)} + \frac{1-a_k}{2}A\,e^{j(2\pi f_2 t+\varphi_2)}\right)\mathrm{w}_T(t-kT)\right] \quad (18.54)$$

where $a_k \in \{-1, 1\}$ represents the information symbol. The complex envelope of s, with respect to the carrier f_0, is given by

$$\begin{aligned} s^{(bb)}(t) = {} & \frac{A}{2}[e^{-j(2\pi f_d t-\varphi_1)} + e^{j(2\pi f_d t+\varphi_2)}]\sum_{k=-\infty}^{+\infty}\mathrm{w}_T(t-kT) \\ & + \frac{A}{2}[e^{-j(2\pi f_d t-\varphi_1)} - e^{j(2\pi f_d t+\varphi_2)}]\sum_{k=-\infty}^{+\infty}a_k\mathrm{w}_T(t-kT) \end{aligned} \quad (18.55)$$

We note that

$$\sum_{k=-\infty}^{+\infty}\mathrm{w}_T(t-kT) = 1 \quad (18.56)$$

while the second term of (18.55) is a PAM signal; using the results of Example 7.1.1 on page 544 it is easy to derive the continuous and the discrete parts of the power spectrum of $s^{(bb)}$.

If $\mathtt{W}_T(f) = T\,\mathrm{sinc}(fT)e^{-j\pi fT}$, we get

$$\begin{aligned}\bar{\mathcal{P}}^{(c)}_{s^{(bb)}}(f) &= \frac{A^2}{4T}[|\mathtt{W}_T(f+f_d)|^2 + |\mathtt{W}_T(f-f_d)|^2 \\ &\quad - 2\gamma(h)Re[e^{j(\varphi_2-\varphi_1)}\mathtt{W}_T(f+f_d)\mathtt{W}^*_T(f-f_d)]] \\ \bar{\mathcal{P}}^{(d)}_{s^{(bb)}}(f) &= \frac{A^2}{4}[\delta(f+f_d)+\delta(f-f_d)]\end{aligned} \tag{18.57}$$

In (18.57), $h = 2f_dT$ is the modulation index, and

$$\gamma(h) = \begin{cases} 1 & h \text{ an integer} \\ 0 & \text{otherwise} \end{cases} \tag{18.58}$$

It can be seen that the power spectrum (18.57) contains different spurious components [3] and this is partially due to the fact that $\varphi_1 \neq \varphi_2$.

Particular case. Applying the previous expressions to the binary case with $\varphi_1 = \varphi_2 = 0$, we have[1]

$$\begin{aligned} s^{(bb)}_1(t) &= Ae^{-j2\pi f_dt}\mathtt{w}_T(t) \quad \xleftrightarrow{\mathcal{F}} \quad \mathcal{S}^{(bb)}_1(f) = A\mathtt{W}_T(f+f_d) \\ s^{(bb)}_2(t) &= Ae^{j2\pi f_dt}\mathtt{w}_T(t) \quad \xleftrightarrow{\mathcal{F}} \quad \mathcal{S}^{(bb)}_2(f) = A\mathtt{W}_T(f-f_d) \end{aligned} \tag{18.59}$$

From (18.57), the continuous part becomes

$$\bar{\mathcal{P}}^{(c)}_{s^{(bb)}}(f) = \frac{A^2}{4T}[|\mathtt{W}_T(f+f_d)|^2 + |\mathtt{W}_T(f-f_d)|^2 - 2\gamma(h)\,Re[\mathtt{W}_T(f+f_d)\mathtt{W}^*_T(f-f_d)]] \tag{18.60}$$

The discrete part is instead given by

$$\begin{aligned}\bar{\mathcal{P}}^{(d)}_{s^{(bb)}}(f) &= \frac{A^2}{4T^2}[|\mathtt{W}_T(f+f_d)|^2 + |\mathtt{W}_T(f-f_d)|^2 \\ &\quad + 2Re[\mathtt{W}_T(f+f_d)\mathtt{W}^*_T(f-f_d)]] \sum_{\ell=-\infty}^{+\infty}\delta(f-\ell/T)\end{aligned} \tag{18.61}$$

The behavior of $\frac{1}{A^2T}\bar{\mathcal{P}}^{(c)}_{s^{(bb)}}(f)$ is illustrated in Figure 18.24 for three values of the modulation index; the amplitude in dB of the spectral lines of $\frac{1}{A^2}\bar{\mathcal{P}}^{(d)}_{s^{(bb)}}(f)$ is listed in Table 18.1.

Power spectrum of coherent M-FSK

Signals in the various symbol intervals are assumed to be statistically independent; furthermore, *they always have the same phase* $\varphi_1 = \varphi_2 = \varphi$; therefore, the generation of s can be rather different from the method of Figure 18.22. Let $s^{(bb)}$ be the complex envelope of s, then the average power spectral density of s is given by (7.28). In general, let M be the number of possible transmitted waveforms, $\{s_i(t)\}$, $i = 1, \ldots, M$, having complex

[1] In any case, unless specific assumptions are made on f_1 and f_2, the phase of the transmitted signal can undergo discontinuities at the instants kT.

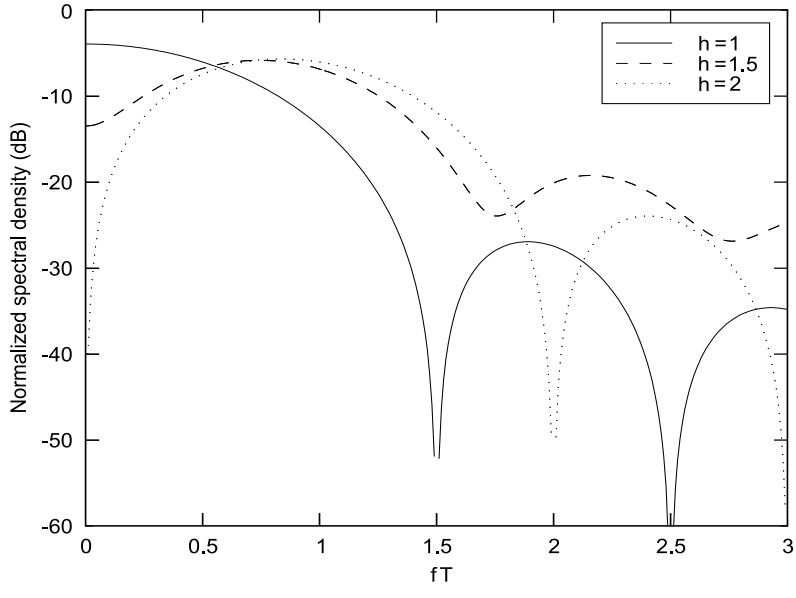


Figure 18.24. Continuous part of the power spectral density of a non-coherent binary FSK for three values of the modulation index.

Table 18.1 Amplitude in dB of several spectral lines of non-coherent binary FSK signals for three values of the modulation index.

	$h = 2f_dT$		
fT	1.0	1.5	2.0
0	—	−13.4	—
1	−7.4	−6.8	−6.0
2	−15.4	−20.1	—
3	−19.2	−24.6	—
4	−21.8	−27.5	—
5	−23.8	−29.6	—
6	−25.4	−31.3	—
7	−26.8	−32.7	—

envelope $\{s_i^{(bb)}(t)\}$. As will be derived next, the continuous and discrete parts of the power spectrum are given by

$$\bar{\mathcal{P}}^{(c)}_{s^{(bb)}}(f) = \frac{1}{MT}\left\{\sum_{i=1}^{M}|\mathcal{S}_i^{(bb)}(f)|^2 - \frac{1}{M}\left|\sum_{i=1}^{M}\mathcal{S}_i^{(bb)}(f)\right|^2\right\} \tag{18.62}$$

$$\bar{\mathcal{P}}^{(d)}_{s^{(bb)}}(f) = \left(\frac{1}{MT}\right)^2 \left|\sum_{i=1}^{M} \mathcal{S}_i^{(bb)}(f)\right|^2 \sum_{\ell=-\infty}^{+\infty} \delta\left(f - \frac{\ell}{T}\right) \tag{18.63}$$

We derive now the expressions (18.62) and (18.63). $s^{(bb)}$ is a cyclostationary process with autocorrelation

$$\begin{aligned} \mathtt{r}_{s^{(bb)}}(t, t-\tau) &= E[s^{(bb)}(t) s^{(bb)*}(t-\tau)] \\ &= \sum_{k_1=-\infty}^{\infty} \sum_{k_2=-\infty}^{\infty} E[s^{(bb)}_{a_{k_1}}(t-k_1T) s^{(bb)*}_{a_{k_2}}(t-\tau-k_2T)] \\ &= \sum_{k_1=-\infty}^{\infty} \sum_{\substack{k_2=-\infty \\ k_1 \neq k_2}}^{\infty} E[s^{(bb)}_{a_{k_1}}(t-k_1T) s^{(bb)*}_{a_{k_2}}(t-\tau-k_2T)] \\ &\quad + \sum_{k_1=-\infty}^{\infty} E[s^{(bb)}_{a_{k_1}}(t-k_1T) s^{(bb)*}_{a_{k_1}}(t-\tau-k_1T)] \end{aligned} \tag{18.64}$$

Assuming a_{k_1} statistically independent of a_{k_2} for $k_2 \neq k_1$, we have that

$$\begin{aligned} \mathtt{r}_{s^{(bb)}}(t, t-\tau) &= \sum_{k_1=-\infty}^{\infty} \sum_{\substack{k_2=-\infty \\ k_1 \neq k_2}}^{\infty} \left[\frac{1}{M} \sum_{i_1=1}^{M} s^{(bb)}_{i_1}(t-k_1T) \frac{1}{M} \sum_{i_2=1}^{M} s^{(bb)*}_{i_2}(t-\tau-k_2T)\right] \\ &\quad + \sum_{k_1=-\infty}^{\infty} \left[\frac{1}{M} \sum_{i=1}^{M} s^{(bb)}_{i}(t-k_1T) s^{(bb)*}_{i}(t-\tau-k_1T)\right] \end{aligned} \tag{18.65}$$

The average autocorrelation is given by

$$\begin{aligned} \bar{\mathtt{r}}_{s^{(bb)}}(\tau) &= \frac{1}{T} \int_0^T \mathtt{r}_{s^{(bb)}}(t, t-\tau)\, dt \\ &= \sum_{m=-\infty}^{+\infty} \frac{1}{TM^2} \sum_{i_1=1}^{M} \sum_{i_2=1}^{M} \mathtt{r}_{s^{(bb)}_{i_1} s^{(bb)}_{i_2}}(\tau + mT) \\ &\quad - \frac{1}{TM^2} \sum_{i_1=1}^{M} \sum_{i_2=1}^{M} \mathtt{r}_{s^{(bb)}_{i_1} s^{(bb)}_{i_2}}(\tau) + \frac{1}{TM} \sum_{i=1}^{M} \mathtt{r}_{s^{(bb)}_{i} s^{(bb)}_{i}}(\tau) \end{aligned} \tag{18.66}$$

where $\mathtt{r}^{(bb)}_{s_1 s_2}$ is the cross-correlation between $s^{(bb)}_1$ and $s^{(bb)}_2$, with Fourier transform $\mathcal{S}^{(bb)}_1(f)\mathcal{S}^{(bb)*}_2(f)$. The second term of (18.66) is the term for $m = 0$ of the first summation. Of the three terms in (18.66) we identify the first as a periodic signal of period T; it is this term that yields a discrete component in the power spectrum of $s^{(bb)}$. Taking the

Fourier transform of $\bar{r}_{s(bb)}(\tau)$ and considering the property

$$\sum_{m=-\infty}^{+\infty} x(\tau + mT) = \text{rep}_T x(\tau) \xleftrightarrow{\mathcal{F}} \frac{1}{T} \sum_{\ell=-\infty}^{+\infty} \mathcal{X}\left(\frac{\ell}{T}\right) \delta\left(f - \frac{\ell}{T}\right) \tag{18.67}$$

where $\mathcal{X}(f) = \mathcal{F}[x(\tau)]$, the result is given in (18.62) and (18.63).

18.4.2 FSK receivers and corresponding performance

We consider both coherent and non-coherent receiver types.

Coherent demodulator

A coherent demodulator is used when the transmitter provides all M signals with a well defined phase (see Example 6.7.1 on page 486); at the receiver there must be a circuit for the recovery of the phase of the various carriers. In practice this coherent receiver is rarely used because of its implementation complexity.

In any case, for orthogonal signals we can adopt the scheme of Figure 6.8, repeated in Figure 18.25 for the binary case.

For this case $\rho = 0$, and the bit error probability has already been derived in (6.71),

$$P_{bit} = Q\left(\sqrt{\frac{E_s}{N_0}}\right) \tag{18.68}$$

From (6.76), we note that the signals are orthogonal if

$$(2f_d)_{min} = \frac{1}{2T} \tag{18.69}$$

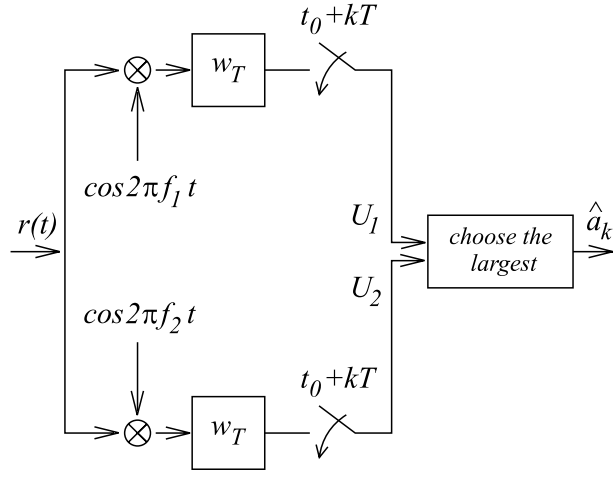


Figure 18.25. Coherent demodulator for orthogonal binary FSK.

Non-coherent demodulator

The transmitted waveforms can now have a different phase, and in any case unknown to the receiver (see Example 6.11.1 on page 509). For orthogonal signals, from the general scheme of Figure 6.59, we give in Figure 18.26 a possible non-coherent demodulator. In this case, for $\rho = 0$, the bit error probability has already been derived in (6.341),

$$P_{bit} = \frac{1}{2} e^{-\frac{E_s}{2N_0}} \tag{18.70}$$

From (6.306), we note that, in the non-coherent case, to have $\rho = 0$ it must be

$$(2f_d)_{min} = \frac{1}{T} \tag{18.71}$$

Therefore a non-coherent FSK system needs a double frequency deviation and has slightly lower performance (see Figure 6.62) as compared to coherent FSK, however, it does not need to acquire the carrier phase.

Limiter-discriminator FM demodulator

A simple non-coherent receiver, with good performance, for $f_1 T \gg 1$, $f_2 T \gg 1$ and sufficiently high E_s/N_0, is depicted in Figure 18.27.

After a passband filter to partially eliminate noise, a classical FM demodulator is used to extract the instantaneous frequency deviation of $r(t)$, equal to $\pm f_d$. Sampling at instants $t_0 + kT$ and using a threshold detector, we get the detected symbol $\hat{a}_k \in \{-1, 1\}$.

In general, for $2 f_d T = 1$, the performance of this scheme is very similar to that of a non-coherent orthogonal FSK.

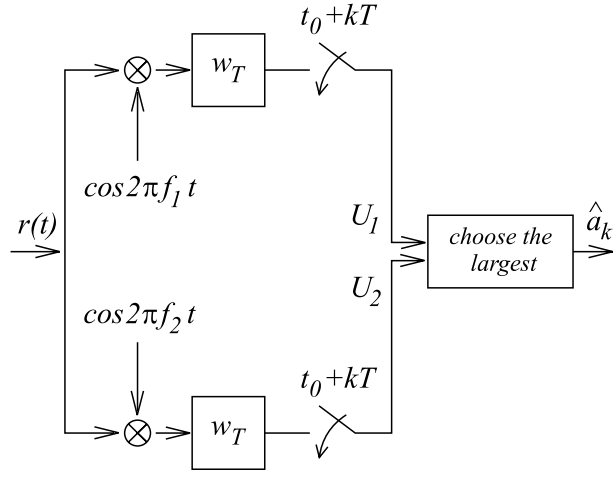


Figure 18.26. Non-coherent demodulator for orthogonal binary FSK.

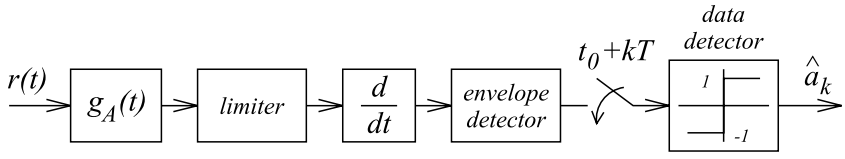


Figure 18.27. Limiter-discriminator FSK demodulator.

18.5 Minimum shift keying (MSK)

We recall the following characteristics of an ideal FSK signal.

- In order to avoid high frequency components in the power spectrum of s, the phase of an FSK signal should be continuous and not as represented in Figure 18.23. For signals as given by (18.52), with $\varphi_1 = \varphi_2$, this implies that it must be $f_n T = I_n$, $n = 1, 2$, I_n an integer.
- To minimize the error probability it must be $s_n(t) \perp s_m(t)$, $m \neq n$.
- To minimize the required bandwidth, the separation between the various frequencies must be minimum.

For signals as given by (18.52), choosing $|I_1 - I_2| = 1$ we have $f_2 = f_1 + 1/T$. In this case s_1 and s_2 are as shown in Figure 18.28.

Moreover, from (18.71) it is easy to prove that the signals are orthogonal. The result is that the frequency deviation is such that $2f_d = |f_2 - f_1| = 1/T$, and the carrier frequency is $f_0 = (f_1 + f_2)/2 = f_1 + 1/(2T)$.

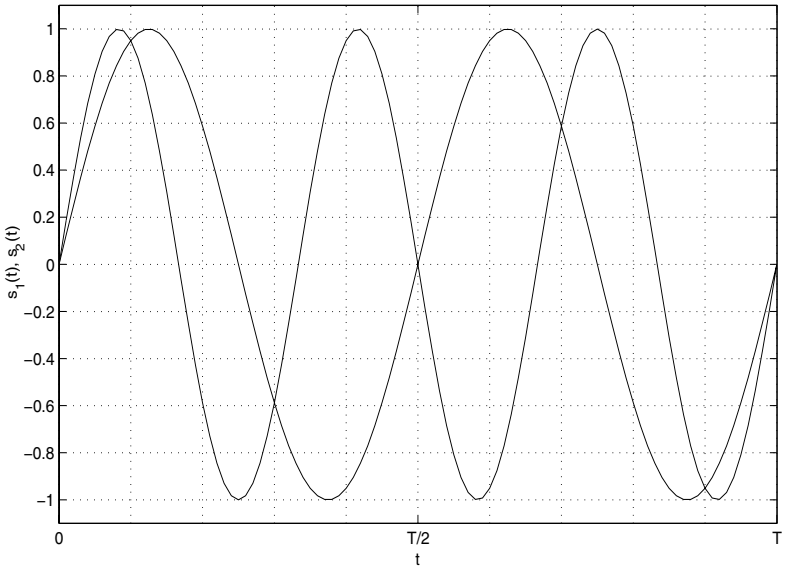


Figure 18.28. Waveforms as given by (18.52) with $A = 1$, $f_1 = 2/T$, $f_2 = 3/T$, and $\varphi_1 = \varphi_2 = 0$.

In a digital implementation of an FSK system, the following observation is very useful. Independent of f_1 and f_2, a method to obtain a phase continuous FSK signal given by

$$s(t) = A\cos(\varphi(t)) \tag{18.72}$$

where

$$\varphi(t) = 2\pi f_0 t + a_k 2\pi f_d t + \eta_k \qquad a_k \in \{-1, 1\} \qquad kT \le t < (k+1)T \tag{18.73}$$

is to employ a single oscillator, whose phase satisfies the constraint $\varphi((k+1)T^{(-)}) = \varphi((k+1)T^{(+)})$; thus it is sufficient that at the beginning of each symbol interval η_{k+1} is set equal to

$$\eta_{k+1} = ((a_k - a_{k+1})2\pi f_d (k+1)T + \eta_k) \bmod 2\pi \tag{18.74}$$

An alternative method is given by the scheme of Figure 18.29, in which the sequence $\{a_k\}$, with binary elements in $\{-1, 1\}$, is filtered by g to produce a PAM signal

$$x_f(t) = \sum_{k=-\infty}^{+\infty} a_k g(t - kT) \qquad a_k \in \{-1, 1\} \tag{18.75}$$

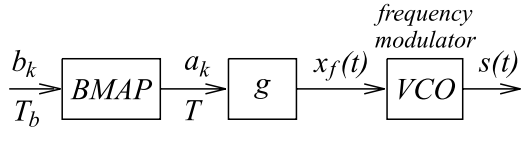
The signal x_f is input to a VCO, whose output is given by

$$s(t) = A\cos\left(2\pi f_0 t + 2\pi h \int_{-\infty}^{t} x_f(\tau)\, d\tau\right) \tag{18.76}$$

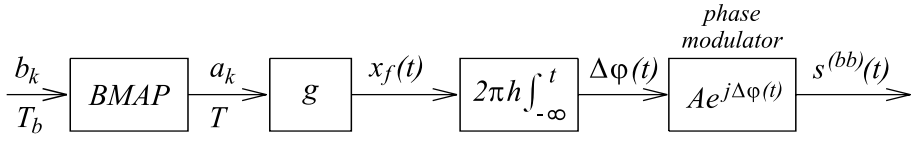
In (18.76), f_0 is the carrier, $h = 2f_d T$ is the modulation index and

$$\varphi(t) = 2\pi f_0 t + 2\pi h \int_{-\infty}^{t} x_f(\tau)\, d\tau \tag{18.77}$$

represents the phase of the modulated signal.



(a) Passband model



(b) Baseband equivalent model

Figure 18.29. CPFSK modulator.

Choosing for g a rectangular pulse

$$g(t) = \frac{1}{2T} \mathrm{w}_T(t) \tag{18.78}$$

we obtain a modulation scheme called *continuous phase* FSK (CPFSK). In turn, CPFSK is a particular case of the *continuous phase modulation* described in Appendix 18.A.

We note that the area of $g(t)$ is equal to 0.5; in this case the information is in the instantaneous frequency of s, in fact

$$f_s(t) = \frac{1}{2\pi}\frac{d\varphi(t)}{dt} = f_0 + h x_f(t) = f_0 + \frac{h a_k}{2T} \qquad kT < t < (k+1)T \tag{18.79}$$

The *minimum shift keying* (MSK) modulation is equivalent to CPFSK with modulation index $h = 0.5$, that is for $f_d = 1/(4T)$. Summarizing: an MSK scheme is a binary FSK scheme ($T_b = T$) in which, besides having $f_d = 1/(4T)$, the modulated signal has continuous phase.

Figure 18.30 illustrates a comparison between an FSK signal with $f_d = 1/(4T)$ as given by (18.52), and an MSK signal for a binary sequence $\{a_k\}$ equal to $\{-1, 1, 1, -1\}$. Note that in the MSK case, it is like having four waveforms $s_1, -s_1, s_2, -s_2$, each pair related to $a_k = 1$ and $a_k = -1$, respectively. Signal selection within the pair is done in a way to guarantee the phase continuity of s.

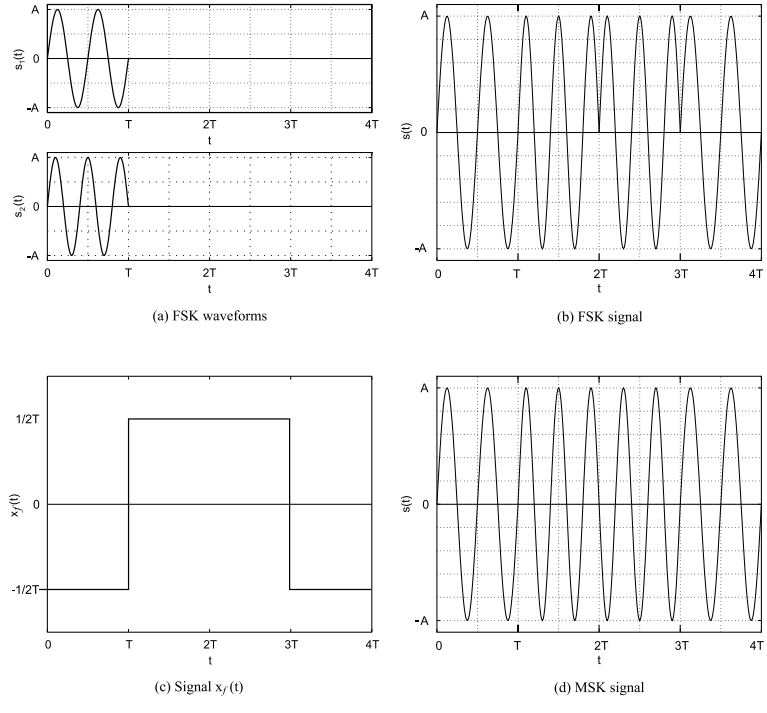


Figure 18.30. Comparison between FSK and MSK signals.

18.5.1 Power spectrum of continuous-phase FSK (CPFSK)

From the expressions given in [3], the behavior of the continuous part of the power spectral density is represented in Figure 18.31. Note that for higher values of h a peak will tend to emerge around $f_d = h/(2T)$, showing the instantaneous frequency of waveforms.

Comparing the spectra of Figure 18.31 with those of Figure 18.24 it is seen that phase continuity implies a lower bandwidth, at least if we use the definition of bandwidth based on a given percentage of signal power.

18.5.2 The MSK signal viewed from two perspectives

First, we consider the following aspect.

Phase of an MSK signal

For an MSK signal, the pulse

$$q(t) = \int_{-\infty}^{t} g(\tau)\, d\tau \tag{18.80}$$

is illustrated in Figure 18.32.

From (18.76) the phase deviation of the modulated signal is given by

$$\Delta\varphi(t) = \varphi(t) - 2\pi f_0 t = \pi \sum_{k=-\infty}^{+\infty} a_k q(t - kT_b) \tag{18.81}$$

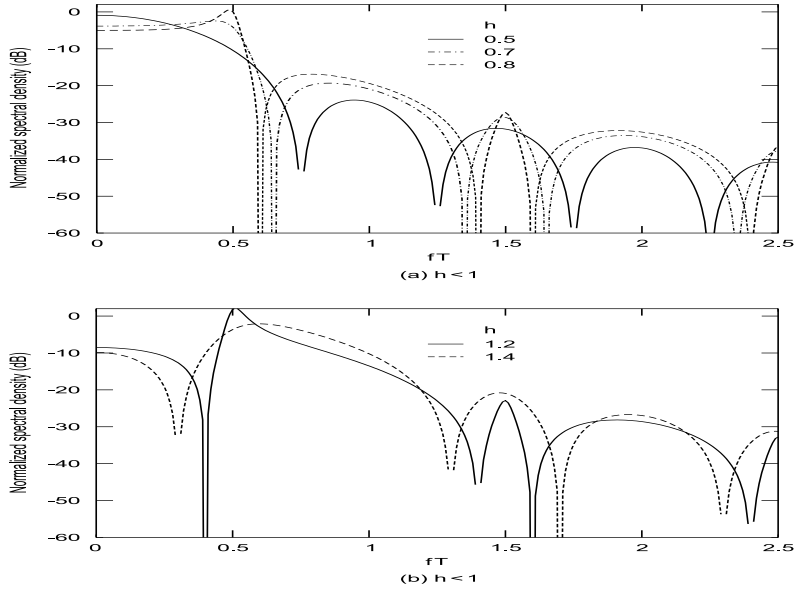


Figure 18.31. Continuous part of the power spectral density of a CPFSK signal for five values of the modulation index.

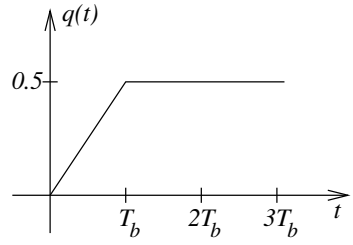


Figure 18.32. Pulse $q(t)$.

Let

$$\psi_{k-1} = \frac{\pi}{2} \sum_{i=-\infty}^{k-1} a_i \tag{18.82}$$

then it follows

$$\Delta\varphi(t) = a_k \frac{\pi}{2} \left(\frac{t}{T_b} - k \right) + \psi_{k-1} \qquad kT_b \le t < (k+1)T_b \tag{18.83}$$

For $t = (k+1)T_b^{(-)}$ we get

$$\Delta\varphi((k+1)T_b^{(-)}) = a_k \frac{\pi}{2} + \psi_{k-1} = \psi_k \tag{18.84}$$

Assuming $\psi_{-1} = 0$, the distinct values assumed by $\psi_k \bmod 2\pi$ are only four:

$$\psi_k = \begin{cases} \dfrac{\pi}{2}, -\dfrac{\pi}{2} & \text{for } k \text{ even} \\ 0, -\pi & \text{for } k \text{ odd} \end{cases} \tag{18.85}$$

From (18.84) we note how the information a_k is also contained in the variation, in two successive instants, of the phase deviation $\Delta\varphi$. The possible trajectories of $\Delta\varphi$ are shown in Figure 18.33.

From (18.81), the complex envelope of the modulated signal is given by

$$s^{(bb)}(t) = Ae^{j\Delta\varphi(t)} \tag{18.86}$$

where $\Delta\varphi(t)$ is related to the message by (18.83). Correspondingly the modulated signal (18.76) becomes

$$s(t) = I(t)\cos(2\pi f_0 t) - Q(t)\sin(2\pi f_0 t) \tag{18.87}$$

where

$$I(t) = Re[s^{(bb)}(t)] \qquad Q(t) = Im[s^{(bb)}(t)] \tag{18.88}$$

The signal

$$s^{(bb)}(t) = I(t) + jQ(t) \tag{18.89}$$

is represented in Figure 18.34. We note that the phase $\Delta\varphi(t)$ continuously varies in time and assumes the values $\{0, \pi/2, \pi, 3\pi/2\}$ at instants kT_b.

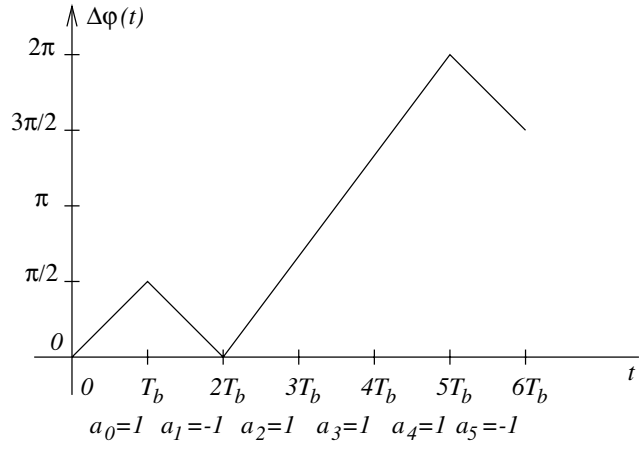


Figure 18.33. Possible trajectories of the phase deviation $\Delta\varphi(t)$.

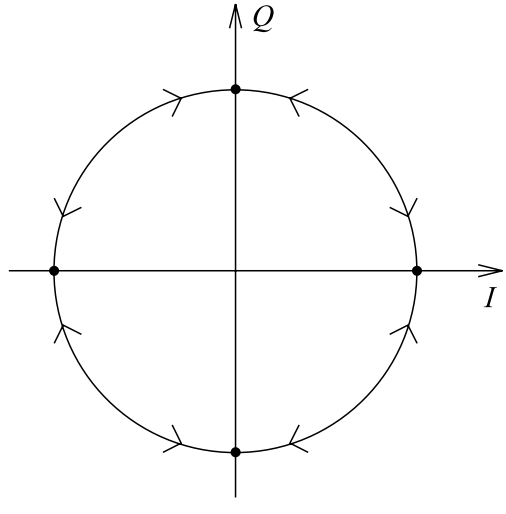


Figure 18.34. MSK: values assumed by $I(t) + jQ(t)$.

MSK as binary CPFSK

From (18.73) for $f_d = 1/(4T)$ and $T = T_b$ we obtain

$$s(t) = A\cos\left(2\pi\left(f_0 + \frac{a_k}{4T_b}\right)t + \eta_k\right) \qquad \text{per } kT_b \le t < (k+1)T_b \tag{18.90}$$

and in general

$$s(t) = A\sum_{k=-\infty}^{+\infty} \mathrm{w}_{T_b}(t - kT_b)\cos\left(2\pi\left(f_0 + \frac{a_k}{4T_b}\right)t + \eta_k\right) \tag{18.91}$$

from which we can observe that an MSK signal is a binary FSK signal with frequencies $f_1 = f_0 - 1/(4T)$ and $f_2 = f_0 + 1/(4T)$, where f_0 is the carrier frequency.

In (18.91) the symbols of the sequence $\{a_k\}$, $a_k \in \{-1, 1\}$, are information symbols, and

$$\begin{aligned} \eta_k &= (a_{k-1} - a_k)2\pi \frac{1}{4T_b} kT_b + \eta_{k-1} \bmod 2\pi \\ &= (a_{k-1} - a_k)\frac{\pi}{2} k + \eta_{k-1} \bmod 2\pi \end{aligned} \tag{18.92}$$

In particular we note that

$$\eta_k = \begin{cases} \begin{cases} \eta_{k-1} & a_k = a_{k-1} \\ \eta_{k-1} \pm 2\pi & a_k \neq a_{k-1} \end{cases} & \text{for } k \text{ even} \\ \begin{cases} \eta_{k-1} & a_k = a_{k-1} \\ \eta_{k-1} \pm \pi & a_k \neq a_{k-1} \end{cases} & \text{for } k \text{ odd} \end{cases} \tag{18.93}$$

hence $\eta_k \in \{0, \pi\}$.

From the comparison of (18.90) with (18.83) it is easy to derive the relation between η_k and ψ_k,

$$\eta_k = \psi_{k-1} - k a_k \frac{\pi}{2} \tag{18.94}$$

MSK as OQPSK

As $a_k \in \{-1, 1\}$, it is easy to verify that

$$\sin \eta_k = 0 \qquad \cos\left(a_k \frac{\pi t}{2T_b}\right) = \cos \frac{\pi t}{2T_b} \qquad \sin\left(a_k \frac{\pi t}{2T_b}\right) = a_k \sin \frac{\pi t}{2T_b} \tag{18.95}$$

Moreover, from (18.91) we obtain

$$\begin{aligned} I(t) &= A \sum_k \mathrm{w}_{T_b}(t - kT_b) \cos\left(a_k \frac{\pi t}{2T_b} + \eta_k\right) \\ &= A \sum_k \mathrm{w}_{T_b}(t - kT_b) \left(\cos \eta_k \cdot \cos \frac{\pi t}{2T_b} + 0\right) \\ Q(t) &= A \sum_k \mathrm{w}_{T_b}(t - kT_b) \sin\left(a_k \frac{\pi t}{2T_b} + \eta_k\right) \\ &= A \sum_k \mathrm{w}_{T_b}(t - kT_b) \left(a_k \cos \eta_k \cdot \sin \frac{\pi t}{2T_b} + 0\right) \end{aligned} \tag{18.96}$$

As the signal s is continuous phase by construction, from (18.87) I and Q must be continuous functions; therefore

1. $\cos \eta_k$ can change only for k odd, that is at instants $(k\,2T_b + T_b)$ in which $\cos\left(\frac{\pi t}{2T_b}\right)$ vanishes;

2. $a_k \cos\eta_k$ can change only for k even, that is at instants $(k\,2T_b)$ in which $\sin\left(\frac{\pi t}{2T_b}\right)$ vanishes.

Therefore, we note that the information symbols associated with I and Q components can change only every $2T_b$ and there is a lag T_b between the two branches. Defining the variable

$$c_k = \begin{cases} \cos\eta_k & \text{for } k \text{ odd} \\ a_k \cos\eta_k & \text{for } k \text{ even} \end{cases} \tag{18.97}$$

from (18.93) the following relations hold:

$$c_k = a_k \cos\eta_{k-1} = a_k c_{k-1} \qquad \text{for } k \text{ even} \tag{18.98}$$

$$\begin{aligned} c_k &= \left\{ \begin{array}{ll} \cos\eta_{k-1} & \text{for } a_k = a_{k-1} \\ -\cos\eta_{k-1} & \text{for } a_k \neq a_{k-1} \end{array} \right\} \\ &= a_k a_{k-1} \cos\eta_{k-1} = a_k c_{k-1} \qquad \text{for } k \text{ odd} \end{aligned} \tag{18.99}$$

Indeed, the transformation that maps the bits of $\{a_k\}$ into $\{c_k\}$ corresponds to a *differential encoder* given by[2]

$$c_k = c_{k-1} a_k \qquad a_k \in \{-1, 1\} \tag{18.100}$$

with $c_k \in \{-1, 1\}$. Then the decoding rule is

$$a_k = c_k c_{k-1} \tag{18.101}$$

From the previous observations, (18.96) becomes

$$\begin{aligned} I(t) &= A \sum_{\ell=-\infty}^{+\infty} c_{2\ell-1} \mathrm{w}_{2T_b}(t - 2\ell T_b + T_b) \cos\left(\frac{\pi t}{2T_b}\right) \\ Q(t) &= A \sum_{\ell=-\infty}^{+\infty} c_{2\ell} \mathrm{w}_{2T_b}(t - 2\ell T_b) \sin\left(\frac{\pi t}{2T_b}\right) \end{aligned} \tag{18.102}$$

Another representation of (18.102) is obtained by recognizing the periodic behavior of the waveforms $\sin(\pi t/2T_b)$ and $\cos(\pi t/2T_b) = \sin(\pi(t + T_b)/2T_b)$, illustrated in Figure 18.35. If we window these waveforms to the intervals $(0, 2T_b)$ and $(-T_b, T_b)$, respectively, in the other intervals it is sufficient to alternate the sign of the encoded symbol, and

$$\begin{aligned} I(t) &= A \sum_{\ell=-\infty}^{+\infty} (-1)^{\ell} c_{2\ell-1} \mathrm{w}_{2T_b}(t - 2\ell T_b + T_b) \sin\left(\frac{\pi(t - \ell 2T_b + T_b)}{2T_b}\right) \\ Q(t) &= A \sum_{\ell=-\infty}^{+\infty} (-1)^{\ell} c_{2\ell} \mathrm{w}_{2T_b}(t - 2\ell T_b) \sin\left(\frac{\pi(t - 2\ell T_b)}{2T_b}\right) \end{aligned} \tag{18.103}$$

[2] It corresponds to the exclusive OR (6.166) if $a_k \in \{0, 1\}$.

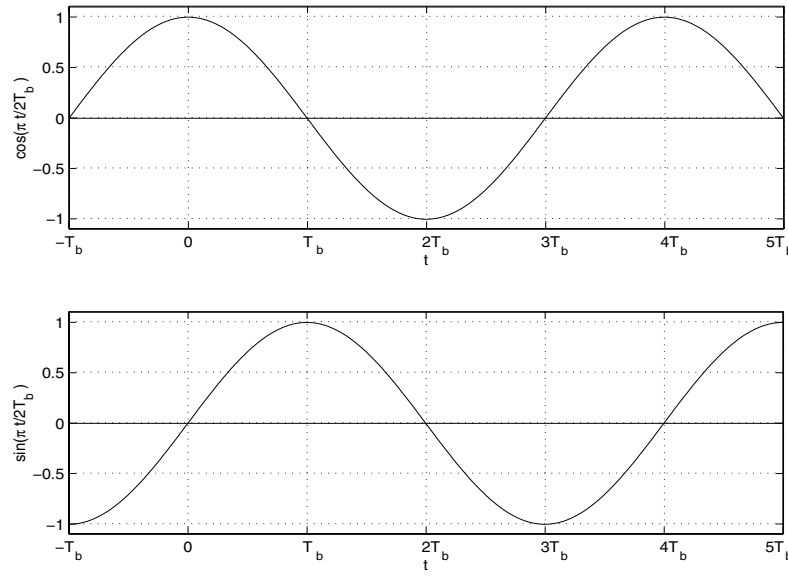


Figure 18.35. Behavior of two sinusoidal waveforms.

Then it follows that an MSK scheme is an OQPSK scheme with modulation interval $T = 2T_b$ and pulse

$$h_{Tx}(t) = A \,\mathrm{rect}\,\frac{t - T_b}{2T_b} \sin\left(\frac{\pi t}{2T_b}\right) \tag{18.104}$$

given in Figure 18.36, with Fourier transform

$$\mathcal{H}_{Tx}(f) = A\frac{4T_b \cos(2\pi f T_b)}{\pi(1 - (4fT_b)^2)} e^{-j2\pi f T_b} \tag{18.105}$$

We note that the transmitted symbols associated with the OQPSK interpretation are encoded with a suitable sign.

The example in Table 18.2 shows how the sequence $\{a_k\}$ is mapped into $\{c_k\}$ and to the data on the I and Q branches. The modulated signals are shown in Figure 18.37; note the phase continuity of s.

Complex notation of an MSK signal

A compact notation of (18.103) is given by

$$s^{(bb)}(t) = \sum_{k=-\infty}^{+\infty} j^{k+1} c_k h_{Tx}(t - kT_b) \tag{18.106}$$

where h_{Tx} is defined in (18.104).

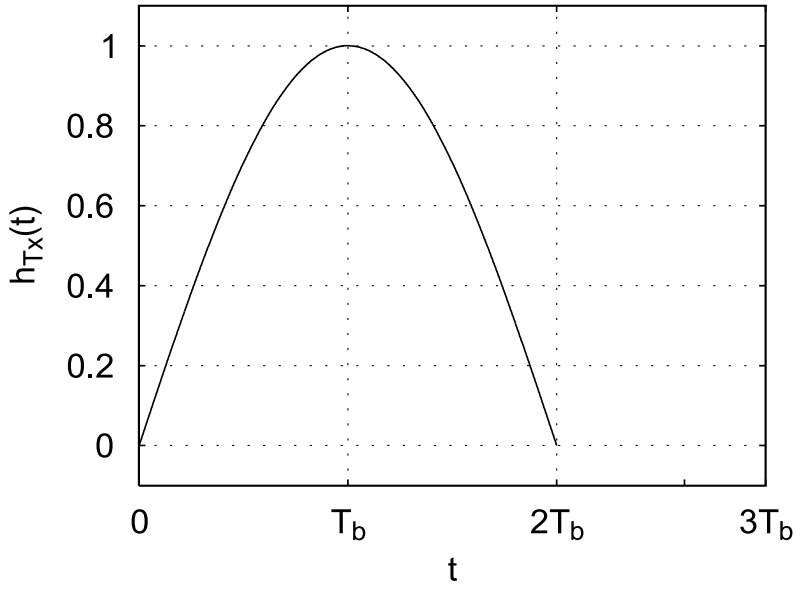


Figure 18.36. Shape of the fundamental MSK pulse for $A = 1$.

Table 18.2 Example of a coded sequence associated with MSK transmission.

a_k	1	-1	1	-1	-1	1	-1	-1
c_k	1	-1	-1	1	-1	-1	1	-1
I data	1		1		-1		-1	
Q data		-1		-1		-1		1

Furthermore, as for $a_k \in \{-1, 1\}$ it is $e^{j(\pi/2)a_k} = ja_k$, then from (18.100) we get

$$j^{k+1}c_k = \exp\left(j\frac{\pi}{2}\sum_{i=-\infty}^{k} a_i\right) = e^{j\psi_k} \tag{18.107}$$

using (18.82). Hence an alternative expression for (18.106) is given by

$$s^{(bb)}(t) = \sum_{k=-\infty}^{+\infty} e^{j\psi_k} h_{Tx}(t - kT_b) \tag{18.108}$$

We note the recursive structure of $e^{j\psi_k}$,

$$e^{j\psi_k} = e^{j\psi_{k-1}} ja_k \tag{18.109}$$

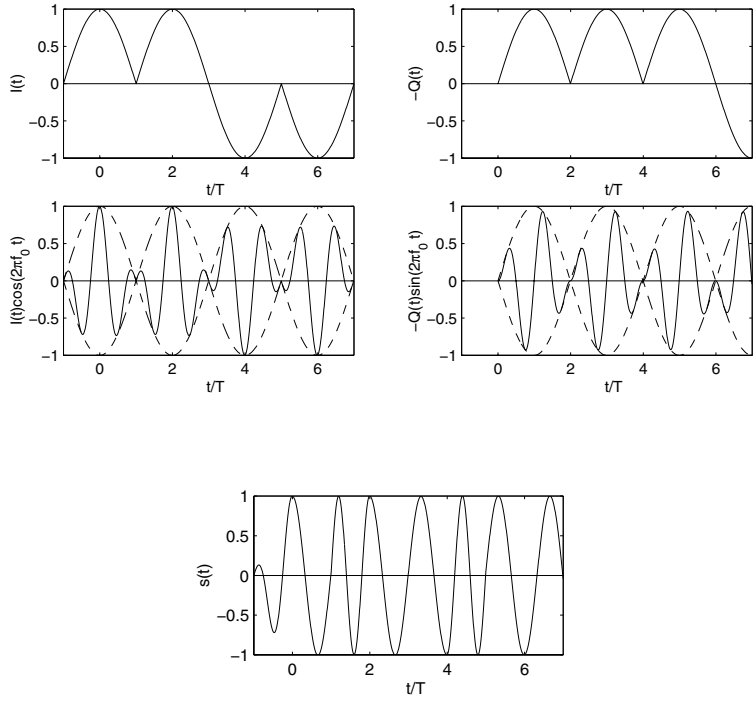


Figure 18.37. MSK signal for the data sequence of Table 18.2. Using the modulated signals *I* and *Q*, *s* is formed by (18.87).

18.5.3 Implementations of an MSK scheme

Interpreting an MSK scheme as a CPFSK scheme, the modulator is as shown in Figure 18.29, with

$$g(t) = \frac{1}{2T_b} \mathrm{w}_{T_b}(t) \tag{18.110}$$

As a whole the modulator is represented in Figure 18.38. Interpreting instead an MSK scheme as an OQPSK scheme, we have the implementation of Figure 18.16 in which $T = 2T_b$, $h_{Tx}(t)$ is given by (18.104), and data are encoded and changed in sign according to (18.103). Considering its importance and widespread usage, this scheme is illustrated in Figure 18.39.

18.5.4 Performance of MSK demodulators

Among the different demodulation schemes listed in Figure 18.40, we show in Figure 18.41 a differential 1BDD non-coherent demodulator. This scheme is based on the fact that the phase deviation of an MSK signal can vary of $\pm\pi/2$ between two suitably instants spaced

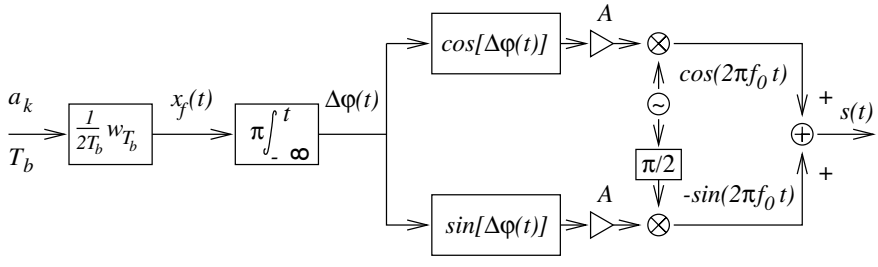


Figure 18.38. MSK as CPFSK.

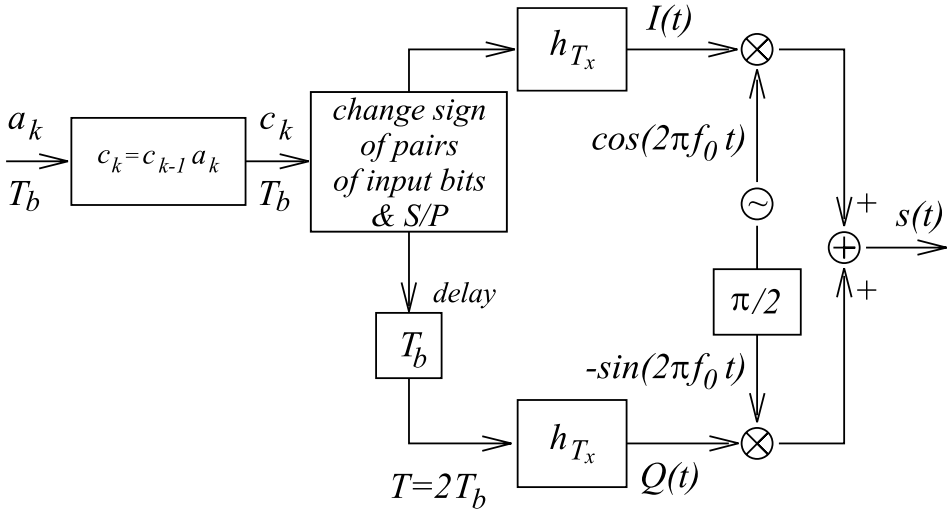


Figure 18.39. MSK as OQPSK.

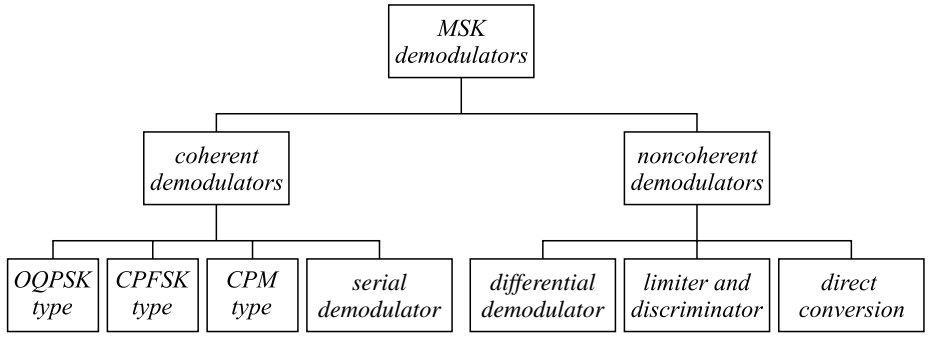
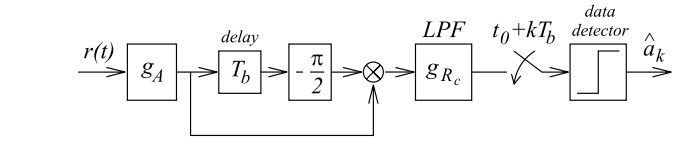
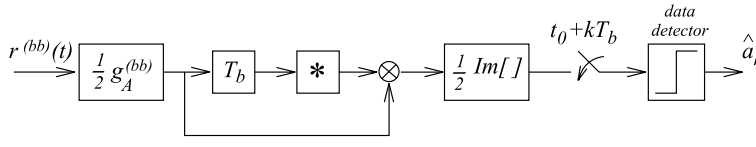


Figure 18.40. MSK demodulator classification.



(a) Passband model



(b) Baseband equivalent model

Figure 18.41. Differential (1BDD) non-coherent MSK demodulator.

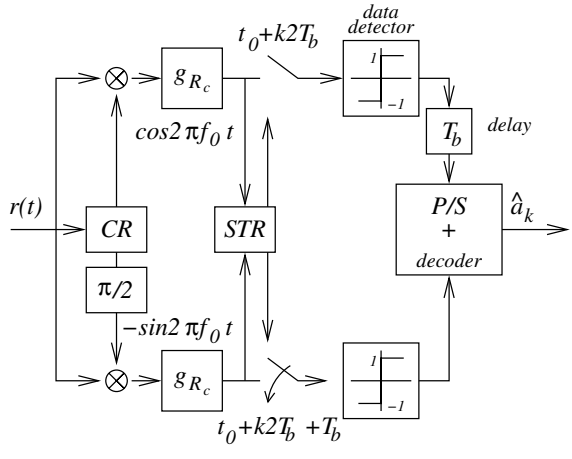


Figure 18.42. Coherent (OQPSK type) MSK demodulator.

of T_b (see (18.85)). In any case, the performance for an AWGN channel is that of a DBPSK scheme, but with half the phase variation; hence from (6.163), with E_s that becomes $E_s/2$, we get

$$P_{bit} = \frac{1}{2} e^{-\frac{E_s}{2N_0}} \tag{18.111}$$

Note that this is also the performance of a non-coherent orthogonal binary FSK scheme (see (18.70)).

In Figure 18.42 we illustrate a coherent (OQPSK type) demodulator that at alternate instants on the I branch and on the Q branch is of the BPSK type. In this case, from

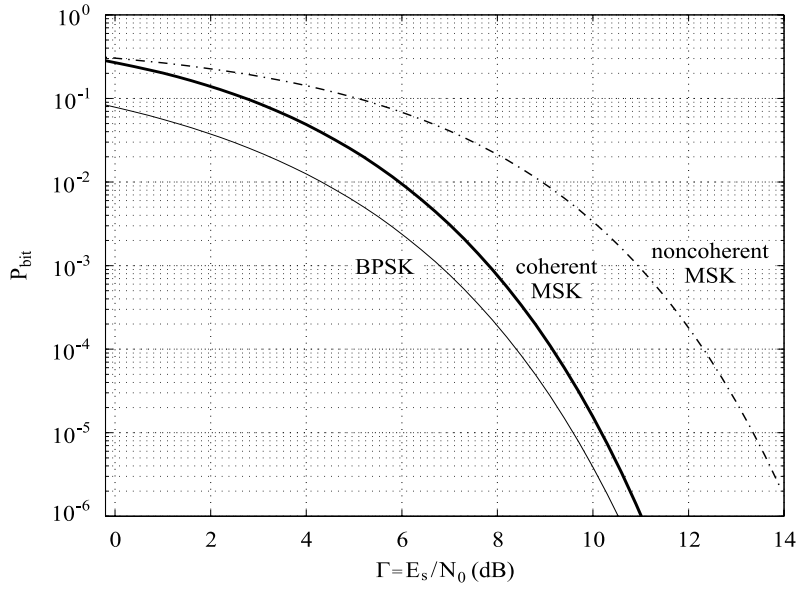


Figure 18.43. Comparison among various error probabilities.

Table 18.3 Increment of Γ (in dB) for an MSK scheme with respect to a coherent BPSK demodulator for $P_{bit} = 10^{-3}$.

	coherent BPSK	coherent MSK	non-coherent MSK
Γ	6.79	+1.1	+4.2

(6.151), the error probability for decisions on the symbols $\{c_k\}$ is given by

$$P_{bit,Ch} = Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \tag{18.112}$$

As it is as if the bits $\{c_k\}$ were differentially encoded, to obtain the bit error probability for decisions on the symbols $\{a_k\}$, we use (6.173):

$$P_{bit} = 4P_{bit,Ch} - 8P_{bit,Ch}^2 + 8P_{bit,Ch}^3 - 4P_{bit,Ch}^4 \tag{18.113}$$

In Figure 18.43 we show error probability curves for various receiver types. From the graph we note that to obtain an error probability of 10^{-3}, going from a coherent system to a non-coherent one, it is necessary to increase the value of $\Gamma = E_s/N_0$ as indicated in Table 18.3.

MSK with differential precoding

In the previously considered coherent scheme the transmitted symbols $\{a_k\}$ are obtained from NRZ mapping of bits $\{b_k\}$. If we now use a differential (pre)coding, where $\tilde{a}_k \in \{0, 1\}$,

$\tilde{a}_k = b_k \oplus b_{k-1}$ and $a_k = 1 - 2\tilde{a}_k = 1 - 2(b_k \oplus b_{k-1})$, then from (18.100) with $\tilde{c}_k \in \{0, 1\}$ and $c_k = 1 - 2\tilde{c}_k$, we get

$$\begin{aligned} \tilde{c}_k &= \tilde{c}_{k-1} \oplus \tilde{a}_k \\ &= \tilde{a}_0 \oplus \tilde{a}_1 \oplus \cdots \oplus \tilde{a}_k \\ &= b_{-1} \oplus b_k \end{aligned} \tag{18.114}$$

In other words, the symbol c_k is directly related to the information bit b_k; the performance loss due to (18.113) is thus avoided and we obtain

$$P_{bit} = P_{bit,Ch} \tag{18.115}$$

We emphasize that this differential (pre)coding scheme should be avoided if differential non-coherent receivers are employed, because one error in $\{\hat{a}_k\}$ generates a long error sequence in $\{\hat{b}_k = \widehat{\tilde{a}}_k \oplus \hat{b}_{k-1}\}$.

18.5.5 Remarks on spectral containment

From the analogy with OQPSK, the power spectrum of the complex envelope $s^{(bb)}$ of an MSK signal is directly obtained from (18.105),

$$\frac{1}{A^2}\bar{\mathcal{P}}^{(c)}_{s^{(bb)}}(f) = \frac{8T_b(1 + \cos 4\pi f T_b)}{\pi^2(1 - 16f^2T_b^2)^2} \tag{18.116}$$

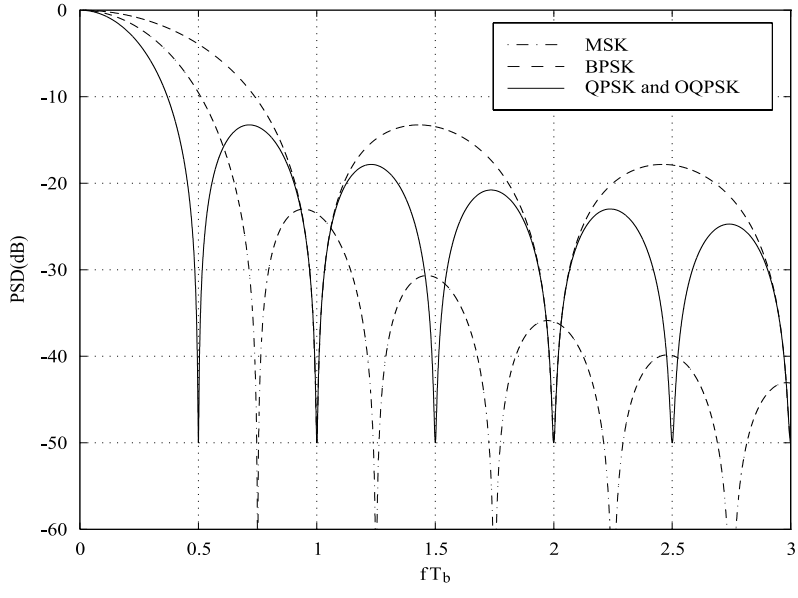


Figure 18.44. Normalized power spectral density of the complex envelope of signals obtained by four modulation schemes.

Modulating both BPSK and QPSK signals with h_{Tx} given by a retangular pulse with duration equal to the symbol period, a comparison between the various power spectra is illustrated in Figure 18.44. We note that for limited bandwidth channels, it is convenient to choose h_{Tx} of the raised cosine or square root raised cosine type. However, in radio applications the choice of a rectangular pulse may be appropriate, as it generates a signal with a lower peak/average power ratio and therefore is more suitable to be amplified with a power amplifier that operates near saturation.

Two observations on Figure 18.44 follow.

- For the same T_b the main lobe of QPSK extends up to $1/T = 0.5/T_b$, whereas that of MSK extends up to $1/T = 0.75/T_b$; thus the lobe of MSK is 50% wider than that of QPSK, consequently requiring a larger bandwidth.
- At high frequencies the spectrum of MSK decays as $1/f^4$, whereas the spectrum of QPSK decays as $1/f^2$.

18.6 Gaussian MSK (GMSK)

18.6.1 GMSK via CPFSK

GMSK is a variation of MSK in which, to reduce the bandwidth of the modulated signal s, the PAM signal x_f is filtered by a Gaussian filter.

Consider the scheme illustrated in Figure 18.45, in which we have the following filters:

- interpolator filter

$$g_I(t) = \frac{1}{2T} \mathrm{w}_T(t) \tag{18.117}$$

- shaping filter

$$g_G(t) = \frac{K}{\sqrt{2\pi}} e^{-K^2 t^2/2} \quad \text{with} \quad K = \frac{2\pi B_t}{\sqrt{\ln(2)}} \quad (B_t \text{ is the 3 dB bandwidth}) \tag{18.118}$$

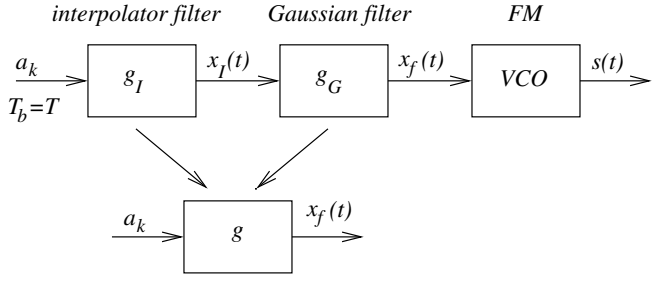


Figure 18.45. GMSK modulator.

- overall filter

$$g(t) = g_I * g_G(t) \tag{18.119}$$

Considering the signals we have:

- transmitted binary symbols

$$a_k \in \{-1, 1\} \tag{18.120}$$

- interpolated signal

$$x_I(t) = \sum_{k=-\infty}^{+\infty} a_k g_I(t - kT) = \sum_{k=-\infty}^{+\infty} a_k \frac{1}{2T} \mathrm{w}_T(t - kT) \tag{18.121}$$

- PAM signal

$$x_f(t) = \sum_{k=-\infty}^{+\infty} a_k g(t - kT) \tag{18.122}$$

- modulated signal

$$s(t) = A\cos\left(2\pi f_0 t + 2\pi h \int_{-\infty}^{t} x_f(\tau)\, d\tau\right) = A\cos(2\pi f_0 t + \Delta\varphi(t)) \tag{18.123}$$

where h is the modulation index, nominally equal to 0.5, and $A = \sqrt{2E_s/T}$.

From the above expressions it is clear that the GMSK signal is a frequency modulated signal with phase deviation $\Delta\varphi(t) = \pi \int_{-\infty}^{t} x_f(\tau)\, d\tau$.

An important parameter is the *3 dB bandwidth*, B_t, of the Gaussian filter. However, a reduction in B_t, useful in making prefiltering more selective, corresponds to a broadening of the PAM pulse with a consequent increase in the intersymbol interference, as can be noted in the plots of Figure 18.46. Thus a trade-off between the two requirements is necessary.

The product $B_t T$ was chosen equal to 0.3 in the GSM and HIPERLAN standards, and equal to 0.5 in the DECT standard (see Appendix 17.A). The case $B_t T = \infty$, i.e. without the Gaussian filter, corresponds to MSK.

Analyzing g in the frequency domain we have

$$g(t) = g_I * g_G(t) \overset{\mathcal{F}}{\longleftrightarrow} G(f) = G_I(f) \cdot G_G(f) \tag{18.124}$$

As

$$\begin{aligned} G_I(f) &= \tfrac{1}{2}\,\mathrm{sinc}(fT)e^{-j\pi fT} \\ G_G(f) &= e^{-2\pi^2(f/K)^2} \end{aligned} \tag{18.125}$$

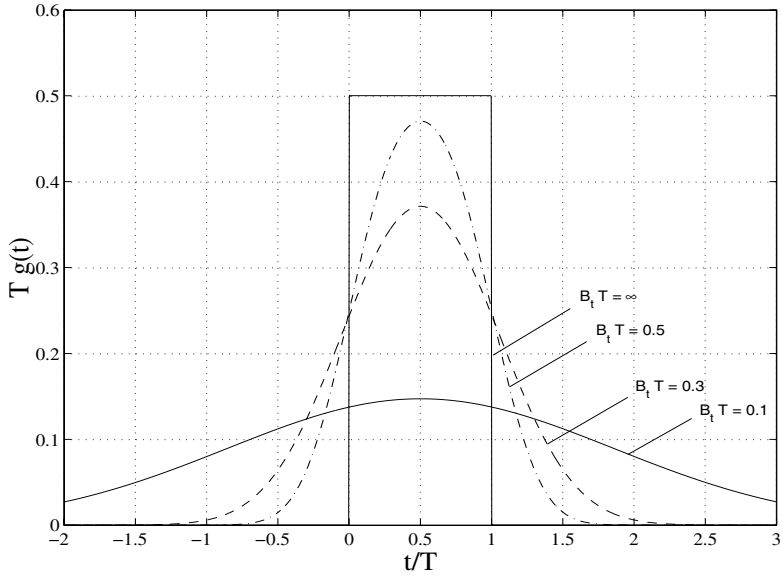


Figure 18.46. Overall pulse $g(t) = g_I * g_G(t)$, with amplitude normalized to $1/T$, for various values of the product B_tT.

it follows that

$$G(f) = \tfrac{1}{2}\,\mathrm{sinc}(fT)e^{-2\pi^2(f/K)^2}e^{-j\pi fT} \tag{18.126}$$

In Figure 18.47, the behavior of the phase deviation of a GMSK signal with $B_tT = 0.3$ is compared with the phase deviation of an MSK signal; note that in both cases the phase is continuous, but for GMSK we get a smoother curve, without discontinuities in the slope.

Possible trajectories of the phase deviation for $B_tT = 0.3$ and $B_tT = \infty$ are illustrated in Figure 18.48.

Values of $e^{j\Delta\varphi(t)}$ at the decision instants $T + kT$ are illustrated in Figure 18.49 for a GMSK signal with $B_tT = 0.3$.

18.6.2 Power spectrum of GMSK

GMSK is a particular case of phase modulation which performs a non-linear transformation on the message, thus it is very difficult to evaluate analytically the power spectrum of a GMSK signal. Hence we resort to a discrete time spectral estimate, for example, the Welch method (see Section 1.11). The estimate is made with reference to the baseband equivalent of a GMSK signal, that is using the complex envelope of the modulated signal given by $s^{(bb)}(t) = Ae^{j\Delta\varphi(t)}$.

The result of the spectral estimate is illustrated in Figure 18.50. Note that the central lobe has an extension up to $0.75/T$; therefore the sampling frequency, F_Q, to simulate the baseband equivalent scheme of Figure 18.45 may be chosen equal to $F_Q = 4/T$ or $F_Q = 8/T$.

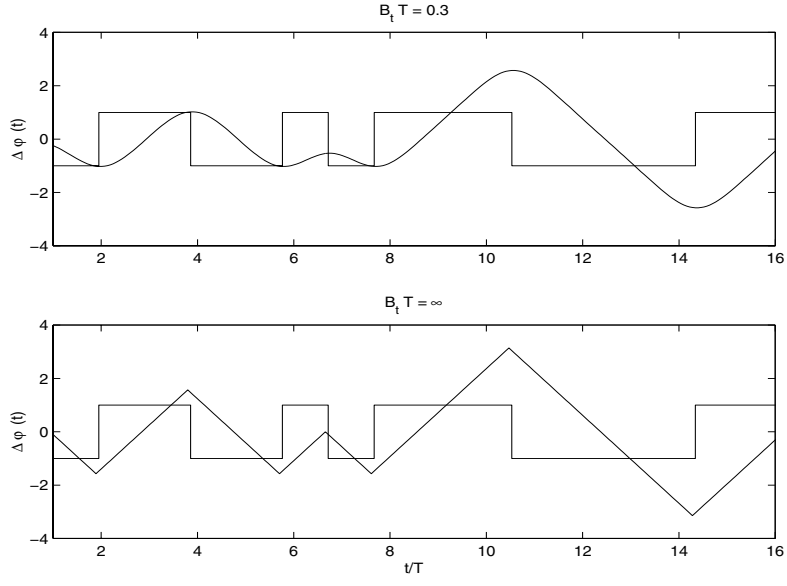


Figure 18.47. Phase deviation $\Delta\varphi$ of a GMSK signal for $B_tT = 0.3$, compared with the phase deviation of an MSK signal.

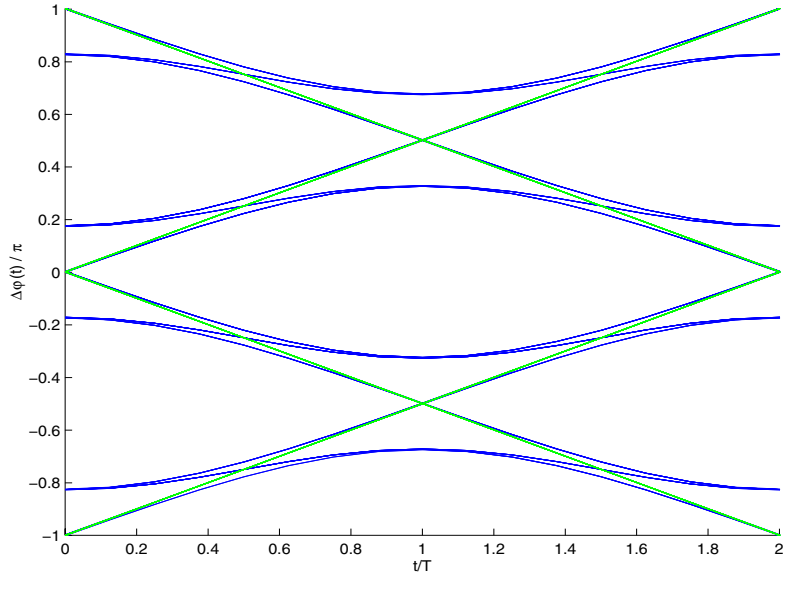


Figure 18.48. Trajectories of the phase deviation of a GMSK signal for $B_tT = 0.3$ (solid line) and $B_tT = \infty$ (dotted line).

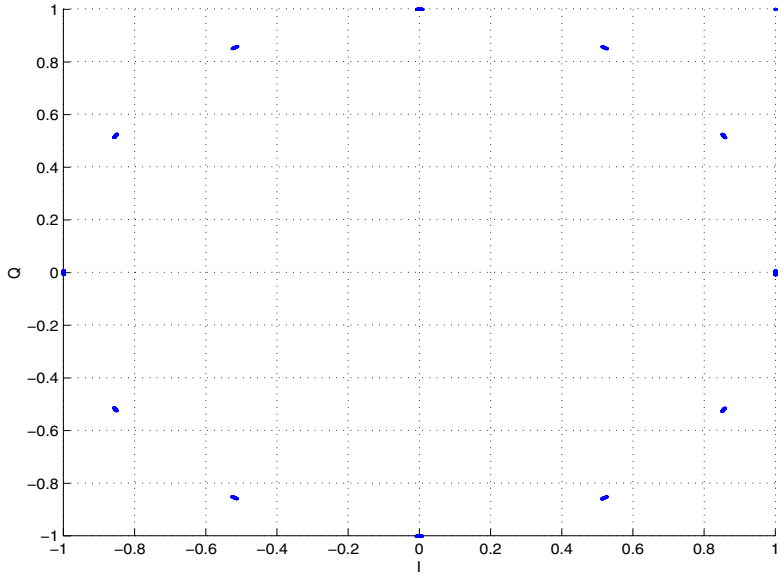


Figure 18.49. GMSK constellation for $B_tT = 0.3$.

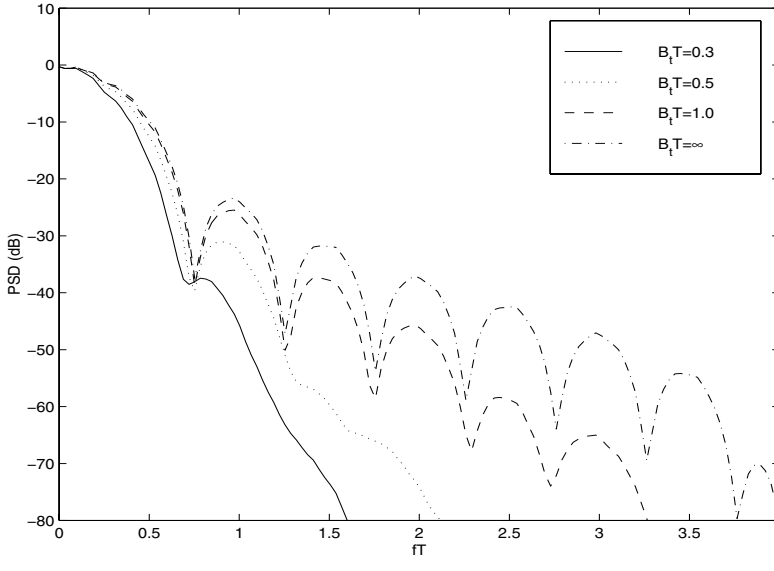


Figure 18.50. Estimate of the power spectral density of a GMSK signal for various values of B_tT.

18.6.3 Implementation of a GMSK scheme

For the scheme of Figure 18.45 three possible configurations are given, depending on the position of the DAC.

Configuration I

The first configuration, illustrated in Figure 18.51, is in the analog domain; in particular, an analog low pass Gaussian filter is employed. As shown in [1], it is possible to implement a good approximation of the Gaussian filter by resorting to simple devices, such as lattice LC filters. However, the weak point of this scheme is represented by the VCO, because in an open loop the voltage/frequency VCO characteristic is non-linear and as a consequence the modulation index can vary even by a factor 10 in the considered frequency range.

Configuration II

The second configuration is represented in Figure 18.52. The digital filter $g(nT_Q)$ that approximates the analog filter $g(t)$ is designed by the window method [4, pag. 444]. For an oversampling factor of $Q_0 = 8$, letting $T_Q = T/Q_0$, we consider four filters that are obtained by windowing the pulse $g(t)$ to intervals $(0, T)$, $(-T/2, 3T/2)$, $(-T, 2T)$, and $(-3T/2, 5T/2)$, respectively; the coefficients of the last filter are listed in Table 18.4, using the fact that $g(nT_Q)$ has even symmetry with respect to the peak at $4T_Q = T/2$.

A comparison among the frequency responses of the four discrete-time filters and the continuous-time filter is illustrated in Figure 18.53. For a good approximation to the analog filter, the possible choices are limited to the two FIR filters with 23 and 31 coefficients, with support $(-T, 2T)$ and $(-3T/2, 5T/2)$, respectively. From now on we will refer to the filter with 31 coefficients.

We note from Figure 18.46 that, for $B_t T \geq 0.3$, most of the pulse energy is contained within the interval $(-T, 2T)$; therefore the effect of interference does not extend over more than three symbol periods. With reference to Figure 18.52, the filter g is an interpolator filter

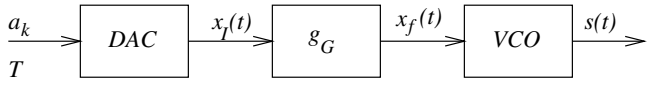


Figure 18.51. GMSK modulator: configuration I.

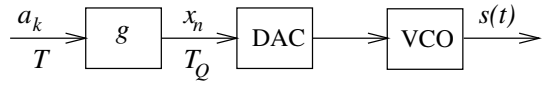


Figure 18.52. GMSK modulator: configuration II.

Table 18.4 Digital filter coefficients obtained by windowing $g(t)$; $T_Q = T/8$.

$g(nT_Q)$	*value*
$g(4T_Q)$	0.37119
$g(5T_Q)$	0.36177
$g(6T_Q)$	0.33478
$g(7T_Q)$	0.29381
$g(8T_Q)$	0.24411
$g(9T_Q)$	0.19158
$g(10T_Q)$	0.14168
$g(11T_Q)$	0.09850
$g(12T_Q)$	0.06423
$g(13T_Q)$	0.03921
$g(14T_Q)$	0.02236
$g(15T_Q)$	0.01189
$g(16T_Q)$	0.00589
$g(17T_Q)$	0.00271
$g(18T_Q)$	0.00116
$g(19T_Q)$	0.00046

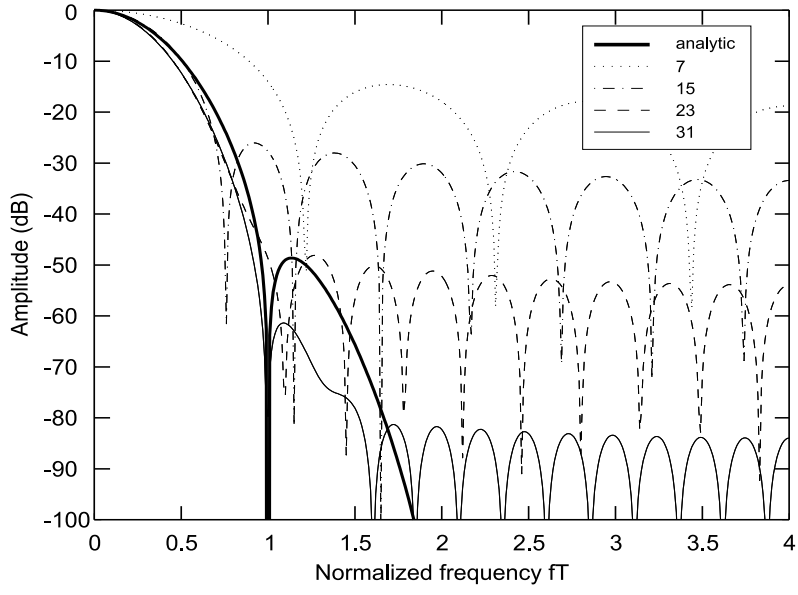


Figure 18.53. Frequency responses of $g(t)$ and $g(nT_Q)$, for $T_Q = T/8$, and various lengths of the FIR filters.

from T to T/Q_0, that can be efficiently implemented by using the polyphase representation of the impulse response (see Appendix 1.A). Then, recognizing that $x_{kQ_0+\ell} = f^{(\ell)}(a_{k+2}, a_{k+1}, a_k, a_{k-1})$, the input–output relation can be memorized in a RAM or look-up table, and let the input vector $[a_{k+2}, a_{k+1}, a_k, a_{k-1}]$ address one of the possible $2^4 = 16$ output values; this must be repeated for every phase $\ell \in \{0, 1, \ldots, Q_0 - 1\}$. Therefore, there are Q_0 RAMs, each with 16 memory locations.

Configuration III

The weak point of the previous scheme is again represented by the analog VCO; thus it is convenient to partially implement in the digital domain also the frequency modulation stage.

Real-valued scheme. The real-valued scheme is illustrated in Figure 18.54. The samples $\{u_n\}$ are given by

$$u_n = u(nT_Q) = A\cos(2\pi f_1 nT_Q + \Delta\varphi(nT_Q)) \tag{18.127}$$

where f_1 is an intermediate frequency smaller than the sampling rate,

$$f_1 = \frac{N_1}{N_2}\frac{1}{T_Q} \tag{18.128}$$

where N_1 and N_2 are relatively prime numbers, with $N_1 < N_2$. Thus $f_1 T_Q = N_1/N_2$, and

$$u_n = A\cos\left(2\pi\frac{N_1}{N_2}n + \Delta\varphi_n\right) = A\cos[\varphi_n] \tag{18.129}$$

where

$$\Delta\varphi_n = \Delta\varphi(nT_Q) = \pi\int_{-\infty}^{nT_Q} x(\tau)\,d\tau \simeq \pi\sum_{i=-\infty}^{n} T_Q x(iT_Q) \tag{18.130}$$

More simply, let $x_n = x_f(nT_Q)$ and $X_n = X_{n-1} + x_n$; then it follows that $\Delta\varphi_n = \pi T_Q X_n$. Therefore in (18.129) φ_n, with carrier frequency f_1, becomes

$$\varphi_n^{(f_1)} = 2\pi\frac{N_1}{N_2}n + \pi T_Q X_n = \varphi_{n-1}^{(f_1)} + 2\pi\frac{N_1}{N_2} + \pi T_Q x_n \tag{18.131}$$

that is the value $\varphi_n^{(f_1)}$ is obtained by suitably scaling the accumulated values of x_n. To obtain u_n, we map the value of $\varphi_n^{(f_1)}$ into the memory address of a RAM which contains values of the cosine function (see Figure 18.55).

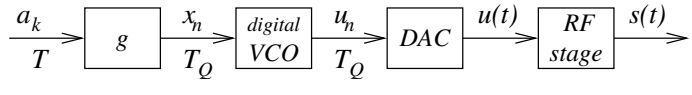


Figure 18.54. GMSK modulator: configuration III.

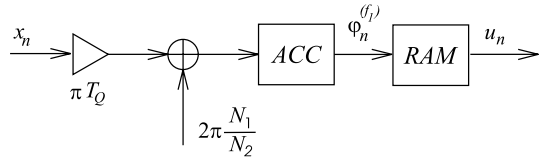


Figure 18.55. Digital implementation of the VCO.

Obviously the size of the RAM depends on the accuracy with which u_n and φ_n are quantized.[3] We note that $u(nT_Q)$ is a real-valued passband signal, with spectrum centered around the frequency f_1; the choice of f_1 is constrained by the bandwidth of the signal $u(nT_Q)$, equal to about $1.5/T$, and also by the sampling period, chosen in this example equal to $T/8$; then it must be

$$-\frac{3}{4}\frac{1}{T} + f_1 > 0 \qquad \frac{3}{4}\frac{1}{T} + f_1 < \frac{4}{T} \tag{18.132}$$

or $3/(4T) < f_1 < 13/(4T)$. A possible choice is $f_1 = 1/(4T_Q)$ assuming $N_1 = 1$ and $N_2 = 4$. With this choice we have a *image spacing/signal bandwidth* ratio equal to 4/3. Moreover, $\cos(\varphi_n) = \cos(2(\pi/4)n + \Delta\varphi_n)$ becomes $\cos((\pi/2)n + \Delta\varphi_n)$, which in turn is equal to $\pm\cos(\Delta\varphi_n)$ for n even and $\pm\sin(\Delta\varphi_n)$ for n odd. Therefore the scheme of Figure 18.54 can be further simplified.

Complex-valued scheme. Instead of digitally shifting the signal to an intermediate frequency, it is possible to process it at baseband, thus simplifying the implementation of the DAC equalizer filter.

Consider the scheme of Figure 18.56, where at the output of the exponential block we have the sampled signal $s^{(bb)}(nT_Q) = Ae^{j\Delta\varphi(nT_Q)} = s_{\mathcal{I}}^{(bb)}(nT_Q) + js_{\mathcal{Q}}^{(bb)}(nT_Q)$, where $s_{\mathcal{I}}^{(bb)}$

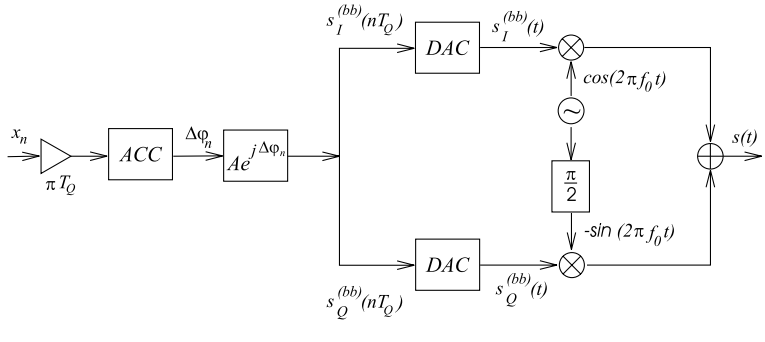


Figure 18.56. GMSK modulator with a complex-valued digital VCO.

[3] To avoid quantization effects, the number of bits used to represent the accumulated values is usually much larger than the number of bits used to represent φ_n. In practice φ_n coincides with the most significant bits of the accumulated values.

and $s_{\mathcal{Q}}^{(bb)}$ are the in phase and quadrature components. Then we obtain

$$\begin{aligned} s_{\mathcal{I}}^{(bb)}(nT_Q) &= A\cos(\Delta\varphi_n) \\ s_{\mathcal{Q}}^{(bb)}(nT_Q) &= A\sin(\Delta\varphi_n) = A\cos\left(\frac{\pi}{2} - \Delta\varphi_n\right) \end{aligned} \tag{18.133}$$

Once the two components have been interpolated by the two DACs, the signal $s(t)$ can be reconstructed as

$$s(t) = s_{\mathcal{I}}^{(bb)}(t)\cos(2\pi f_0 t) - s_{\mathcal{Q}}^{(bb)}(t)\sin(2\pi f_0 t) \tag{18.134}$$

With respect to the real-valued scheme, we still have a RAM which stores the values of the cosine function, but two DACs are now required.

18.6.4 Linear approximation of a GMSK signal

According to Laurent [5], if $B_t T \geq 0.3$, a GMSK signal can be approximated by a QAM signal given by

$$\begin{aligned} s^{(bb)}(t) &= \sum_{k=-\infty}^{+\infty} e^{j\frac{\pi}{2}\Sigma_{i=-\infty}^{k} a_i} h_{Tx}(t - kT_b) \\ &= \sum_{k=-\infty}^{+\infty} j^{k+1} c_k\, h_{Tx}(t - kT_b) \qquad c_k = c_{k-1}a_k \end{aligned} \tag{18.135}$$

where $h_{Tx}(t)$ is a suitable real-valued pulse that depends on the parameter $B_t T$ and has a support equal to $(L+1)T_b$, if LT_b is the support of $g(t)$. For example, we show in Figure 18.57 the plot of h_{Tx} for a GMSK signal with $B_t T = 0.3$.

The linearization of $s^{(bb)}$, that leads to interpreting GMSK as a QAM extension of MSK with a different transmit pulse, is very useful for the design of the optimum receiver, which is the same as for QAM systems. Figure 18.58 illustrates the linear approximation of the GMSK model.

As for MSK, also for GMSK it is useful to differentially (pre)code the data $\{a_k\}$ if a coherent demodulator is employed.

Performance of GMSK demodulators

We consider the performance of a GMSK system for an ideal AWGN channel, and compare it with the that of $\pi/4$-DQPSK.

Coherent demodulator. Assuming a coherent receiver, as illustrated in Figure 18.42, performance of the optimum receiver according to the MAP criterion, evaluated on the basis of the minimum distance of the received signals, is approximated by the following relation [6]

$$P_{bit,Ch} = Q(\sqrt{c\Gamma}) \tag{18.136}$$

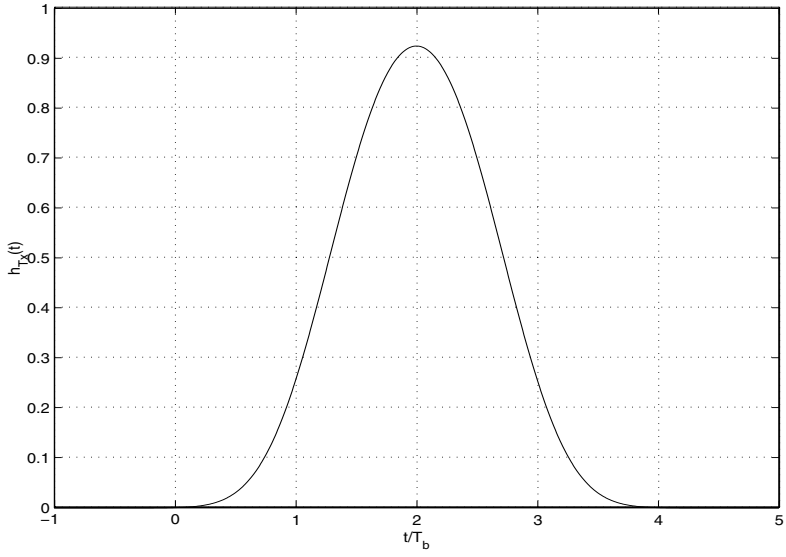


Figure 18.57. Pulse h_{Tx} for a GMSK signal with $B_t T = 0.3$.

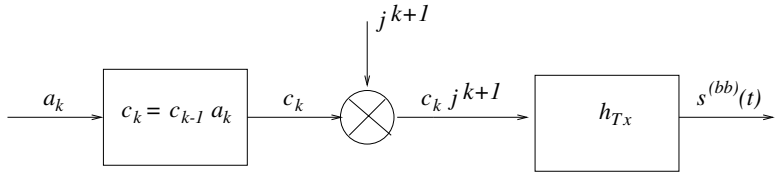


Figure 18.58. Linear approximation of a GMSK signal. h_{Tx} is a suitable pulse which depends on the value of $B_t T$.

Table 18.5 Values of coefficient c as a function of the modulation system.

Modulation system	c
MSK	2.0
GMSK, $B_t T = 0.5$	1.93
GMSK, $B_t T = 0.3$	1.78
$\pi/4$-DQPSK, $\forall \rho$	1.0

where the coefficient c assumes the values given in Table 18.5 for four modulation systems. The plots of $P_{bit,Ch}$ for the various cases are illustrated in Figure 18.59.

As usual, if the data $\{a_k\}$ are differentially (pre)coded, $P_{bit} = P_{bit,Ch}$ holds, otherwise the relation (18.113) holds. From now on we assume that a differential precoder is employed in the presence of coherent demodulation.

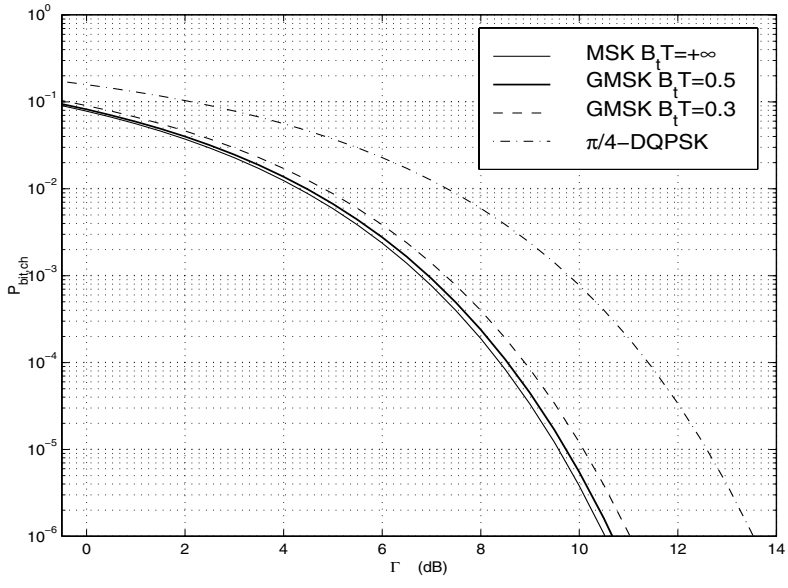


Figure 18.59. $P_{bit,Ch}$ as a function of Γ for the four modulation systems of Table 18.5.

For the case $B_tT = 0.3$, and for an ideal AWGN channel, in Figure 18.60 we also give the performance obtained for a receive filter g_{Rc} of the Gaussian type [7, 8], whose impulse response is given in (18.118), where the 3 dB bandwidth is now denoted by B_r. Clearly the optimum value of B_rT depends on the modulator type and in particular on B_tT. System performance is evaluated using a 4-state VA or a threshold detector (TD). The VA uses an estimated overall system impulse response obtained by the linear approximation of GMSK. The Gaussian receive filter is characterized by $B_rT = 0.3$, chosen for best performance. We observe that the VA gains a fraction of dB as compared to the TD; furthermore, the performance is slightly better than the approximation (18.136).

Non-coherent demodulator (1BDD). For a non-coherent receiver, as illustrated in Figure 18.41, without including the receive filter g_A, we illustrate in Figure 18.61 the eye diagram at the decision point of a GMSK system, for three values of B_tT.

We note that, for decreasing values of B_tT, the system exhibits an increasing level of ISI, and the eye tends to shut; including the receive filter g_A with finite bandwidth this phenomenon is further emphasized.

Simulation results obtained by considering a receive Gaussian filter g_A, and a Gaussian baseband equivalent system, whose bandwidth is optimized for each different modulation, are shown in Figure 18.62.

The sensitivity of the 1BDD to the parameter B_tT of GMSK is higher as compared to a coherent demodulator. With decreasing B_tT, we observe a considerable worsening of performance; in fact, to achieve a $P_{bit} = 10^{-3}$ the GMSK with $B_tT = 0.5$ requires a signal-to-noise ratio Γ that is 2.5 dB higher with respect to MSK, whereas GMSK with

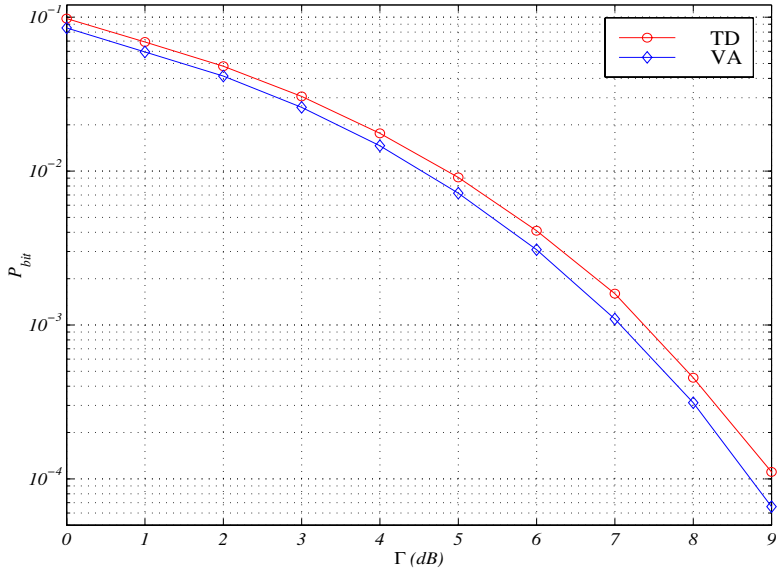


Figure 18.60. P_{bit} as a function of Γ for a coherently demodulated GMSK ($B_tT = 0.3$), for an ideal channel and Gaussian receive filter with $B_rT = 0.3$. Two detectors are compared: 1) four-state Viterbi algorithm and 2) threshold detector.

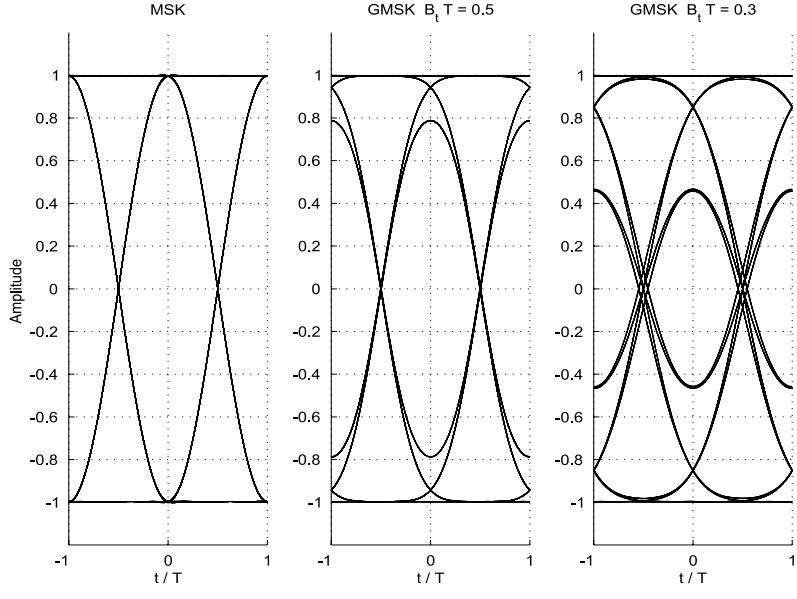


Figure 18.61. Eye diagram at the decision point of the 1BDD for a GMSK system, for an ideal channel and without the filter g_A: (a) $B_tT = +\infty$, (b) $B_tT = 0.5$, (c) $B_tT = 0.3$.

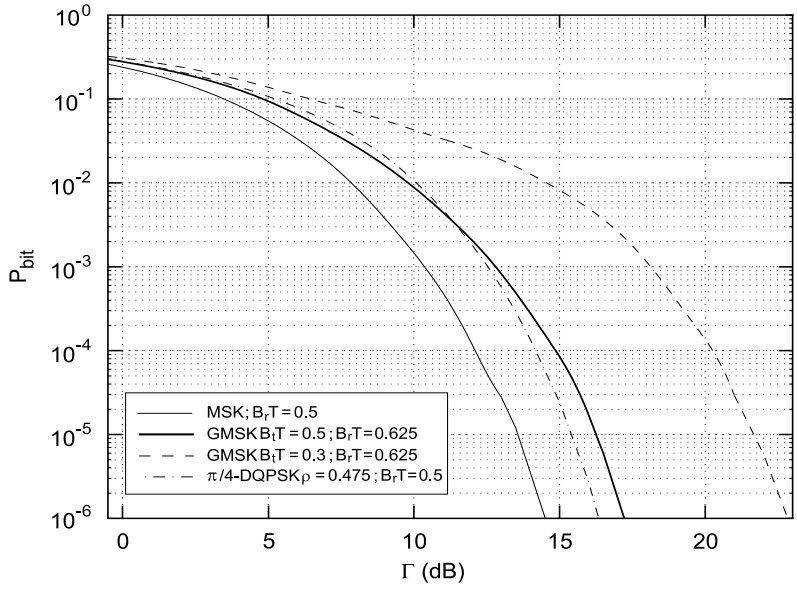


Figure 18.62. P_{bit} as a function of Γ obtained with the 1BDD for GMSK, for an ideal channel and Gaussian receive filter having a normalized bandwidth B_rT.

$B_tT = 0.3$ requires an increment of 7.8 dB. In Figure 18.62 we also show for comparison purposes the performance of $\pi/4$-DQPSK with a receive Gaussian filter.

The performance of another widely used demodulator, LDI (see Section 18.2.3), is quite similar to that of 1BDD, showing a substantial equivalence between the two non-coherent demodulation techniques applied to GMSK and $\pi/4$-DQPSK.

Comparison. Always for an ideal AWGN channel, a comparison among the various modulators and demodulators is given in Table 18.6. As a first observation, we note that a coherent receiver with an optimized Gaussian receive filter provides the same performance,

Table 18.6 Required values of Γ, in dB, to achieve a $P_{bit} = 10^{-3}$ for various modulation and demodulation schemes.

Modulation	Demodulation		
	coherent (MAP)	coherent (g_A gauss. + TD)	differential (g_A gauss. + 1BDD)
$\pi/4$-DQPSK o QPSK ($\rho = 0.3$)	9.8	9.8 ($\rho = 0.3$)	12.5 ($B_rT = 0.5$)
MSK	6.8	6.8 ($B_rT = 0.25$)	10.3 ($B_rT = 0.5$)
GMSK ($B_tT = 0.5$)	6.9	6.9 ($B_rT = 0.3$)	12.8 ($B_rT = 0.625$)
GMSK ($B_tT = 0.3$)	7.3	7.3 ($B_rT = 0.3$)	18.1 ($B_rT = 0.625$)

evaluated by the approximate relation (18.136), of the MAP criterion. Furthermore, we note that for GMSK with $B_tT \leq 0.5$, because of strong ISI, a differential receiver undergoes a substantial penalty in terms of Γ to achieve a given P_{bit} with respect to a coherent receiver; this effect is mitigated by canceling ISI by suitable equalizers [9].

Another method is to detect the signal in the presence of ISI, in part due to the channel and in part to the differential receiver, by the Viterbi algorithm. Substantial improvements with respect to the simple threshold detector are obtained, as shown in [10].

In the previous comparison between amplitude modulation ($\pi/4$-DQPSK) and phase modulation (GMSK) schemes we did not take into account the non-linearity introduced by the power amplifier (see Section 4.8), which leads to: 1) signal distortion and 2) spectral spreading that creates interference in adjacent channels. Usually, the latter effect is dominant and is controlled using a HPA with a back-off that can be even of several dB. In some cases signal predistortion before the HPA allows a decrease of the OBO.

Overall, the best system is the one that achieves, for the same P_{bit}, the smaller value of

$$(\Gamma)_{dB} + (\text{OBO})_{dB} \tag{18.137}$$

In other words (18.137), for the same P_{bit}, additive channel noise and transmit HPA, selects the system for which the transmitted signal has the lowest power. Obviously in (18.137) Γ depends on the OBO; at high frequencies, where the HPA usually introduces large levels of distortion, the OBO for a linear modulation scheme may be so large that a phase modulation scheme may be the best solution in terms of (18.137).

Performance of a GSM receiver in the presence of multipath

We conclude this section giving the performance in terms of P_{bit} cdf (see Appendix 7.E) of a GMSK scheme with $B_tT = 0.3$ for transmission over channels in the presence of frequency selective Rayleigh fading (see Chapter 4). The receive filter is Gaussian with $B_rT = 0.3$, implemented as a FIR filter with 24 $T/8$-spaced coefficients; the detector is a 32-state VA or a DFE with 13 coefficients of the T-spaced FF filter and 7 coefficients of the FB filter. For the DFE, a substantial performance improvement is observed by placing before the FF filter a matched filter (MF) that, to save computational complexity, operates at T; the improvement is mainly due to a better acquisition of the timing phase.

Table 18.7 Power delay profiles for the analyzed channels.

Coefficient	Relative delay	Power delay profile			
		EQ	HT	UA	RA
0	0	1/6	0.02	0.00566	0.00086
1	T	1/6	0.75	0.01725	0.99762
2	$2T$	1/6	0.08	0.80256	0.00109
3	$3T$	1/6	0.02	0.15264	0.00026
4	$4T$	1/6	0.01	0.01668	0.00011
5	$5T$	1/6	0.12	0.00521	0.00006

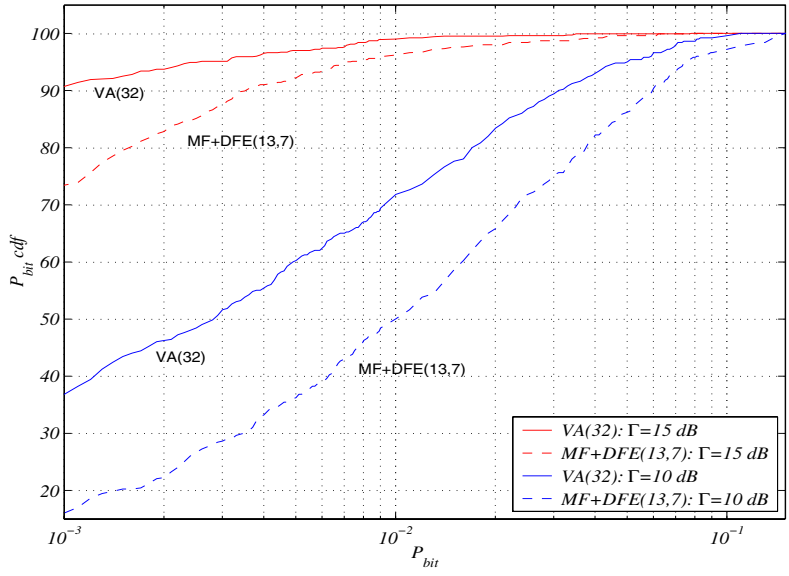


Figure 18.63. Comparison between Viterbi algorithm and DFE preceded by a MF for the multipath channel EQ6, in terms of BER cdf.

The channel model is also obtained by considering a T-spaced impulse response; in particular, the performance is evaluated for the four models given in Table 18.7: *equal gain* (EQ), *hilly terrain* (HT), *urban area* (UA), *rural area* (RA). The difference in performance between the two receivers is higher for the EQ channel; this is the case shown in Figure 18.63 for two values of the signal-to-noise ratio Γ at the receiver.

Bibliography

[1] B. Razavi, *RF microelectronics*. Englewood Cliffs, NJ: Prentice-Hall, 1997.

[2] K. Feher, *Wireless digital communications*. Upper Saddle River, NJ: Prentice-Hall, 1995.

[3] S. Benedetto and E. Biglieri, *Principles of digital transmission with wireless applications*. New York: Kluwer Academic Publishers, 1999.

[4] A. V. Oppenheim and R. W. Schafer, *Discrete-time signal processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[5] P. Laurent, "Exact and approximate construction of digital phase modulations by superposition of amplitude modulated pulses (AMP)", *IEEE Trans. on Communications*, vol. 34, pp. 150–160, Feb. 1986.

[6] S. Ohmori, H. Wakana, and S. Kawase, *Digital communications technologies*. Boston, MA: Artech House, 1998.

[7] K. Murota and K. H. Hirade, "GMSK modulation for digital mobile radio telephony", *IEEE Trans. on Communications*, vol. 29, p. 1045, July 1991.

[8] M. K. Simon and C. C. Wang, "Differential detection of Gaussian MSK in a mobile radio environment", *IEEE Trans. on Vehicular Technology*, vol. 33, pp. 311–312, Nov. 1984.

[9] N. Benvenuto, P. Bisaglia, A. Salloum, and L. Tomba, "Worst case equalizer for noncoherent HIPERLAN receivers", *IEEE Trans. on Communications*, vol. 48, pp. 28–36, Jan. 2000.

[10] N. Benvenuto, P. Bisaglia, and A. E. Jones, "Complex noncoherent receivers for GMSK signals", *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1876–1885, Nov. 1999.

Appendix 18.A *Continuous phase modulation (CPM)*

A signal with constant envelope can be defined by its passband version as

$$s(t) = \sqrt{\frac{2E_s}{T}} \cos(2\pi f_0 t + \Delta\varphi(t, \mathbf{a})) \tag{18.138}$$

where E_s is the energy per symbol, T the symbol period, f_0 the carrier frequency and $\mathbf{a}$ denotes the symbol message $\{a_k\}$ at the modulator input. For a continuous phase modulation scheme, the phase deviation $\Delta\varphi(t, \mathbf{a})$ can be expressed as

$$\Delta\varphi(t, \mathbf{a}) = 2\pi h \int_{-\infty}^{t} x_f(\tau)\, d\tau \tag{18.139}$$

with

$$x_f(t) = \sum_{k=-\infty}^{+\infty} a_k g(t - kT) \tag{18.140}$$

where $g(t)$ is called *instantaneous frequency pulse*. In general, the pulse $g(t)$ satisfies the following properties:

limited duration $g(t) = 0$ for $t < 0$ and $t > LT$ (18.141)

symmetry $g(t) = g(LT - t)$ (18.142)

normalization $\int_0^{LT} g(\tau)\, d\tau = \frac{1}{2}$ (18.143)

Alternative definition of CPM

From (18.140) and (18.139) we can redefine the phase deviation as follows

$$\Delta\varphi(t, \mathbf{a}) = 2\pi h \int_{-\infty}^{t} \sum_{i=-\infty}^{k} a_i g(\tau - iT)\, d\tau = 2\pi h \sum_{i=-\infty}^{k} a_i \int_{-\infty}^{t-iT} g(\tau)\, d\tau \qquad kT \le t < (k+1)T \tag{18.144}$$

or

$$\Delta\varphi(t, \mathbf{a}) = 2\pi h \sum_{i=-\infty}^{k} q(t - iT) \qquad kT \le t < (k+1)T \tag{18.145}$$

with

$$q(t) = \int_{-\infty}^{t} g(\tau)\, d\tau \tag{18.146}$$

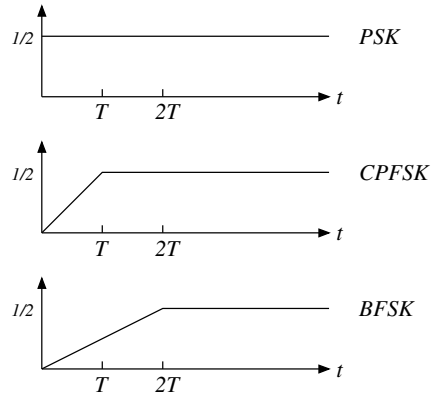


Figure 18.64. Three examples of phase response pulses for CPM.

The pulse $q(t)$ is called *phase response pulse* and represents the most important part of the CPM signal because it indicates to what extent each information symbol contributes to the overall phase deviation. In Figure 18.64 the phase response pulse $q(t)$ is plotted for PSK, CPFSK, and BFSK. In general the maximum value of the slope of the pulse $q(t)$ is related to the width of the main lobe of the PSD of the modulated signal $s(t)$, and the number of continuous derivatives of $q(t)$ influences the shape of the secondary lobes.

In general, information symbols $\{a_k\}$ belong to an M-ary alphabet $\mathcal{A}$, that for M even is given by $\{\pm 1, \pm 3, \ldots, \pm(M-1)\}$. The constant h is called modulation index and determines, together with the dimension of the alphabet, the maximum phase variation in a symbol period, equal to $(M-1)h\pi$. By changing $q(t)$ (or $g(t)$), h, and M, we can generate several continuous phase modulation schemes. The modulation index h is always given by the ratio of two integers, $h = \ell/p$, because this implies that the phase deviation, evaluated modulo 2π, assumes values in a finite alphabet. In fact, we can write

$$\Delta\varphi(t, \mathbf{a}) = 2\pi h \sum_{i=k-L+1}^{k} a_i q(t - iT) + \psi_{k-L} \qquad kT \le t < (k+1)T \tag{18.147}$$

with

$$\psi_{k-L} = \left[\pi \frac{\ell}{p} \sum_{i=-\infty}^{k-L} a_i \right]_{mod\, 2\pi} \tag{18.148}$$

ψ_{k-L} is called *phase state*; it represents the overall contribution given by the symbols $\ldots, a_{-1}, a_0, a_1, \ldots, a_{k-L}$ to the phase duration in the interval $[kT, (k+1)T)$, and can only assume $2p$ distinct values. The first term in (18.147) is called *corrective state* and, because it depends on L symbols $a_{k-L+1}, \ldots, a_k$, at a certain instant t it can only assume M^{L-1} distinct values. The phase deviation is therefore characterized by a total number of values equal to $2pM^{L-1}$.

CPM schemes with $L = 1$, i.e. CPFSK, are called *full response schemes* and have a reduced complexity with a number of states equal to $2p$; schemes with $L \neq 1$ are instead

called *partial response schemes*. Because of the memory in the modulation process a partial response scheme allows a trade-off between the error probability at the receiver and the shaping of power spectra of the modulated signal. However, this advantage is obtained at the expense of a greater complexity of the receiver. For this reason the modulation index is usually a simple rational number as $1, 1/2, 1/4, 1/8$.

Advantages of CPM

The popularity of the continuous phase modulation technique derives first of all from the constant envelope property of the CPM signal; in fact, a signal with a constant envelope allows using very efficient power amplifiers. In the case of linear modulation techniques, as QAM or OFDM, it is necessary to compensate for the non-linearity of the amplifier by predistortion or to decrease the average power in order to work in linear conditions.

Before the introduction of TCM, it was believed that source and/or channel coding would allow an improvement in performance only at the expense of a loss in transmission efficiency, and hence would require a larger bandwidth; CPM permits both good performance and highly efficient transmission. However, one of the drawbacks of the CPM is the implementation complexity of the optimum receiver, especially in the presence of dispersive channels.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Chapter 19

Design of high speed transmission systems over unshielded twisted pair cables

In this chapter we describe the design of two high speed data transmission systems over unshielded twisted pair cables [1, 2].

19.1 Design of a quaternary partial response class-IV system for data transmission at 125 Mbit/s

Figure 19.1 shows the block diagram of a transceiver for a *quaternary partial response class-IV* (QPR-IV), or quaternary modified duobinary (see Appendix 7.A), system for data transmission at 125 Mbit/s over unshielded twisted-pair cables [1]. In the transmitter, information bits are first scrambled and then input to a 2B1Q differential encoder that yields output symbols belonging to the quaternary alphabet $\mathcal{A} = \{-3, -1, +1, +3\}$ (see Example 6.5.1 on page 479); differential encoding makes the transmission of information insensitive to the polarity of the received signals. Signal shaping into PR-IV form is accomplished by the cascade of the following elements: the D/A converter, the *analog transmit filter* (ATF), the cable, the *analog receive filter* (ARF) with *automatic gain control* (AGC), the A/D converter, and, in the digital domain, the fixed *decorrelation filter* (DCF) and the adaptive equalizer. After equalization, the sequence of transmitted quaternary symbols is detected by a Viterbi algorithm. As the Viterbi algorithm introduces a delay in the detection of the transmitted sequence, as an alternative it is possible to use a threshold detector that individually detects transmitted symbols with a negligible delay. If the *mean-square error* (MSE) is below a fixed value, threshold detection, or *symbol-by-symbol detection* (SSD), is selected as the corresponding symbol decisions are then sufficiently reliable. If, instead, the MSE exceeds the fixed value, the Viterbi algorithm is employed. Finally, differential 1Q2B decoding and descrambling are performed.

Analog filter design

Figure 19.2 depicts the overall analog channel considered for the design of ATF and ARF. The impulse response is denoted by $h(t, L, u_c)$, where L is the cable length and u_c is the control signal used for AGC. To determine ATF and ARF with low implementation

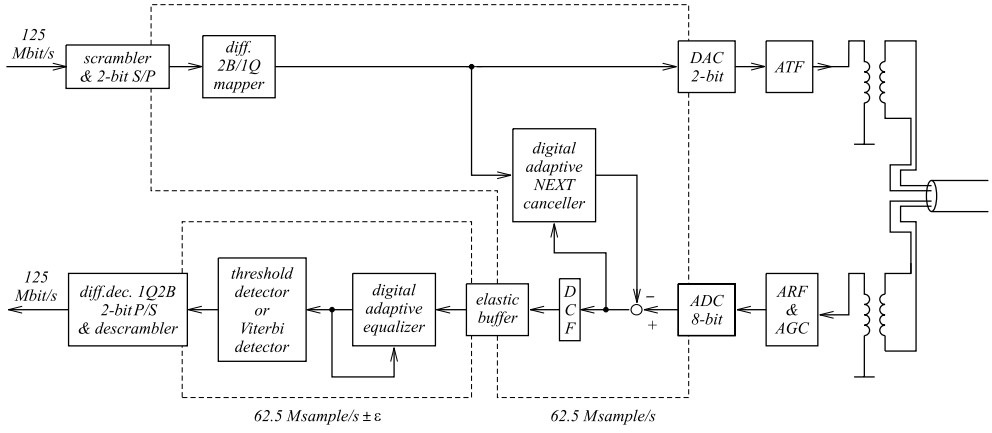


Figure 19.1. Block diagram of a QPR-IV transceiver. [*From [1], © 1995 IEEE*.]

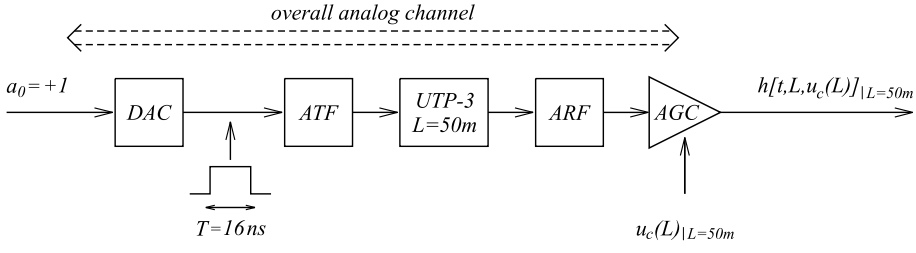


Figure 19.2. Overall analog channel considered for the joint optimization of the analog transmit (ATF) and receive (ARF) filters. [*From [1], © 1995 IEEE*.]

complexity, the transfer functions of these filters are first expressed in terms of poles and zeros; the pole-zero configurations are then jointly optimized by simulated annealing as described in [3]. The cost function that is used for the optimization reflects two criteria: a) the mean-square error between the impulse response $h(t, L, u_c)$ for a cable length equal to 50 m, and the ideal PR-IV response must be below a certain value and b) spectral components of the transmitted signal above 30 MHz should be well suppressed to achieve compliance with regulations on radiation limits. Investigations with various orders of the respective transfer functions have shown that a good approximation of an ideal PR-IV response is obtained with 5 poles and 3 zeros for the ATF, and 3 poles for the ARF.

Received signal and adaptive gain control

The received signal at the input of the ADC is expressed as

$$x(t) = \sum_{k=-\infty}^{+\infty} a_k\, h[t - kT, L, u_c(L)] + \sum_{k=-\infty}^{+\infty} a_k^N\, h_N[t - kT, L, u_c(L)] + w_R(t) \quad (19.1)$$

where $\{a_k\}$ and $\{a_k^N\}$ are the sequences of quaternary symbols generated by the local transmitter and remote transmitter, respectively, $h_N[t, L, u_c(L)]$ is the NEXT channel response and $w_R(t)$ is additive Gaussian noise. The signal $x(t)$ is sampled by the ADC that operates synchronously with the DAC at the modulation rate of $1/T = 62.5$ MBaud. The adjustment of the AGC circuit is computed digitally, using the sampled signal $x_k = x(kT)$, such that the ADC output signal achieves a constant average statistical power $\mathtt{M}_R$, i.e.,

$$E[x_k^2]\Big|_{u_c=u_c(L)} = \frac{1}{T}\int_0^T E[x^2(t)]\,dt\Big|_{u_c=u_c(L)} = \mathtt{M}_R \tag{19.2}$$

This ensures that the received signal is converted with optimal precision independently of the cable length; moreover, a controlled level of the signal at the adaptive digital equalizer input is required for achieving optimal convergence properties.

Near-end cross-talk cancellation

As illustrated in Chapter 16, *near-end cross-talk* (NEXT) cancellation is achieved by storing the transmit symbols a_k^N in a delay line and computing an estimated NEXT signal $\hat{u}_k^N$, which is then subtracted from the received signal x_k, that is

$$\tilde{x}_k = x_k - \hat{u}_k^N = x_k - \sum_{i=0}^{N_N-1} c_{i,k}^N\, a_{k-i}^N \tag{19.3}$$

where $\{c_{i,k}^N\}$, $i = 0, \ldots, N_N - 1$, are the coefficients of the adaptive NEXT canceller. Using the minimization of $E[\tilde{x}_k^2]$ as a criterion for updating the NEXT canceller coefficients (see Section 16.1), leads to the LMS algorithm

$$c_{i,k+1}^N = c_{i,k}^N + \mu_N\, \tilde{x}_k\, a_{k-i}^N \qquad 0 \le i \le N_N - 1 \tag{19.4}$$

where μ_N is the adaptation gain.

As discussed in Chapter 16, high-speed full duplex transmission over two separate wire pairs with NEXT cancellation and full-duplex transmission over a single pair with echo cancellation pose similar challenges. In the latter case, a hybrid is included to separate the two directions of transmission; a QPR-IV transceiver with NEXT cancellation can then be used also for full-duplex transmission over a single pair, as in this case the NEXT canceller acts as an echo canceller.

Decorrelation filter

After NEXT cancellation, the signal is filtered by a decorrelation filter, which is used to improve the convergence properties of the adaptive digital equalizer by reducing the correlation between the samples of the sequence $\{\tilde{x}_k\}$. The filtering operation performed by the DCF represents an approximate inversion of the PR-IV frequency response. The DCF has frequency response $1/(1-\beta\, e^{-j4\pi fT})$, with $0 < \beta < 1$, and provides at its output the signal

$$z_k = \tilde{x}_k + \beta\, z_{k-2} \tag{19.5}$$

Adaptive equalizer

The samples $\{z_k\}$ are stored in an elastic buffer, from which they are transferred into the equalizer delay line. Before describing this operation in more detail, we make some observations about the adaptive equalizer. As mentioned above, the received signal $x(t)$ is sampled in synchronism with the timing of the local transmitter. Due to the frequency offset between local and remote transmitter clocks, the phase of the remote transmitter clock will drift in time relative to the sampling phase. As the received signal is bandlimited to one half of the modulation rate, signal samples taken at the symbol rate are not affected by aliasing; hence, a fractionally-spaced equalizer is not required for a QPR-IV system. Furthermore, as the signal value $x(t)$ can be reconstructed from the T-spaced samples $\{z_k\}$, an equalizer of sufficient length acts also as an interpolator. The adaptive equalizer output signal is given by

$$y_k = \sum_{i=0}^{N_E-1} c_{i,k}^E \, z_{k-i} \tag{19.6}$$

where $\{c_{i,k}^E\}$, $i = 0, \ldots, N_E - 1$, denote the filter coefficients.

Compensation of the timing phase drift

The effect of timing phase drift can be compensated by continuously adjusting the equalizer coefficients. As a result of these adjustments, for a positive frequency offset of $+\delta/T$ Hz, i.e. for a frequency of the local transmitter clock larger than the frequency of the remote transmitter clock, the value of the i-th coefficient at time k is approximately equal to the value assumed by the $(i + 1)$-th coefficient $1/\delta$ modulation intervals earlier. In other words, the coefficients move to the left by one position relative to the equalizer delay line after $1/\delta$ modulation intervals; conversely, for a negative frequency offset of $-\delta/T$ Hz, the coefficients move to the right by one position relative to the equalizer delay line after $1/\delta$ modulation intervals. Hence, the center of gravity of the filter coefficients drifts. Proper operation of a finite-length equalizer requires that the coefficients be recentered; this is accomplished as follows:

- normally, for each equalizer output y_k one new signal z_k is transferred from the buffer into the equalizer delay line;
- periodically, after a given number of modulation intervals, the sums of the magnitudes of N' first and last coefficients are compared; if the first coefficients are too large, the coefficients are shifted by one position towards the end of the delay line (right shift) and two new signals z_k and z_{k+1} are transferred from the buffer into the delay line; if the last coefficients are too large, the coefficients are shifted by one position towards the beginning of the delay line (left shift) and no new signal is retrieved from the buffer.

To prevent buffer overflow or underflow, the rate of the remaining receiver operations is controlled so that the elastic buffer is kept half full on average, thus performing indirect timing recovery. The control algorithm used to adjust the VCO providing the timing signal for the remaining receiver operations is described in Chapter 14.

Adaptive equalizer coefficient adaptation

Overall system complexity is reduced if self-training adaptive equalization is employed, as in this case a start-up procedure with the transmission of a known training sequence is not needed. We describe here the operation of the adaptive equalizer and omit the presentation of the self-training algorithm, discussed in Section 15.4.

The MSE at the equalizer output is continuously monitored using seven-level tentative decisions. If the MSE is too large, self-training adaptive equalization is performed during a fixed time period T_{ST}. At the end of the self-training period, if the MSE is sufficiently small, equalizer operation is continued with the decision directed LMS algorithm

$$c_{i,k+1}^{E} = c_{i,k}^{E} - \mu_E \, z_{k-i} \, \hat{e}_k \qquad 0 \le i \le N_E - 1 \tag{19.7}$$

where $\hat{e}_k = y_k - (\hat{a}_k - \hat{a}_{k-2})$ is the error obtained using tentative decisions $\hat{a}_k$ on the transmitted quaternary symbols, and μ_E is the adaptation gain.

Convergence behavior of the various algorithms

We resort to computer simulations to study the convergence behavior of the adaptive digital NEXT canceller and of the adaptive equalizer. The length of the NEXT canceller is chosen equal to $N_N = 48$ to ensure that, in the worst case signal attenuation, the power of the residual NEXT is with high probability more than 30 dB below the power of the signal from the remote transmitter. In Figure 19.3, the residual NEXT statistical power at the canceller output is plotted versus the number of iterations for a worst-case cable length of $L = 100$ m and an adaptation gain $\mu_N = 2^{-18}$. In the same figure the statistical power in dB of the portion of the NEXT signal that cannot be cancelled due to finite NEXT-canceller length

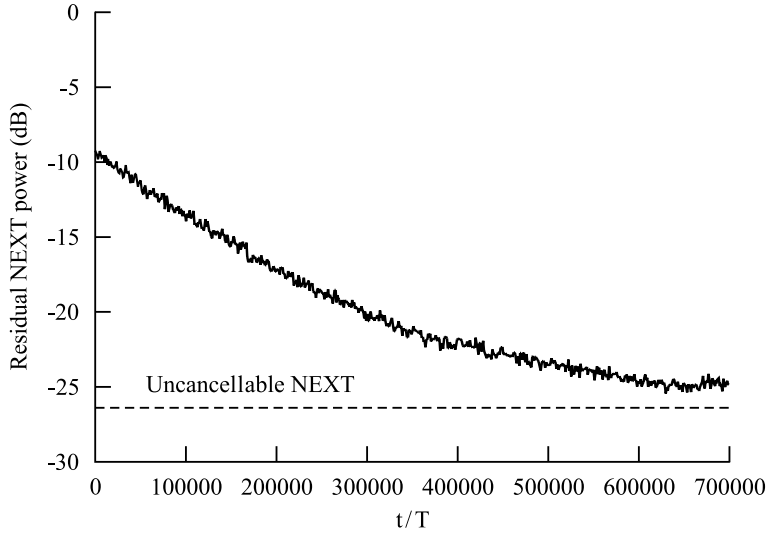
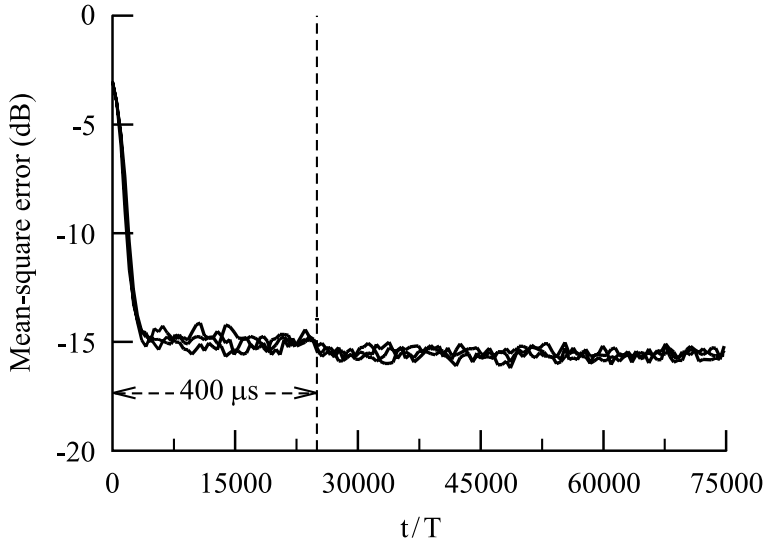
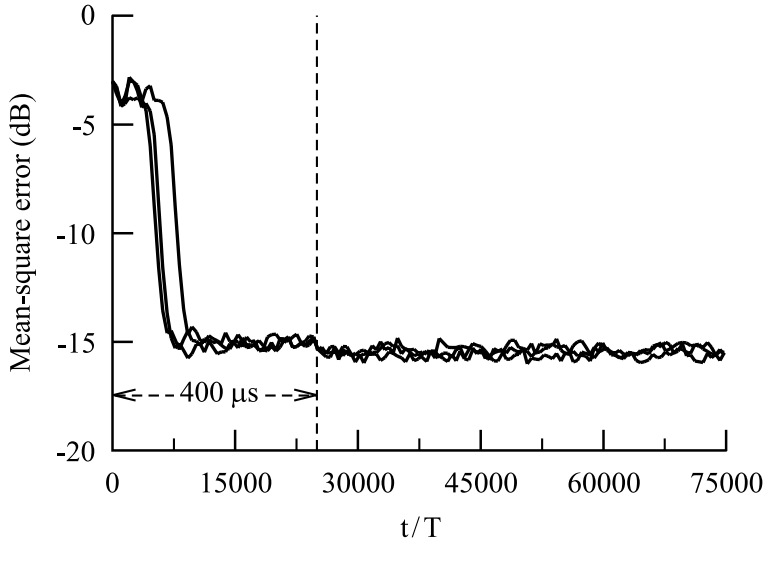


Figure 19.3. Convergence of the adaptive NEXT canceller. [*From [1], © 1995 IEEE.*]



(a)



(b)

Figure 19.4. Convergence of the adaptive equalizer for (a) best-case sampling phase and (b) worst-case sampling phase. [*From [1], © 1995 IEEE*.]

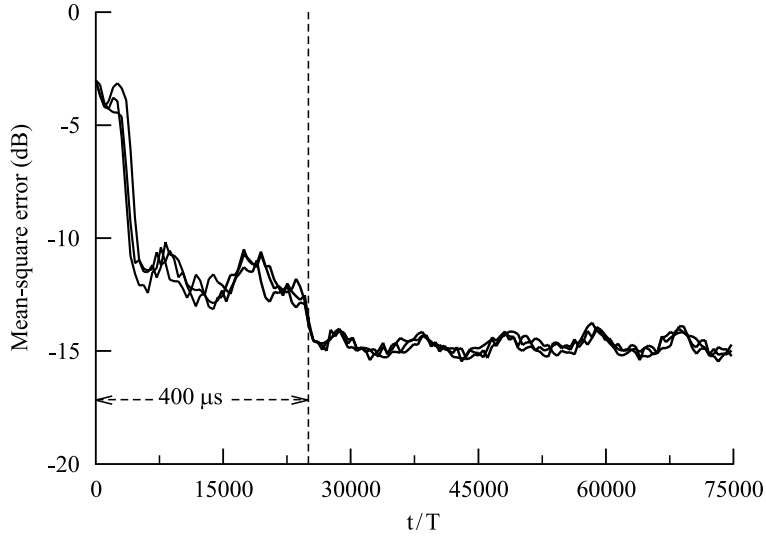


Figure 19.5. Convergence of the adaptive equalizer for worst-case timing phase drift. [*From [1], © 1995 IEEE.*]

is also indicated. For the simulations, the NEXT canceller was assumed to be realized in the distributed-arithmetic form (see Section 16.1).

The convergence of the MSE at the output of the equalizer in the absence of timing phase drift is shown in Figure 19.4a and b for best and worst-case sampling phase, respectively, and a value $\Gamma_{Tx} = 2E_{Tx}/N_0$ of 43 dB, where E_{Tx} is the average energy per modulation interval of the transmitted signal. The NEXT canceller is assumed to have converged to the optimum setting. An equalizer length of $N_E = 24$ is chosen, which guarantees that the mean-square interpolation error with respect to an ideal QPR-IV signal is less than -25 dB for the worst-case sampling phase. The self-training period is $T_{ST} \approx 400$ μs, corresponding to approximately $25000T$. The adaptation gains for self-training and decision directed adjustment have the same value $\mu_E = 2^{-9}$, that is chosen for best performance in the presence of a worst-case timing phase drift $\delta = 10^{-4}$. Figure 19.5 shows the mean-square error convergence curves obtained for $\delta = 10^{-4}$.

19.1.1 VLSI implementation

Adaptive digital NEXT canceller

As shown in Chapter 16, for the VLSI implementation of the adaptive digital NEXT canceller, a distributed-arithmetic filter presents significant advantages over a transversal filter in terms of implementation complexity. In a NEXT canceller distributed-arithmetic filter, the partial products that appear in the expression of a transversal filter output are not individually computed; evaluation of partial products is replaced by table look-up and shift-and-add operations of binary words. To compute the estimate of the NEXT signal to be subtracted

from the received signal, look-up values are selected by the bits in the NEXT canceller delay line and added by a carry-save adder. By segmenting the delay line of the NEXT canceller into sections of shorter lengths, a trade-off concerning the number of operations per modulation interval and the number of memory locations that are needed to store the look-up values is possible. The convergence of the look-up values to the optimum setting is achieved by an LMS algorithm.

If the delay line of the NEXT canceller is segmented into L sections with $K = N_N/L$ delay elements each, the NEXT canceller output signal is given by

$$\hat{u}_k^N = \sum_{i=0}^{N_N-1} a_{k-i}^N \, c_{i,k}^N = \sum_{\ell=0}^{L-1} \sum_{m=0}^{K-1} a_{k-\ell K-m}^N \, c_{\ell K+m,k}^N \tag{19.8}$$

In a distributed-arithmetic implementation, the quaternary symbol a_k^N is represented by the binary vector $[a_k^{(0)}, a_k^{(1)}]$, that is

$$a_k^N = \sum_{w=0}^{1} (2a_k^{(w)} - 1)2^w = \sum_{w=0}^{1} b_k^{(w)} \, 2^w \tag{19.9}$$

where $b_k^{(w)} = (2a_k^{(w)} - 1) \in \{-1, +1\}$. Introducing (19.9) into (19.8) we obtain (see (16.12))

$$\hat{u}_k^N = \sum_{\ell=0}^{L-1} \sum_{w=0}^{1} 2^w \left[\sum_{m=0}^{K-1} b_{k-\ell K-m}^{(w)} \, c_{\ell K+m,k}^N \right] \tag{19.10}$$

Equation (19.10) suggests that the filter output can be computed using a set of $L2^K$ look-up values that are stored in L look-up tables with 2^K memory locations each. Extracting the term $b_{k-\ell K}^{(w)}$ out of the square bracket in (19.10), to determine the output of a distributed-arithmetic filter with reduced memory size $L2^{K-1}$ we rewrite (19.10) as (see (16.13))

$$\hat{u}_k^N = \sum_{\ell=0}^{L-1} \sum_{w=0}^{1} 2^w \, b_{k-\ell K}^{(w)} \, d_k^N(i_{k,\ell}^{(w)}, \ell) \tag{19.11}$$

where $\{d_k^N(n, \ell)\}$, $n = 0, \ldots, 2^{K-1} - 1$, $\ell = 0, \ldots, L-1$, are the look-up values, and $i_{k,\ell}^{(w)}$ denotes the selected look-up address that is computed as follows:

$$i_{k,\ell}^{(w)} = \begin{cases} \displaystyle\sum_{m=1}^{K-1} a_{k-\ell K-m}^{(w)} \, 2^{m-1} & \text{if } a_{k-\ell K}^{(w)} = 1 \\ \displaystyle\sum_{m=1}^{K-1} \bar{a}_{k-\ell K-m}^{(w)} \, 2^{m-1} & \text{if } a_{k-\ell K}^{(w)} = 0 \end{cases} \tag{19.12}$$

where $\bar{a}_n$ is the one's complement of a_n. The expression of the LMS algorithm to update the look-up values of a distributed-arithmetic NEXT canceller takes the form

$$d_{k+1}^N(n, \ell) = d_k^N(n, \ell) + \mu_N \, \tilde{x}_k \sum_{w=0}^{1} 2^w \, b_{k-\ell K}^{(w)} \, \delta_{n-i_{k,\ell}^{(w)}} \tag{19.13}$$

$$n = 0, \ldots, 2^{K-1} - 1 \qquad \ell = 0, \ldots, L-1$$

where δ_n is the Kronecker delta. We note that at each iteration only those look-up values that are selected to generate the filter output are updated. The implementation of the NEXT canceller is further simplified by updating at each iteration only the look-up values that are addressed by the most significant bits of the symbols, i.e. those with index $w = 1$, stored in the delay line (see (16.23)). The block diagram of an adaptive distributed-arithmetic NEXT canceller is shown in Figure 19.6. In the QPR-IV transceiver, for the implementation of a NEXT canceller with a time span of $48T$, $L = 16$ segments with $K = 3$ delay elements each are employed. The look-up values are stored in 16 tables with four 16-bit registers each.

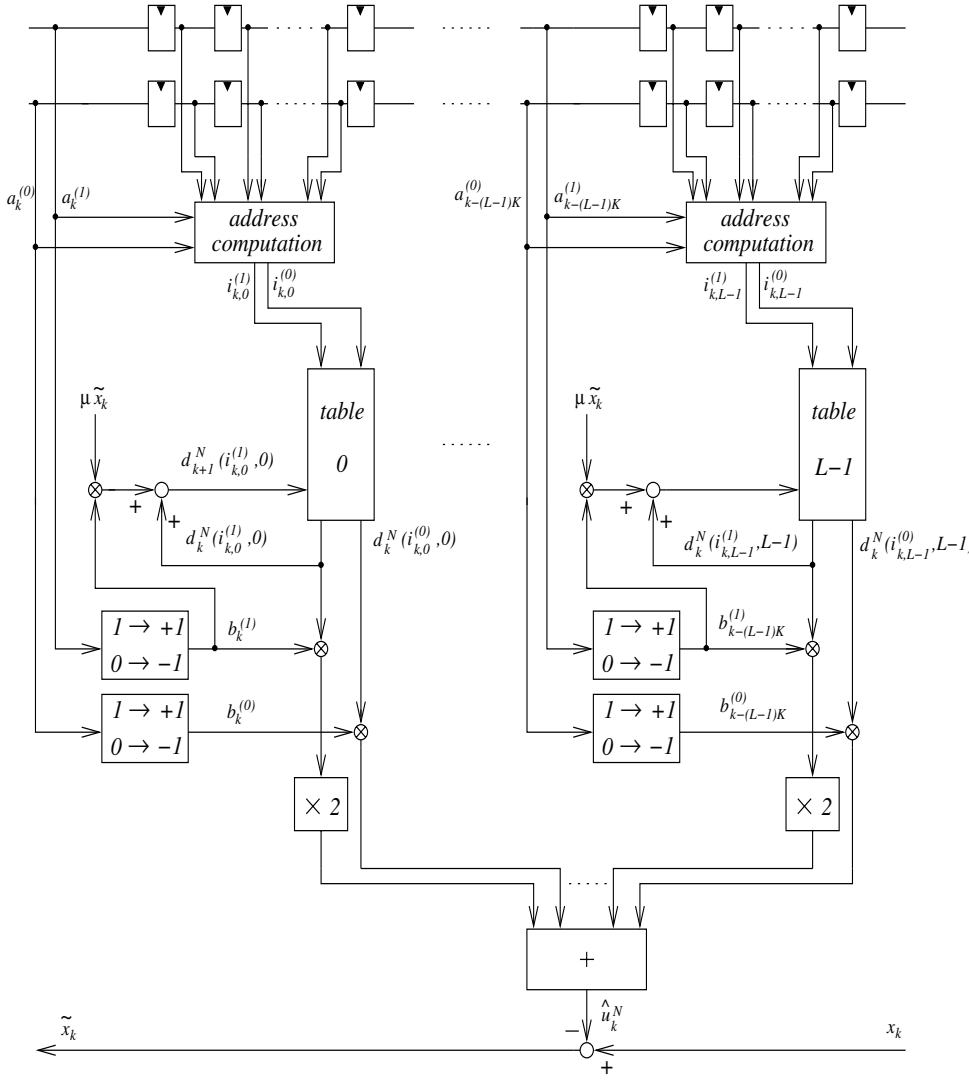


Figure 19.6. Adaptive distributed-arithmetic NEXT canceller. [*From [1], © 1995 IEEE.*]

Adaptive digital equalizer

As discussed in the previous section, the effect of the drift in time of the phase of the remote transmitter clock relative to the sampling phase is compensated by continuously updating and occasionally recentering the coefficients of the digital equalizer. The need for fast equalizer adaptation excludes a distributed-arithmetic approach for the implementation of the digital equalizer. An efficient solution for occasional recentering of the N_E equalizer coefficients is obtained by the following structure, which is based on an approach where N_E *multiply-accumulate* (MAC) units are employed.

According to (19.6), assuming that the coefficients are not time varying, at a given time instant k each MAC unit is presented with a different tap coefficient and carries out the multiplication of this tap coefficient with the signal sample z_k. The result is a partial product that is added to a value stored in the MAC unit, which represents the sum of partial products up until time instant k. The MAC unit that has accumulated N_E partial products provides the equalizer output signal at time instant k, $y_k = \sum_{i=0}^{N_E-1} c_i^E \, z_{k-i}$, and its memory is cleared to allow for the accumulation of the next N_E partial products. At this time instant, the MAC unit that has accumulated the result of $(N_E - 1)$ partial products has stored the term $\sum_{i=1}^{N_E-1} c_i^E \, z_{k-(i-1)}$. At time instant $(k+1)$ this unit computes the term $c_0^E \, z_{k+1}$ and provides the next equalizer output signal

$$y_{k+1} = c_0^E \, z_{k+1} + \sum_{i=1}^{N_E-1} c_i^E \, z_{k-(i-1)} \tag{19.14}$$

This MAC unit is then reset and its output will be considered again N_E time instants later.

Figure 19.7 depicts the implementation of the digital equalizer. The N_E coefficients $\{c_i^E\}$, $i = 0, \ldots, N_E - 1$, normally circulate in the delay line shown at the top of the figure. Except when recentering of the equalizer coefficients is needed, N_E coefficients in the delay line are presented each to a different MAC unit, and the signal sample z_k is input to all MAC units. At the next time instant, the coefficients are cyclically shifted by one position and the new signal sample z_{k+1} is input to all the units. The multiplexer shown at the bottom of the figure selects in turn the MAC unit that provides the equalizer output signal.

To explain the operations for recentering of the equalizer coefficients, we consider as an example a simple equalizer with $N_E = 4$ coefficients; in Figure 19.8, where for simplicity the coefficients are denoted by $\{c_0, c_1, c_2, c_3\}$, the coefficients and signal samples at the input of the 4 MAC units are given as a function of the time instant. At time instant k, the output of the MAC unit 0 is selected, at time instant $k + 1$ the output of the MAC unit 1 is selected, and so on. For a negative frequency offset between the local and remote transmitter clocks, a recentering operation corresponding to a left shift of the equalizer coefficients occasionally occurs as illustrated in the upper part of Figure 19.8. We note that as a result of this operation, a new coefficient c_4, initially set equal to zero, is introduced. We also note that signal samples with proper delay need to be input to the MAC units. A similar operation occurs for a right shift of the equalizer coefficients, as illustrated in the lower part of the figure; in this case a new coefficient c_{-1} , initially set equal to zero, is introduced. In the equalizer implementation shown in Figure 19.7, the control operations to select the filter coefficients and the signal samples are implemented by the multiplexer

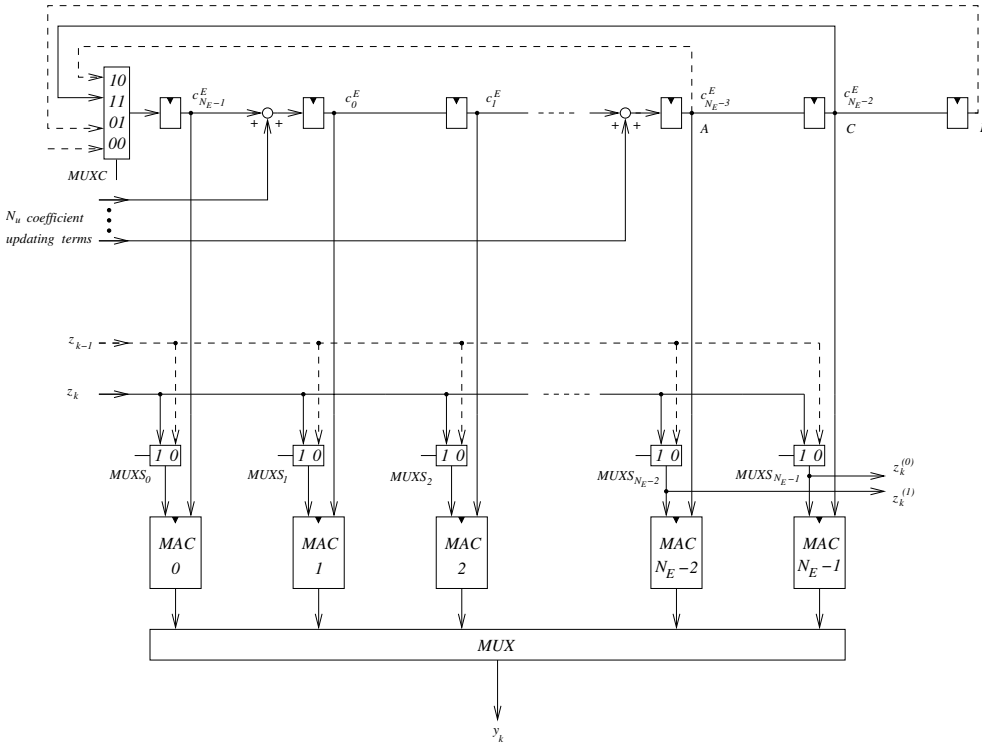


Figure 19.7. Digital adaptive equalizer: coefficient circulation and updating, and computation of output signal. [*From [1], © 1995 IEEE.*]

MUXC at the input of the delay line and by the multiplexers MUXS(0), ..., MUXS(N_E-1) at the input of the MAC units, respectively. A left or right shift of the equalizer coefficients is completed in N_E cycles. To perform a left shift, in the first cycle the multiplexer MUXC is controlled so that a new coefficient $c^E_{N_E} = 0$ is inserted into the delay line. During the following $(N_E - 1)$ cycles, the input of the delay line is connected to point B. After inserting the coefficient c^E_1 at the N_E-th cycle, the input of the delay line is connected to point C and normal equalizer operations are restored. For a right shift, the multiplexer MUXC is controlled so that during the first $N_E - 1$ cycles the input of the delay line is connected to point A. A new coefficient $c^E_{-1} = 0$ is inserted into the delay line at the N_E-th cycle and normal equalizer operations are thereafter restored. At the beginning of the equalizer operations, the equalizer coefficients are initialized by inserting the sequence $\{0, \ldots, 0, +1, 0, -1, 0, \ldots, 0\}$ into the delay line.

The adaptation of the equalizer coefficients in decision-directed mode is performed according to the LMS algorithm (19.7). However, to reduce implementation complexity, equalizer coefficients are not updated at every cycle; during normal equalizer operations, each coefficient is updated every N_E/N_U cycles by adding correction terms at N_U equally spaced fixed positions in the delay line, as shown in Figure 19.9. The architecture adopted for the computation of the correction terms is similar to the architecture for the computation

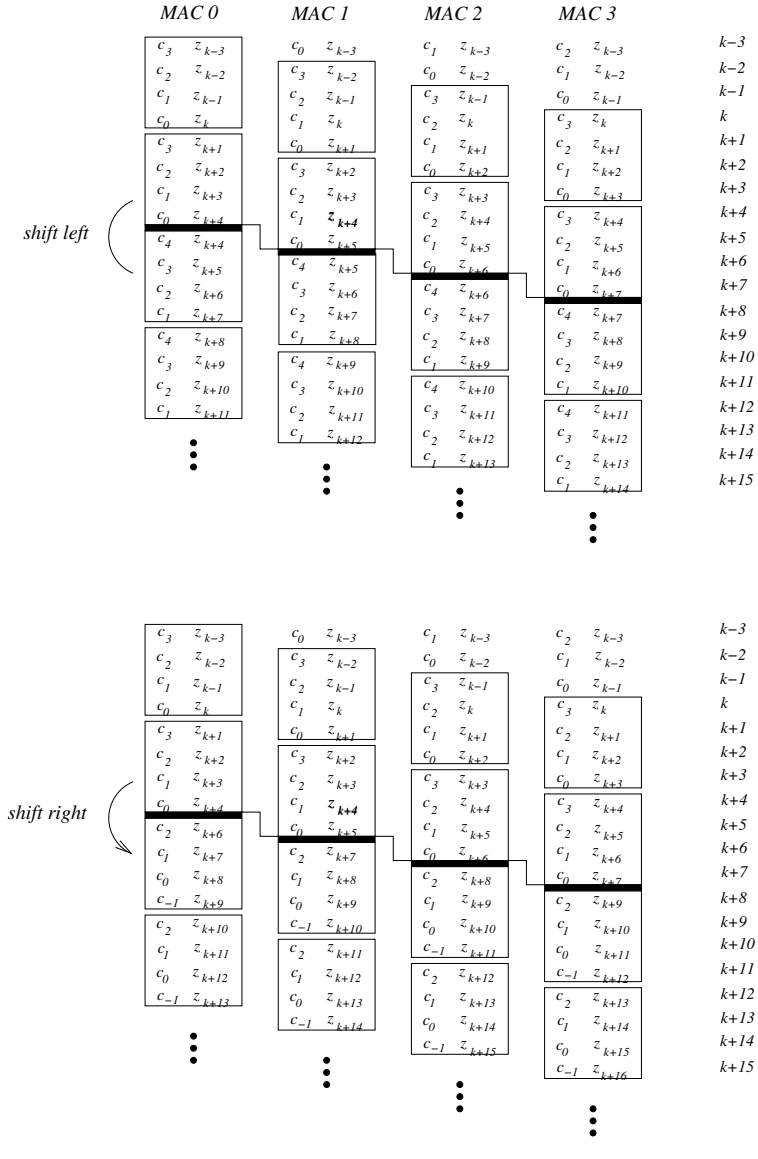


Figure 19.8. Coefficients and signals at the input of the multiply-accumulate (MAC) units during coefficient shifting. [*From [1], © 1995 IEEE.*]

of the equalizer output signal. The gradient components $-\mu_E\, z_{k-i}\, \hat{e}_k$, $i = 0, \ldots, N_E - 1$, are accumulated in N_E MAC units, as illustrated in the figure. The delay line stores the signal samples $z_k^{(0)}$ and $z_k^{(1)}$. The inputs to each MAC unit are given by the error signal and a signal from the delay line. The multiplexers at the input of each register in the delay line allow selecting the appropriate inputs to the MAC units in connection with the recentering of the equalizer coefficients. At each cycle, the output of N_U MAC units are selected and

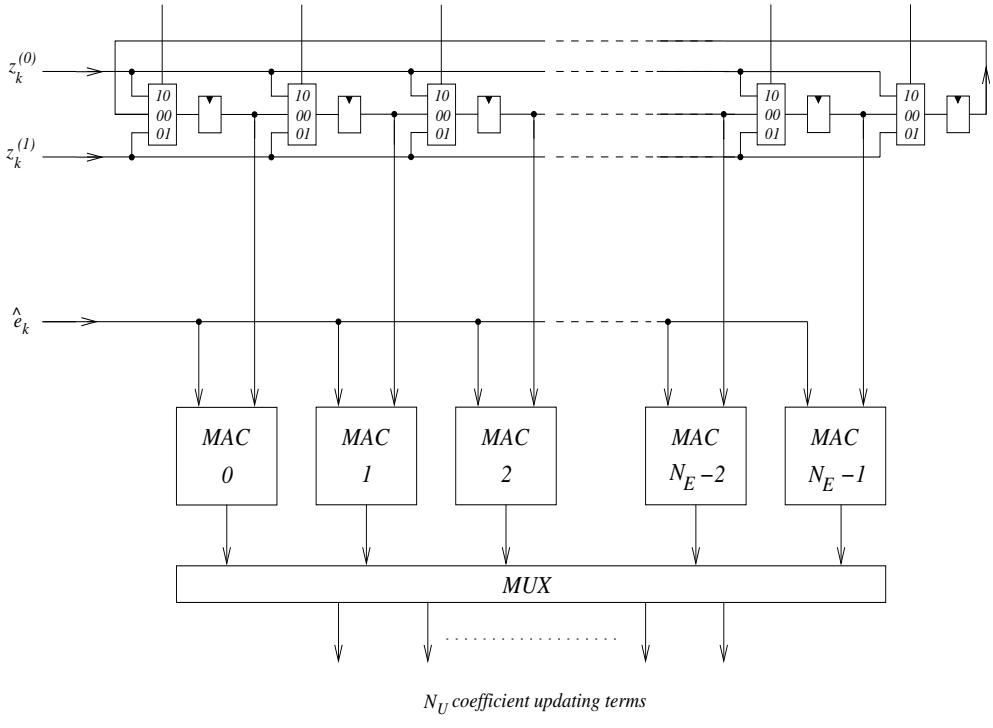


Figure 19.9. Adaptive digital equalizer: computation of coefficient adaptation. [*From [1], © 1995 IEEE.*]

input to the N_U adders in the delay line where the equalizer coefficients are circulating, as illustrated in Figure 19.7.

Timing control

An elastic buffer is provided at the boundary between the transceiver sections that operate at the transmit and receive timings. The signal samples at the output of the DCF are obtained at a rate that is given by the transmit timing and stored at the same rate into the elastic buffer. Signal samples from the elastic buffer are read at the same rate that is given by the receive timing. The VCO that generates the receive timing signal is controlled in order to prevent buffer underflow or overflow.

Let WP_k and RP_k denote the values of the two pointers that specify the write and read addresses, respectively, for the elastic buffer at the k-th cycle of the receiver clock. We consider a buffer with eight memory locations, so that $WP_k, RP_k \in \{0, 1, 2, \ldots, 7\}$. The write pointer is incremented by one unit at every cycle of the transmitter clock, while the read pointer is also incremented by one unit at every cycle of the receiver clock.

The difference pointer,

$$DP_k = WP_k - RP_k \pmod 8 \tag{19.15}$$

is used to generate a binary control signal $\Delta_k \in \{\pm 1\}$ that indicates whether the frequency of the VCO must be increased or decreased:

$$\Delta_k = \begin{cases} +1 & \text{if } DP_k = 4, 5 \\ -1 & \text{if } DP_k = 2, 3 \\ \Delta_{k-1} & \text{otherwise} \end{cases} \tag{19.16}$$

The signal Δ_k is input to a digital loop filter which provides the control signal to adjust the VCO. If the loop filter comprises both a proportional and an integral term, with corresponding gains of μ_τ and $\mu_{\Delta\tau}$, respectively, the resulting second-order phase-locked loop is described by (see Section 14.7)

$$\begin{aligned} \tau_{k+1} &= \tau_k + \mu_\tau \Delta_k + \Delta\tau_k \\ \Delta\tau_{k+1} &= \Delta\tau_k + \mu_{\Delta\tau} \Delta_k \end{aligned} \tag{19.17}$$

where τ_k denotes the difference between the phases of the transmit and receive timing signals. With a proper setting of the gains μ_τ and $\mu_{\Delta\tau}$, the algorithm (19.17) allows for correct initial frequency acquisition of the VCO and guarantees that the write and read pointers do not overrun each other during steady-state operations.

For every time instant k, the two consecutive signal samples stored in the memory locations with the address values RP_k and $(RP_k - 1)$ are read and transferred to the equalizer. These signal samples are denoted by z_k and z_{k-1} in Figure 19.10. When a recentering of the equalizer coefficients has to take place, for one cycle of the receiver clock the read pointer is either not incremented (left shift), or incremented by two units (right shift). These operations are illustrated in the figure, where the elastic buffer is represented as a circular memory. We note that by the combined effect of the timing control scheme and the recentering of the adaptive equalizer coefficients the frequency of the receive timing signal equals on average the modulation rate at the remote transceiver.

Viterbi detector

For the efficient implementation of near MLSD of QPR-IV signals, we consider the reduced-state Viterbi detector of Example 8.12.1 on page 687. In other words, the signal samples at the output the of $(1 - D^2)$ partial response channel are viewed as being generated by two interlaced $(1 - D')$ dicode channels, where $D' = D^2$ corresponds to a delay of two modulation intervals, $2T$. The received signal samples are hence deinterlaced into even and odd time-indexed sequences. The Viterbi algorithm using a 2-state trellis is performed independently for each sequence. This reduced-state Viterbi algorithm retains at any time instant k only the two states with the smallest and second smallest metrics and their survivor sequences, and propagates the difference between these metrics instead of two metrics. Because the minimum distance error events in the partial-response trellis lead to quasi-catastrophic error propagation, a sufficiently long path memory depth is needed. A path memory depth of $64T$ has been found to be appropriate for this application.

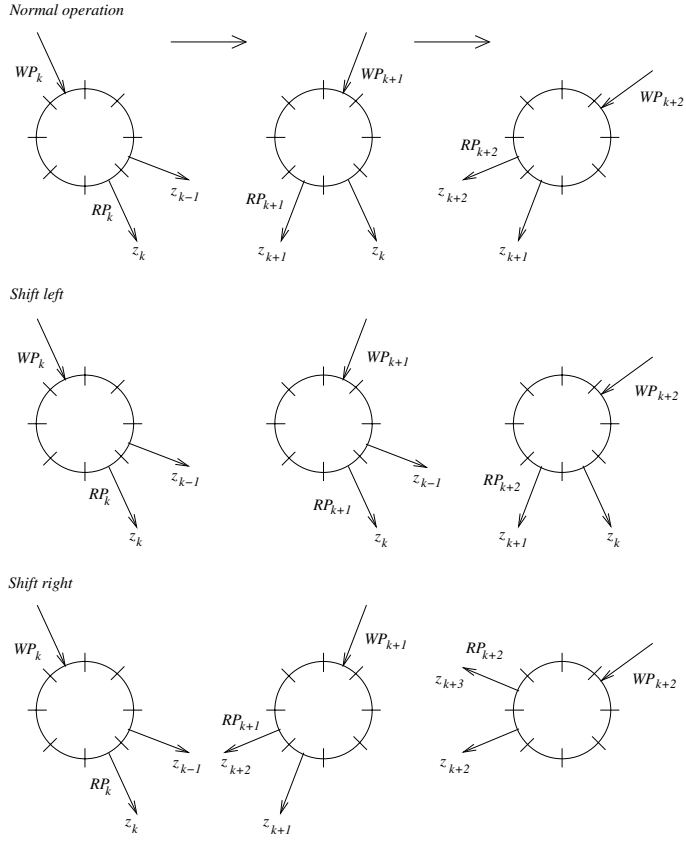


Figure 19.10. Elastic buffer: control of the read pointer.

19.2 Design of a dual duplex transmission system at 100 Mbit/s

We now describe the 100BASE-T2 system for fast Ethernet mentioned in Section 17.1.2 [4, 2].

Dual duplex transmission

The characteristics of the transmission channel used to develop the 100BASE-T2 standard are described in Section 4.4. Figure 4.23 indicates that near-end cross-talk (NEXT) represents the main disturbance for transmission at high data rates over UTP-3 cables. As illustrated in Figure 19.11, showing the principle of dual duplex transmission (see also Section 16.1), *self NEXT* is defined as NEXT from the transmitter output to the receiver input of the same transceiver, and can be cancelled by adaptive filters as discussed in Chapter 16. *Alien NEXT* is defined instead as NEXT from the transmitter output to the receiver input of another transceiver; this is generated in the case of simultaneous transmission over multiple links within one multi-pair cable, typically with 4 or 25 pairs. Suppression of alien NEXT

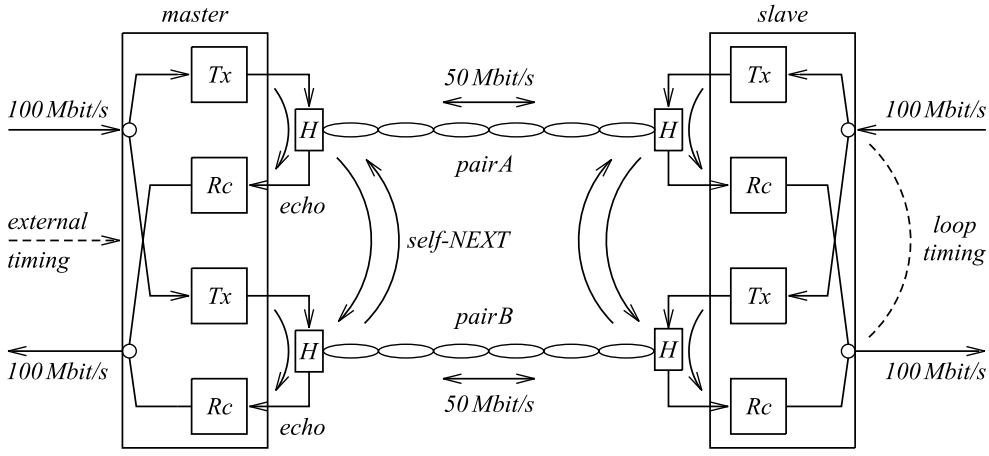


Figure 19.11. Dual duplex transmission over two wire pairs. [*From [2], © 1997 IEEE.*]

from other transmissions in multi-pair cables and far-end cross-talk (FEXT), although normally not very significant, requires specific structures (see, for example, Section 16.4).

To achieve best performance for data transmission over UTP-3 cables, signal bandwidth must be confined to frequencies not exceeding 30 MHz. As shown in Chapter 17, this restriction is further mandated by the requirement to meet FCC and CENELEC class B limits on emitted radiation from communication systems. These limits are defined for frequencies above 30 MHz. Twisted pairs used in UTP-3 cables have fewer twists per unit of length and generally exhibit a lower degree of homogeneity than pairs in UTP-5 cables; therefore transmission over UTP-3 cables produces a higher level of radiation than over UTP-5 cables. Thus it is very difficult to comply with the class B limits if signals containing spectral components above 30 MHz are transmitted over UTP-3 cables.

As illustrated in Figure 19.11, for 100BASE-T2 a *dual duplex* baseband transmission concept was adopted. Bidirectional 100 Mbit/s transmission over two pairs is accomplished by full duplex transmission of 50 Mbit/s streams over each of two wire pairs. The lower modulation rate and/or spectral modulation efficiency required per pair for achieving the 100 Mbit/s aggregate rate represents an obvious advantage over *mono duplex* transmission, where one pair would be used to transmit only in one direction and the other to transmit only in the reverse direction. Dual duplex transmission requires two transmitters and two receivers at each end of a link, as well as separation of the simultaneously transmitted and received signals on each wire pair. Sufficient separation cannot be accomplished by analog hybrid circuits only. In 100BASE-T2 transceivers it is necessary to suppress residual echoes returning from the hybrids and impedance discontinuities in the cable as well as self NEXT by adaptive digital echo and NEXT cancellation. Furthermore, by sending transmit signals with nearly 100% excess bandwidth, received 100BASE-T2 signals exhibit spectral redundancy that can be exploited to mitigate the effect of alien NEXT by adaptive digital equalization. It will be shown later in this chapter that, for digital NEXT cancellation and equalization as well as echo cancellation in the case of dual-duplex transmission, dual-duplex and mono-duplex schemes require a comparable number of multiply-add operations per second.

The dual transmitters and receivers of a 100BASE-T2 transceiver will henceforth be referred to simply as transmitter and receiver. Signal transmission in 100BASE-T2 systems takes place in an uninterrupted fashion over both wire pairs in order to maintain timing synchronization and the settings of adaptive filters at all times. Quinary pulse-amplitude baseband modulation at the rate of 25 MBaud is employed for transmission over each wire pair. The transmitted quinary symbols are randomized by *side-stream scrambling*. The redundancy of the quinary symbol sets is needed to encode 4-bit data *nibbles*, to send between data packets an idle sequence that also conveys information about the status of the local receiver, and to insert special delimiters marking the beginning and end of data packets.

Physical layer control

The diagram in Figure 19.12 shows in a simplified form the operational states defined for the 100BASE-T2 physical layer. Upon power-up or following a request to re-establish a link, an auto-negotiation process is executed during which two stations connected to a link segment advertise their transmission capabilities by a simple pulse-transmission technique. While auto-negotiation is in progress, the 100BASE-T2 transmitters remain silent. If the physical layers of both stations are capable of 100BASE-T2 operation, the auto-negotiation process further determines a master/slave relation between the two 100BASE-T2 transceivers: the master transceiver will employ an externally provided reference clock for its transmit and receive operations. The slave transceiver will extract timing from the received signal, and use this timing for its transmitter operations. This operation is usually referred to as *loop timing*. If signals were transmitted in opposite directions with independent clocks, signals

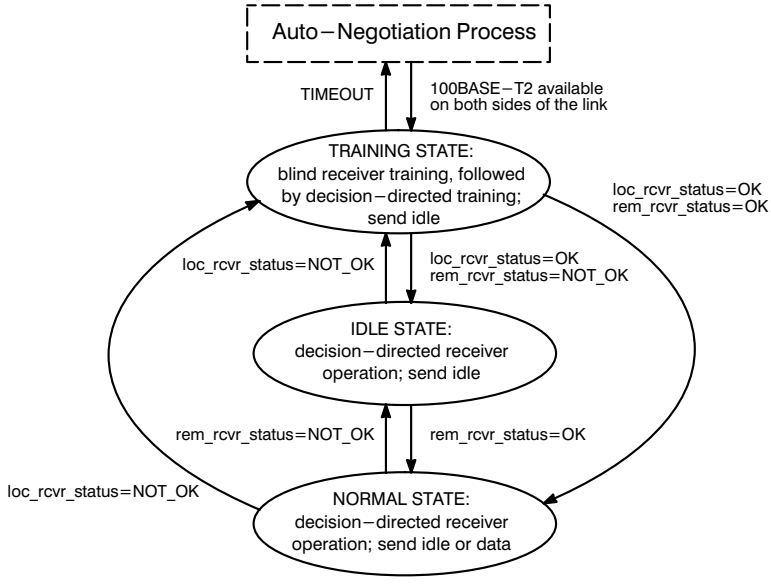


Figure 19.12. State diagram of 100BASE-T2 physical layer control. [*From [2], © 1997 IEEE.*]

received from the remote transmitter would generally shift in phase relative to the also-received echo and self-NEXT signals, as discussed in the previous section. To cope with this effect some form of interpolation would be required, which can significantly increase the transceiver complexity.

After auto-negotiation is completed, both 100BASE-T2 transceivers enter the TRAINING state. In this state a transceiver expects to receive an idle sequence and also sends an idle sequence, which indicates that its local receiver is not yet trained (`loc_rcvr_status =NOT_OK`). When proper local receiver operation has been achieved by blind training and then by further decision-directed training, a transition to the IDLE state occurs. In the IDLE state a transceiver sends an idle sequence expressing normal operation at its receiver (`loc_rcvr_status = OK`) and waits until the received idle sequence indicates correct operation of the remote receiver (`rem_rcvr_status = OK`). At this time a transceiver enters the NORMAL state, during which data nibbles or idle sequences are sent and received as demanded by the higher protocol layers. The remaining transitions shown in the state diagram of Figure 19.12 mainly define recovery functions.

The *medium independent interface* (MII) between the 100BASE-T2 physical layer and higher protocol layers is the same as for the other 10/100 Mbit/s IEEE 802.3 physical layers. If the control line `TX_EN` is inactive, the transceiver sends an idle sequence. If `TX_EN` is asserted, 4-bit data nibbles TXD(3:0) are transferred from the MII to the transmitter at the transmit clock rate of 25 MHz. Similarly, reception of data results in transferring 4-bit data nibbles RXD(3:0) from the receiver to the MII at the receive clock of 25 MHz. Control line `RX_DV` is asserted to indicate valid data reception. Other control lines, such as CRS (carrier sense) and COL (collision), are required for CSMA/CD specific functions.

Coding and decoding

The encoding and decoding rules for 100BASE-T2 are now described. During the k-th modulation interval, symbols a_k^A and a_k^B from the quinary set $\{-2, -1, 0, +1, +2\}$ are sent over pair A and pair B, respectively. The encoding functions are designed to meet the following objectives:

- the symbols $-2, -1, 0, +1, +2$ occur with probabilities 1/8, 1/4, 1/4, 1/4, 1/8, respectively;
- idle sequences and data sequences exhibit identical power spectral densities;
- reception of an idle sequence can rapidly be distinguished from reception of data;
- scrambler state, pair A and pair B assignment, and temporal alignment and polarities of signals received on these pairs can easily be recovered from a received idle sequence.

At the core of idle sequence generation and side-stream scrambling is a binary *maximum-length shift-register* (MLSR) sequence $\{p_k\}$ (see Appendix 3.A) of period $2^{33} - 1$. One new bit of this sequence is produced at every modulation interval. The transmitters in the master and slave transceivers generate the sequence $\{p_k\}$ using feedback polynomials $g_M(x) = 1 + x^{13} + x^{33}$ and $g_S(x) = 1 + x^{20} + x^{33}$, respectively. The encoding operations

are otherwise identical for the master and slave transceivers. From delayed elements $\{p_k\}$ four *derived bits* are obtained at each modulation interval as follows:

$$\begin{aligned} x_k &= p_{k-3} \oplus p_{k-8} \\ y_k &= p_{k-4} \oplus p_{k-6} \\ a_k &= p_{k-1} \oplus p_{k-5} \\ b_k &= p_{k-2} \oplus p_{k-12} \end{aligned} \tag{19.18}$$

where $\oplus$ denotes modulo 2 addition. The sequences $\{x_k\}$, $\{y_k\}$, $\{a_k\}$, and $\{b_k\}$ represent shifted versions of $\{p_k\}$, that differ from $\{p_k\}$ and from each other only by large delays. When observed in a constrained time window, the five sequences appear as mutually uncorrelated sequences. Figures 19.13 and 19.14 illustrate the encoding process for the idle mode and data mode, respectively. Encoding is based in both cases on the generation of pairs of two-bit vectors $(\mathbf{X}_k, \mathbf{Y}_k)$, $(\mathbf{S}_k^a, \mathbf{S}_k^b)$, and $(\mathbf{T}_k^a, \mathbf{T}_k^b)$, and Gray-code mapping of $(\mathbf{T}_k^a, \mathbf{T}_k^b)$ into symbol pairs (D_k^a, D_k^b), where $D_k^\gamma \in \{-2, -1, 0, +1\}$, $\gamma = a, b$. The generation of these quantities is determined by the sequences $\{x_k\}$, $\{y_k\}$ and $\{p_k\}$, the even/odd state of the time index k (equal to $2n$ or $2n+1$), and the local receiver status. Finally, pairs of transmit symbols (a_k^A, a_k^B) are obtained by scrambling the signs of (D_k^a, D_k^b) with the sequences $\{a_k\}$ and $\{b_k\}$. In Figure 19.15, the symbol pairs transmitted in the idle and data modes are depicted as two-dimensional signal points.

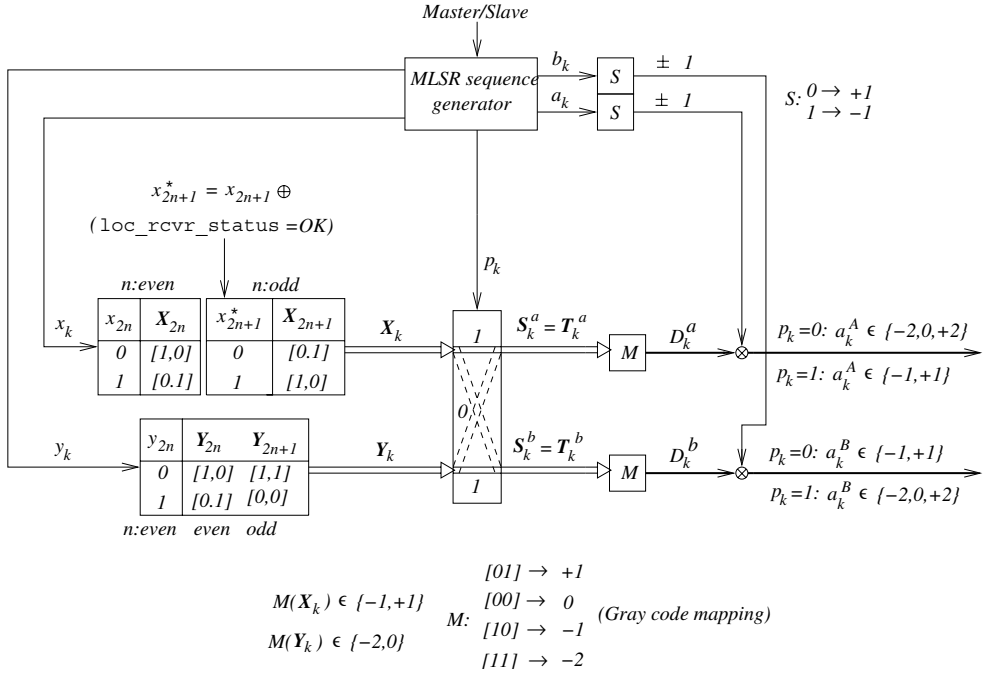


Figure 19.13. Signal encoding during idle mode.

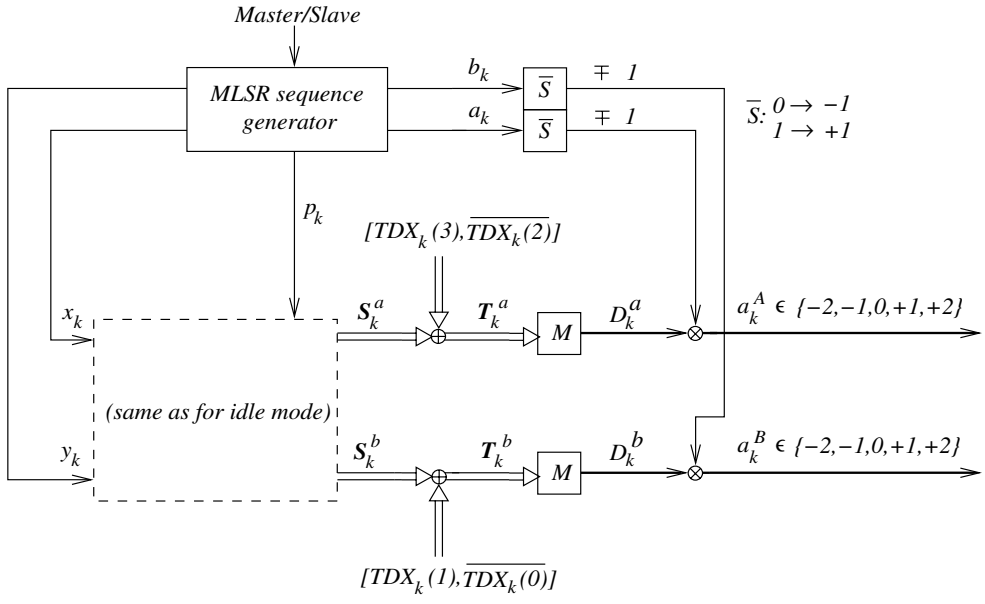


Figure 19.14. Signal encoding during data mode.

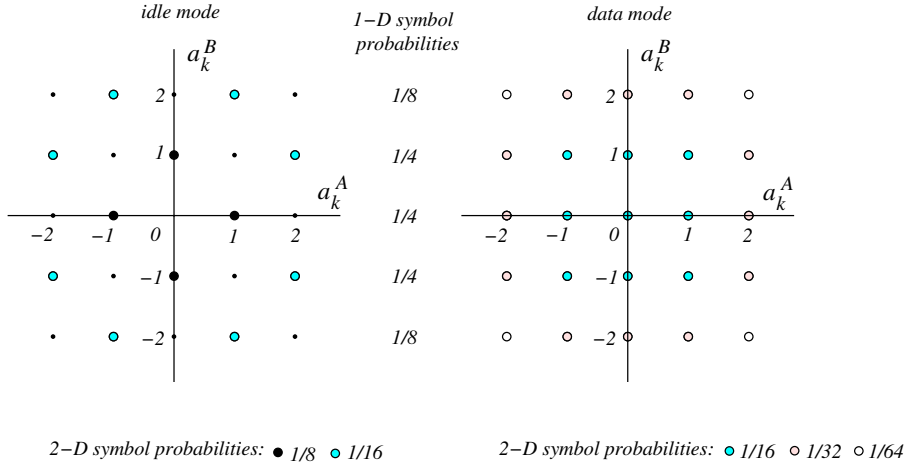


Figure 19.15. Two-dimensional symbols sent during idle and data transmission.

We note that, in idle mode, if $p_k = 1$ then symbols $a_k^A \in \mathcal{A}_x = \{-1, +1\}$ and $a_k^B \in \mathcal{A}_y = \{-2, 0, +2\}$ are transmitted; if $p_k = 0$ then $a_k^A \in \mathcal{A}_y$ and $a_k^B \in \mathcal{A}_x$ are transmitted. This property enables a receiver to recover a local replica of $\{p_k\}$ from the two received quinary symbol sequences.

The associations of the two sequences with pair A and pair B can be checked, and a possible temporal shift between these sequences can be corrected. Idle sequences have the

further property that in every two-symbol interval with even and odd time indices k, two symbols ± 1, one symbol 0, and one symbol ± 2 occur. The signs depend on the receiver status of the transmitting transceiver and on the elements of the sequences $\{x_k\}$, $\{y_k\}$, $\{a_k\}$ and $\{b_k\}$. Once a receiver has recovered $\{p_k\}$, these sequences are known, and correct signal polarities, the even/odd state of the time index k, and remote receiver status can be determined.

In data mode, the two-bit vectors $\mathbf{S}_k^a$ and $\mathbf{S}_k^b$ are employed to side-stream scramble the data nibble bits. Compared to the idle mode, the signs of the transmitted symbols are scrambled with opposite polarity. In the event that detection of the delimiter marking transitions between idle mode and data mode fails due to noise, a receiver can nevertheless rapidly distinguish an idle sequence from a data sequence by inspecting the signs of the two received ± 2 symbols. As mentioned above, during the transmission of idle sequences one symbol ± 2, i.e. with absolute value equal to 2, occurs in every two-symbol interval.

The previous description does not yet explain the generation of delimiters. A *start-of-stream delimiter* (SSD) indicates a transition from idle-sequence transmission to sending packet data. Similarly, an *end-of-stream delimiter* (ESD) marks a transition from sending packet data to idle sequence transmission. These delimiters consist of two consecutive symbol pairs ($a_k^A = \pm 2, a_k^B = \pm 2$) and ($a_{k+1}^A = \pm 2, a_{k+1}^B = 0$). The signs of symbols ± 2 in a SSD and in an ESD are selected opposite to the signs normally used in the idle mode and data mode, respectively. The choice of these delimiters allows detection of mode transitions with increased robustness against noise.

19.2.1 Signal processing functions

The principal signal processing functions performed in a 100BASE-T2 transmitter and receiver are illustrated in Figure 19.16. The digital-to-analog and analog-to-digital converters operate synchronously, although possibly at different multiples of 25 MBaud symbol rate. Timing recovery from the received signals, as required in slave transceivers, is not shown. This function can be achieved, for example, by exploiting the strongly cyclostationary nature of the received signals (see Chapter 14).

The 100BASE-T2 transmitter

At the transmitter, pairs of quinary symbols $(a_k^{A_T}, a_k^{B_T})$ are generated at the modulation rate of $1/T = 25$ MBaud. The power spectral density of transmit signals $s_A(t)$ and $s_B(t)$ has to conform to the spectral template given in Figure 19.17. An excess bandwidth of about 100%, beyond the Nyquist frequency of 12.5 MHz is specified to allow for alien NEXT suppression in the receiver by adaptive fractionally spaced equalization, as explained below. The transmit signals are produced by *digital pulse-shaping and interpolation filters* (DFTs), conversion from digital to analog signals at a multiple rate of 25 Msamples/s, and *analog transmit filters* (ATFs). The PSD obtained with a particular implementation in which the digital-to-analog conversion occurs at a sampling rate of 100 Msamples/s (oversampling factor $F_0 = 4$) is included in Figure 19.17.

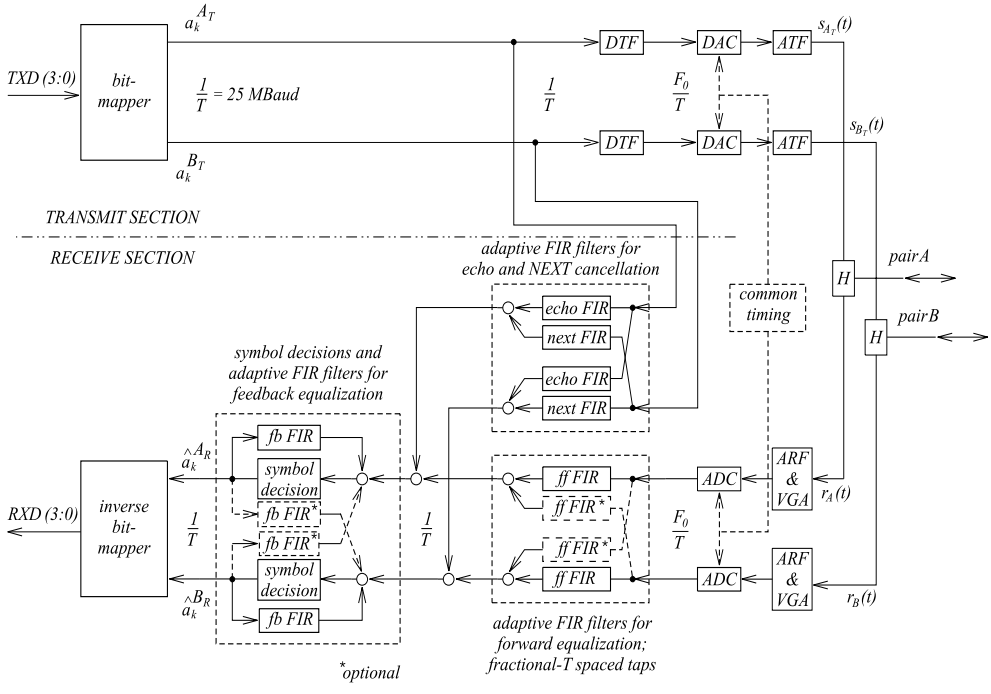


Figure 19.16. Principal signal processing functions performed in a 100BASE-T2 transceiver. [*From [2] © 1997 IEEE.*]

The 100BASE-T2 receiver

The receive signals $r_A(t)$ and $r_B(t)$ are bandlimited by *analog receive filters* (ARFs) to approximately 25 MHz, adjusted in amplitude by *variable gain amplifiers* (VGAs) and converted from analog-to-digital at a multiple rate of 25 Msamples/s. For the following discussion a sampling rate of 50 Msamples/s will be assumed (oversampling factor $F_0 = 2$, or equivalently sampling at the rate $2/T$).

The remaining receiver operations are performed digitally. Before detection of the pairs of quinary symbols $(a_k^{A_R}, a_k^{B_R})$ transmitted by the remote transceiver, an adaptive decision-feedback equalizer (DFE) structure together with adaptive echo and self NEXT cancellers is employed, as shown in Figure 19.16. The forward equalizer sections of the DFE operate on $T/2$-spaced input signals. The estimated echo and self NEXT signals are subtracted from the T-spaced equalizer output signals. When signal attenuation and disturbances increase with frequency, as is the case for 100BASE-T2 transmission, a DFE receiver provides noticeably higher noise immunity compared to that achieved by a receiver with linear forward equalization only. In Figure 19.16 additional feedback filters for DC restoration are not shown. In a complete receiver implementation, these filters are needed to compensate for a spectral null at DC introduced by linear transform coupling. This spectral notch may be broadened in a well-defined manner by the analog receive filters and compensated for by non-adaptive IIR filters.

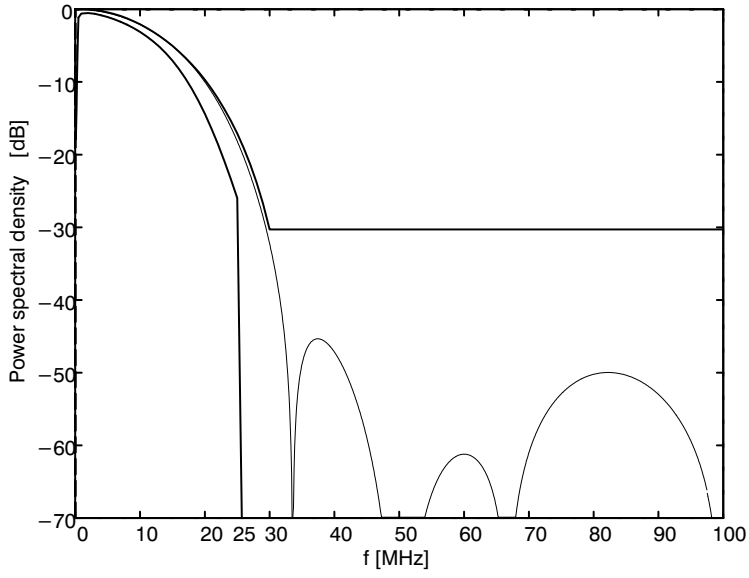


Figure 19.17. Spectral template specified by the 100BASE-T2 standard for the power spectral density of transmit signals and achieved power spectral density for a particular transmitter implementation comprising a 5-tap digital transmit filter, 100 MHz D/A conversion, and a 3° order Butterworth analog transmit filter. [*From [2], © 1997 IEEE.*]

The use of forward equalizers with $T/2$-spaced coefficients serves two purposes. First, as illustrated in Section 8.4, equalization becomes essentially independent of the sampling phase. Second, when the received signals exhibit excess bandwidth, the superposition of spectral input-signal components at frequencies f and $f - 1/T$, for $0 < f < 1/(2T)$, in the T-sampled equalizer output signals, can mitigate the effects of synchronous interference and asynchronous disturbances, as shown in Appendix 19.A. Interference suppression achieved in this manner can be interpreted as a frequency diversity technique [5]. Inclusion of the optional cross-coupling feedforward and backward filters shown in Figure 19.16 significantly enhances the capability of suppressing alien NEXT. This corresponds to adding space diversity at the expense of higher implementation complexity. Mathematical explanations for the ability to suppress synchronous and asynchronous interference with the cross-coupled forward equalizer structure are given in the Appendix 19.A. This structure permits the complete suppression of the alien NEXT interferences stemming from another 100BASE-T2 transceiver operating in the same multi-pair cable at identical clock rate. Alternatively, the interference from a single asynchronous source, e.g. alien NEXT from 10BASE-T2 transmission over an adjacent pair, can also be eliminated.

The 100BASE-T2 standard does not provide the transmission of specific training sequences. Hence, for initial receiver-filter adjustments, blind adaptation algorithms must be employed. When the mean-square errors at the symbol-decision points reach sufficiently low values, filter adaptation is continued in decision directed mode based on quinary symbol

decisions. The filter coefficients can henceforth be continuously updated by the LMS algorithm to track slow variations of channel and interference characteristics.

The 100BASE-T2 Task Force adopted a symbol-error probability target value of 10^{-10} that must not be exceeded under the worst-case channel attenuation and NEXT coupling conditions when two 100BASE-T2 links operate in a four-pair UTP-3 cable, that are illustrated in Figure 4.23. During the development of the standard, the performance of candidate 100BASE-T2 systems has been extensively investigated by computer simulation. For the scheme ultimately adopted, it was shown that by adopting time spans of $32T$ for the echo and self NEXT cancellers, $12T$ for the forward filters, and $10T$ for the feedback filters, the MSEs at the symbol-decision points remain consistently below a value corresponding to a symbol-error probability of 10^{-12}.

Computational complexity of digital receive filters

The digital receive filters account for most of the transceiver implementation cost. It is worthwhile comparing the filter complexities for a dual duplex and a mono duplex scheme. Intuitively, the dual-duplex scheme may appear to be more complex, because it requires two transceivers. We define the complexity of a finite-impulse response FIR as

$$\begin{aligned} \text{Filter complexity} &= \text{time span} \times \text{input sampling rate} \times \text{output sampling rate} \\ &= \text{number of coefficients} \times \text{output sampling rate} \\ &= \text{number of multiply-and-adds per second} \end{aligned} \tag{19.19}$$

Note that the time span of an FIR filter is given in seconds by the product of the number of filter coefficients times the sampling period of the input signal. Transmission in a four-pair cable environment with suppression of alien NEXT from a similar transceiver is considered. Only the echo and self NEXT cancellers and forward equalizers will be compared. Updating of filter coefficients will be ignored.

For dual duplex transmission, the modulation rate is 25 MBaud and signals are transmitted with about 100% excess bandwidth. Echo and self NEXT cancellation requires four FIR filters with time spans T_C and input/output rates of 25 Msamples/s. For equalization and alien NEXT suppression, four forward FIR filters with time spans T_E, an input rate of 50 Msamples/s and an output rate of 25 Msamples/s are needed.

The modulation rate for mono duplex transmission is $1/T = 50$ MBaud and signals are transmitted with no significant excess bandwidth. Hence, both schemes transmit within a comparable bandwidth (≤ 25 MHz). For an obvious receiver structure that does not allow alien NEXT suppression, one self NEXT canceller with time span T_C and input/output rates of 50 Msamples/s, and one equalizer with time span T_E and input/output rates of 50 Msamples/s will be needed. However, for a fair comparison, a mono duplex receiver must have the capability to suppress alien NEXT from another mono duplex transmission. This can be achieved by receiving signals not only from the receive pair but also in the reverse direction of the transmit pair, and combining this signal via a second equalizer with the output of the first equalizer. The additionally required equalizer exhibits the same complexity as the first equalizer.

The filter complexities for the two schemes are summarized in Table 19.1. As the required time spans of the echo and self NEXT cancellers and the forward equalizers are similar for

Table 19.1 Complexities of filtering for two transmission schemes.

	dual duplex	mono duplex
Echo and self NEXT cancellers	$4 \times T_C \times 25 \times 10^{12}$	$1 \times T_C \times 50 \times 50 \times 10^{12}$
Forward equalizers	$4 \times T_E \times 50 \times 25 \times 10^{12}$	$2 \times T_E \times 50 \times 50 \times 10^{12}$

the two schemes, it can be concluded that the two schemes have the same implementation complexity. The arguments can be extended to the feedback filters. Finally, we note that with the filter time spans considered in the preceding section ($T_C = 32T$, $T_E = 12T$ and $T_{Fb} = 10T$), in a 100BASE-T2 receiver on the order of 10^{10} multiply-and-add operations/s need to be executed.

Bibliography

[1] G. Cherubini, S. Ölçer, and G. Ungerboeck, "A quaternary partial response class-IV transceiver for 125 Mbit/s data transmission over unshielded twisted-pair cables: principles of operation and VLSI realization", *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 1656–1669, Dec. 1995.

[2] G. Cherubini, S. Ölçer, G. Ungerboeck, J. Creigh, and S. K. Rao, "100BASE-T2: a new standard for 100 Mb/s ethernet transmission over voice-grade cables", *IEEE Communications Magazine*, vol. 35, pp. 115–122, Nov. 1997.

[3] G. Cherubini, S. Ölçer, and G. Ungerboeck, "Adaptive analog equalization and receiver front-end control for multilevel partial-response transmission over metallic cables", *IEEE Trans. on Communications*, vol. 44, pp. 675–685, June 1996.

[4] "Supplement to carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications: physical layer specification for 100 Mb/s operation on two pairs of Category 3 or better balanced twisted pair cable (100BASE-T2, Clause 32)", Standard IEEE 802.3y, IEEE, Mar. 1997.

[5] B. R. Petersen and D. D. Falconer, "Minimum mean-square equalization in cyclostationary and stationary interference–Analysis and subscriber line calculations", *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 931–940, Aug. 1991.

Appendix 19.A *Interference suppression*

Figure 19.18 illustrates the interference situations considered here. Equalization by linear forward filters only is assumed. Reception of 100BASE-T2 signals is disturbed either by alien NEXT from another synchronous 100BASE-T2 transmitter or by cross-talk from a single asynchronous source. Only one of these disturbances may be present. The symbol sequences $\{a_k^{A_R}\}$ and $\{a_k^{B_R}\}$ denote the sequences transmitted by the remote 100BASE-T2 transceiver, whereas $\{a_k^{A'_T}\}$ and $\{a_k^{B'_T}\}$ denote the sequences transmitted by an adjacent synchronous 100BASE-T2 transmitter. The spectrum $S(f)$ of the asynchronous source may be aperiodic or exhibit a period different from $1/T$. The functions "$H(f)$" represent the spectral responses of the signal or cross-talk paths from the respective sources to the inputs of the forward equalizer filters with transfer functions $C_{AA}(f)$ and $C_{BA}(f)$. Because of $2/T$ sampling rate, these functions exhibit $2/T$-periodicity. All signals and filter coefficients are real-valued. It is therefore sufficient to consider only frequencies f and $f - 1/T$, for $0 < f < 1/(2T)$. We will concentrate on the signals arriving at decision point DPA; the analysis for signals at DPB is similar.

Intersymbol-interference free reception of the symbol sequence $\{a_k^{A_R}\}$ and the suppression of signal components stemming from $\{a_k^{B_R}\}$ at DPA require

$$\begin{aligned} H_A^c(f)\, C_{AA}^c(f) + H_A^c\left(f - \frac{1}{T}\right) C_{AA}\left(f - \frac{1}{T}\right) &= 1 \\ H_B^c(f)\, C_{BA}^c(f) + H_B^c\left(f - \frac{1}{T}\right) C_{BA}\left(f - \frac{1}{T}\right) &= 0 \end{aligned} \tag{19.20}$$

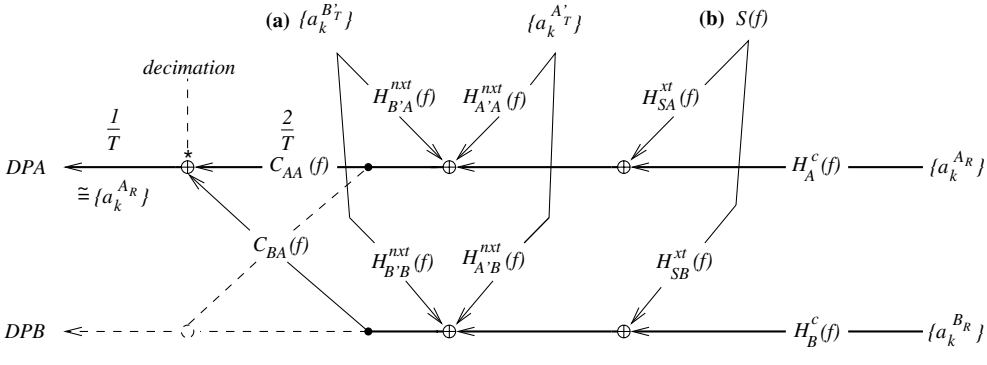


Figure 19.18. Cross-talk disturbance by: (a) alien NEXT from another synchronous 100BASE-T2 transmitter, (b) an asynchronous single source, for example, a 10BASE-T transmitter. [*From [2], © 1997 IEEE.*]

To suppress alien NEXT from a 100BASE-T2 transmitter, two additional conditions must be met:

$$\begin{aligned}\sum_{\ell=0,1}\left[H_{A'A}^{nxt}\left(f-\frac{\ell}{T}\right)C_{AA}\left(f-\frac{\ell}{T}\right)+H_{A'B}^{nxt}\left(f-\frac{\ell}{T}\right)C_{BA}\left(f-\frac{\ell}{T}\right)\right]=0\\ \sum_{\ell=0,1}\left[H_{B'A}^{nxt}\left(f-\frac{\ell}{T}\right)C_{AA}\left(f-\frac{\ell}{T}\right)+H_{B'B}^{nxt}\left(f-\frac{\ell}{T}\right)C_{BA}\left(f-\frac{\ell}{T}\right)\right]=0\end{aligned} \tag{19.21}$$

Alternatively, the additional conditions for the suppression of cross-talk caused by a single asynchronous source become

$$\begin{aligned}H_{SA}^{xt}(f)\,C_{AA}(f)+H_{SB}^{xt}(f)\,C_{BA}(f)=0\\ H_{SA}^{xt}\left(f-\frac{1}{T}\right)C_{AA}\left(f-\frac{1}{T}\right)+H_{SB}^{xt}\left(f-\frac{1}{T}\right)C_{BA}\left(f-\frac{1}{T}\right)=0\end{aligned} \tag{19.22}$$

Therefore in each case the interference is completely suppressed if for every frequency in the interval $0 < f < 1/(2T)$ the transfer function values $C_{AA}(f)$, $C_{AA}(f-(1/T))$, $C_{BA}(f)$ and $C_{BA}(f-(1/T))$ satisfy four linear equations. It will be highly unlikely that the cross-talk responses are such that the coefficient matrix of these equations becomes singular. Hence a solution will exist with high probability. In the absence of filter-length constraints, the $T/2$-spaced coefficients of these filters can be adjusted to achieve these transfer functions. For a practical implementation a trade-off between filter lengths and achieved interference suppression has to be made.

Algorithms for Communications Systems and Their Applications.
Nevio Benvenuto and Giovanni Cherubini
Copyright © 2002 John Wiley & Sons, Ltd. ISBN: 0-470-84389-6

Index

Access methods, 523
Active device, 257
Adaptation gain, 168, 170, 177, 183–185, 189, 212, 645, 703, 1086, 1100, 1104, 1120, 1124, 1250
Adaptive differential pulse code modulation (ADPCM), 341, 393, 398, 456, 1183
Advanced Mobile Phone Service (AMPS), 1180
Algorithm
  Bahl–Cocke–Jelinek–Raviv (BCJR), 668
  Benveniste–Goursat, 1091
  Contour (CA), 1093, 1095, 1098, 1102, 1113
  Delsarte–Genin, 110, 147
  Fano, 917–918
  forward backward (FBA), 670, 915, 930, 939, 941, 948
  Godard, 1101
  Jacobi, 158
  Linde–Buzo–Gray (LBG), 424
  least-mean-square (LMS), 173–179, 186–191, 205–216, 628, 633, 638, 648–649, 701, 722, 819, 1090, 1120, 1121, 1128, 1251, 1253, 1258, 1272
  Lempel–Ziv, 434
  Levinson–Durbin, 145
  Lloyd, 422
  Max, 369
  Mueller–Muller, 1064
  recursive least-squares (RLS), 197–203, 217, 645, 722, 1180
  Sato, 1090, 1096, 1100, 1106
  stochastic gradient (SGA), 1086, 1113
  stop-and-go, 1092
  Viterbi (VA), 663, 677, 682, 686, 691, 823, 915, 921–923, 968, 993, 1007, 1177, 1238, 1262
American National Standards Institute (ANSI), 1152
Analog-to-digital converter (ADC), 331, 338, 341, 385, 572–573, 1189–1191, 1249
Antenna
  array, 226
  directional, 296, 299, 811, 1165
  gain, 296
  isotropic, 296
Autocorrelation
  average, 1212
  matrix, 63–66, 132, 135, 149, 166, 420, 635, 719, 744
  sequence, 50, 53, 86, 144, 401, 622, 655, 678
Automatic gain control (AGC), 1249
Automatic repeat query (ARQ), 807, 827
Autoregressive model (AR), 91–94, 96–101, 101, 115, 143, 197, 414
Autoregressive moving average model (ARMA), 90, 94

Bandwidth
  definition of, 29
  excess, 764, 777, 1131, 1152, 1263, 1269

Bandwidth (*continued*)
minimum, 458, 552, 559, 578, 590, 609, 771, 808, 1003
Bit error probability, 340, 456, 475, 485, 494, 499, 578, 613, 899, 919, 941, 954, 1212, 1224
Bit loading, 1002, 1152
Blind equalization, 1083
Block codes
Bose–Chaudhuri–Hocquenghem (BCH), 878–898
generator matrix, 836
generator polynomial, 864
Hamming, 872
low-density parity check (LDPC), 946–955
non binary, 960–965
parity-check matrix, 833
Reed–Solomon, 885
simplex, 875–877
syndrome decoding, 839

Calculus of variations, 731, 1000
Carrier sense multiple access (CSMA), 1164
Cancellation
cross-talk, 1116, 1251
echo, 1116–1130, 1145, 1269
Cell radius, 1165, 1182
Cellular systems, 1171, 1182
Channel
additive white Gaussian noise (AWGN), 326, 439, 458, 503–506, 543, 713, 999,
binary symmetric (BSC), 456, 571
Channel model, 251, 296, 309, 313, 316, 322, 658, 811, 1116
Channel capacity, 503, 999
Cholesky decomposition, 155
Circular convolution, 21–23, 205, 242, 707
Code division multiple access (CDMA), 523, 795, 802, 810, 818, 1180
Code rate, 827, 828, 903, 1177, 1180, 1182, 1185

Codes
block, 827, 899, 946, 990
Bose–Chaudhuri–Hocquenghem (BCH), 878–898
channel, 827–965
concatenated, 921, 924
convolutional, 900–920
cyclic, 862–898
forward error correction (FEC), 827
linear, 830, 968
low-density parity check (LDPC), 946–955
Reed–Solomon, 885
turbo, 924–942
Walsh–Hadamard, 536
Coding
adaptive predictive (APC), 341, 401
adaptive transform (ATC), 341, 433
code excited linear predictive (CELP), 341, 416, 433, 578
linear predictive (LPC), 341, 414
residual excited linear predictive (RELP), 341, 415, 1226
subband (SBC), 341, 456
vector sum excited liner predictive (VSELP), 457, 1177
Coding by modelling, 413
Coherence
time, 311, 314
Combining
equal gain (EGC), 720
maximal ratio (MRC), 720, 815
optimum (OC), 721
selective, 719
switched, 720
Convolution, 13
Convolutional codes, 900–920
catastrophic error propagation, 910
decoding algorithms, 912–918
general description, 903
transfer function, 907
Correlation coefficient
partial (PARCOR), 146

Correlogram, 86
Coset, 837–841, 871, 964–965, 977
Costas loop, 1040–1043
Criterion
  least squares (LS), 148, 197, 429
  mean-square error (MSE), 166, 366
Cross-talk
  far-end (FEXT), 290, 1116
  near-end (NEXT), 288–289, 1116, 1251, 1269
Cut-off rate, 509
Cyclostationary process, 56, 1057

Decimation, 106, 236, 380, 766, 1126
Decoding
  iterative, 929, 939
  sequential, 958
Delay spread
  rms, 307, 1177
Delta modulation (DM), 343, 404
  adaptive, 343
  continuously variable slope (CVSDM), 411
  linear, 407
Demodulation
  coherent, 499
  non coherent, 487, 806, 1213
Despreading, 795, 801
Detection
  decision feedback sequence (DFSE), 695
  maximum likelihood (MLSD), 662, 969, 1262
  multiuser, 820, 823, 1131
  reduced state sequence (RSSE), 691–696, 716
  single-user, 818
  threshold, 462, 474, 542, 555
Diagram
  state, 902, 909, 910
  tree, 694, 901, 917, 974, 977
  trellis, 664, 667, 677, 682, 903, 921, 968, 971, 981, 986, 990, 1014
Differential PCM (DPCM), 343, 385, 407
Digital European Cordless Telephone (DECT), 523, 1145, 1162, 1182–1185
Digital signal processor (DSP), 709
Digital subscriber line (DSL), 285, 1131–1142, 1147
  asymmetric (ADSL), 1148–1150
  high bit rate (HDSL), 523, 1147–1148
  single line high speed (SHDSL), 919, 1148
  very high speed (VDSL), 285, 523, 1115, 1131–1142
Digital-to-analog converter (DAC), 113, 331, 339, 408, 413 572, 1234, 1251
Distance
  Euclidean, 7, 420, 499, 682, 915, 967
  free Euclidean, 919, 967, 986
  free Hamming, 907
  Hamming, 830, 875, 960
Distortion
  envelope delay, 319
Diversity
  frequency, 521, 1167, 1271
  polarization, 522
  space, 522, 1271
  time, 522
Division multiplexing
  frequency (FDM), 753, 1175
  time (TDM), 524, 1175
  wavelength (WDM), 292
Doppler
  frequency, 318
  shift, 303
  spectrum, 311, 313, 315
  spread, 304, 313, 1167
Downlink, 802, 819, 1156, 1175, 1183
Duplex transmission
  dual, 1154, 1263
  full, 522, 1117, 1145–1148, 1251, 1264
  half, 522, 1146, 1154
  mono, 1154, 1264, 1272

Duplexing
digital, 1149
frequency division (FDD), 523, 1146, 1175
time division (TDD), 522, 1184
Dynamic channel allocation (DCA), 1182
Dynamic channel selection (DCS), 1182

Echo cancellation, 225, 1116–1130
Envelope detector, 512
Equalization
adaptive, 304, 645, 1018, 1083, 1252
fractionally spaced, 630, 642, 1269
self-training, 1083
Equalizer
adaptive, 628, 1083, 1252
decision feedback (DFE), 595, 617, 635–645, 649–656, 695, 710, 741, 777, 823, 1010, 1131, 1270
decision feedback-zero forcing (DFE-ZF), 649, 657, 679, 695
fractionally spaced (FSE), 617, 630–634, 642, 699, 703, 1107, 1270
linear (LE), 594, 619–620, 627–628
linear zero forcing (LE-ZF), 619
passband, 697
Estimation error, 129, 151, 159, 165, 174, 200, 213, 241, 622, 741, 1064, 1105
Estimate
biased, 83
unbiased, 82
Estimator
early-late, 1055
feedback, 1053
phasor, 1066
timing, 1055
Ethernet, 290, 1152–1155, 1263
European Telecommunications Standards Institute (ETSI), 415, 435, 1137, 1148, 1163, 1185
Excess MSE, 854
Extension field, 890,
Eye diagram, 562

Fading channel
flat, 305, 311, 718
frequency selective, 305, 311, 718
Rayleigh, 308, 518
Rice, 308
FCC, 1162, 1264
Fiber-to-the-curb architecture (FTTC), 1148
Fiber distributed data interface (FDDI), 1155
Filter
allpass, 28
bank, 214, 433, 753–755, 757–773, 783, 818, 1152, 1190
Butterworth, 313
decimator, 110, 119–126, 628, 754–755
distributed arithmetic, 1121
finite impulse response (FIR), 25, 116, 129, 165, 399, 754, 1234, 1272
frequency response, 18, 543, 591, 760, 1028
highpass, 28
impulse response, 18, 542, 558, 754, 1005
infinite impulse response (IIR), 25, 90, 136, 317, 622
integrate and dump, 1197
interpolator, 112–126, 315, 754, 773, 1049, 1229
lattice, 146, 191, 204
loop, 1030, 1036, 1054, 1262
lowpass, 28
matched, 73, 462, 494, 567, 621, 698, 755, 799, 813, 1003, 1045, 1243
narrowband, 28
notch, 224
prediction error, 142, 386, 650, 679
transfer function, 16, 91
transversal, 165, 627, 1119–1120, 1255

whitened matched (WMF), 651, 1003, 1007
whitening (WF), 94, 651, 1005
Wiener, 129–140, 165, 622
Filter bank
critically sampled, 764–767, 769
non critically sampled, 764–769, 777
Finite field, 844
Finite state machine (FSM), 663, 751, 970
Flexible precoding, 1018–1025, 1146
Free space path loss, 298
Frequency deviation, 44, 453, 1197, 1207, 1214
Frequency division duplexing (FDD), 523, 1146, 1175
Frequency reuse, 810, 1165, 1172
Front-end architectures, 1189
Function
Bessel, 308, 511, 1056
Marcum, 529
saw-tooth, 1048

Galois field, 844, 851
Gauss quadrature rule, 607
Gaussian random process, 68, 309, 503, 518
Gaussian random variable, 67, 314, 439
Generator
matrix, 836, 865, 906
polynomial, 863, 869, 875, 905, 928
Geometric mean, 53, 1001
Global system for mobile communication (GSM), 523, 1172–1177
Gradient vector, 132, 150, 167, 1086
Gram–Schmidt orthonormalization procedure, 8
Granular
error, 350, 354, 364
noise, 389, 408, 411
Graph, 907, 947, 952
Gray coding, 462, 531
Group, 830, 961, 977
Guard time, 1176, 1184

Hard input decoding, 913
Heaviside conditions, 32, 259
Householder transformation, 158
Hybrid fiber/coax (HFC) networks, 1156–1160

IEEE 802 Working Group, 1152–1156, 1163–1164, 1266
Indoor environment, 1185
Inner product, 3
Integrated Services Digital Network (ISDN), 1119, 1147–1148, 1173
Intensity profile of multipath, 307
Interference
cancellation, 1115
co-channel (CCI), 803
intersymbol (ISI), 456, 557, 604, 620, 655, 674, 749
multiuser (MUI), 803, 817
Interleaving, 913
International Telecommunications Union (ITU), 435, 1146, 1148
Interpolation, 31, 109, 116–118, 339, 764, 1255, 1269

Jakes model, 313
Japanese Digital Cellular (JDC), 1171, 1180, 1235

Lagrange interpolation, 118
Lattice
Gosset, 977, 991
Schlaefli, 976, 991
Law
A, 363
μ, 364
Leakage, 86, 1190
Likelihood function, 511, 674, 1051
Likelihood ratio, 442, 675
Limiter-discriminator, 1197
Line coding, 583–601
Line codes
biphase, 584

Line codes (*continued*)
block, 585
dicode, 583, 591
duobinary, 591
modified duobinary, 592
NRZ, 583
RZ, 584
Line-of-sight (LOS), 295, 302
LMS algorithm
for lattice filters, 191
in a transformed domain, 211
leaky, 187, 634
normalized, 189
sign, 187
Local area network (LAN), 290, 523, 1152, 1155
Local multipoint distribution service (LMDS), 1165
Log-likelihood function, 512, 677
Log-likelihood ratio, 676, 916, 934

Matrix
circulant, 243, 776
diagonal, 20, 22, 155
generator, 836, 865, 906
Hadamard, 536
Hermitian, 65
inverse, 145
parity check, 833–836, 840, 865, 870, 905, 906, 946–950, 962–965
Toeplitz, 63
triangular, 214
unitary, 66, 168, 182, 212
Vandermonde, 890
Maximum a posteriori (MAP) criterion, 160, 441, 661, 675, 677, 915, 921, 932, 948
Max-Log-MAP, 675, 677, 916, 921
Log-MAP, 677, 916, 952
Mean
convergence of the, 178
Medium access control (MAC), 1156, 1163–1164
Message-passing decoding, 946, 953
Minimum function, 857–861

Modulation
amplitude (AM), 295, 461, 480, 539
amplitude and phase (AM-PM), 480
binary, 437, 487, 520
binary phase shift keying (BPSK), 450, 470, 477, 795, 919
biorthogonal, 493
carrierless AM/PM (CAP), 568–570
continuous phase (CPM), 1246–1248
continuous phase FSK (CPFSK), 1217, 1219
differential PSK (DPSK), 474–475
DMT, 770, 775, 781
double sideband (DSB), 41, 58, 608
DWMT, 782
FMT, 771, 777, 781
frequency (FM), 1197, 1213
frequency-shift keying (FSK), 452, 516, 519, 1207
Gaussian minimum-shift keying (GMSK), 1229–1243
index, 453, 1209
minimum-shift keying (MSK), 454, 1214–1228
multicarrier (MC), 753
offset QPSK (OQPSK), 1203, 1220
on-off keying (OOK), 510
orthogonal, 486
phase-shift keying (PSK), 465, 474, 995
pulse amplitude (PAM), 69, 72, 461–464, 539–543, 583, 1040
pulse duration (PDM), 464, 532, 534
pulse position (PPM), 464, 532
quadrature amplitude (QAM), 480–485, 502, 544–548, 611
quadrature PSK (QPSK), 472–473
single sideband (SSB), 58, 499
trellis coded (TCM), 967–998
vestigial sideband (VSB), 781
Moving average model (MA), 91, 94, 398

Multichannel multipoint distribution service (MMDS), 1165
Multipath, 299, 302, 307, 521, 717, 803, 811, 1167, 1243
Multiple access
  code division (CDMA), 524, 795, 802, 810, 818, 1180
  frequency division (FDMA), 523, 1178
  time division (TDMA), 524, 1158, 1175

Noble identities, 118, 759
Noise
  figure, 268, 270, 274
  impulse, 782, 1158
  shot, 265, 294, 326
  temperature, 265, 273
  thermal, 263–264
Norm, 3
Numerically controlled oscillator (NCO), 1054
Nyquist
  criterion, 559, 562, 589
  frequency, 462, 559, 616
  pulse, 559, 801

OFDM, 753–794
  passband, 780
  synchronization of, 779
  systems, 769, 773, 780, 1002, 1128
Optical fibers, 291–294, 1148
Orthogonality principle, 134, 151
Oversampling, 404, 627, 630, 699, 1062

Parseval theorem, 15, 84
Partial response systems, 587, 1096, 1249
Per survivor processing (PSP), 695
Periodogram, 84
  Welch, 85
Personal communication services (PCS), 810
Phase detector (PD), 1030
Phase deviation, 44, 322, 1193
Phase jitter, 321
Phase noise, 321, 697, 1189
Phase-locked loop (PLL), 326, 1029–1039, 1074, 1105, 1262
Pilot signal, 1027
Power amplifier (HPA), 322, 1171
Power back-off, 1136–1141
Power delay profile, 310, 812, 1243
Power spectral density (PSD), 46–62
Precoding, 596, 777, 599, 1008–1025
Prediction
  error, 141, 176, 192, 385, 399, 657
  linear, 129, 140–147, 414, 1180
  optimum, 143, 392
Predictor
  backward, 140
  forward, 140
  linear, 140, 385, 391
  optimum, 141, 399
Probability
  a posteriori, 160, 441, 661, 668, 931, 954
  a priori, 440, 669
  conditional, 159, 442, 510
  transition, 932
Probability density function
  Gaussian, 567
  Rayleigh, 308
  Rice, 308, 518
Probability of error, 454, 483, 489, 494, 899
Processing gain, 809
Projection operator, 153
Pseudo-inverse matrix, 155–157
Pseudo-noise (PN) sequences, 233–238
  CAZAC, 235
  Gold, 236
  maximal length, 233
Public switched telephone network (PSTN), 331, 1018, 1145
Pulse code modulation (PCM), 345, 357, 377, 385, 571

QPR-IV transceiver, 1249–1262
Quantization
  adaptive, 377

Quantization (*continued*)
error, 345, 347, 350, 352, 387, 404, 575
scalar, 417, 420, 424
uniform, 355
vector, 417–432
Quantizer
adaptive, 377–379, 381
non uniform, 358–377
uniform, 346, 353
vector, 418, 421
Quotient group, 978

Radio link, 294–318
Raised cosine pulse, 559–561
Rake receiver, 811, 815, 1181
Rate
bit, 331, 338, 340, 379, 406, 457, 504, 539, 578, 827–830, 1002
sampling, 407, 555, 616, 626, 769
symbol, 458, 540, 754
Receiver
direct conversion, 1190
superheterodyne, 1189
optimization, 731
Recovery
carrier frequency, 1068
carrier phase, 1050, 1104
timing, 1046, 1053, 1060, 1074, 1252, 1269
Reflection coefficient, 146, 278
Regenerative repeaters, 575–581

Sampling theorem, 8, 30, 337, 627, 1046
Scattering, 295, 300, 310
Scrambling, 803, 1153, 1265
Self-training equalization 1083–1114
Set partitioning, 694, 973, 1014
Shadowing, 313
Shannon limit, 499, 504, 508, 924, 947
Signal
analytic, 33, 38, 780, 804
bandwidth, 28–29
baseband, 31, 36, 459
baseband equivalent, 33–34, 549
complex envelope, 34
envelope, 43, 322
passband, 33, 697, 780
space, 1–4, 439, 990, 1008
Signal constellation, 12, 458
Signal-to-interference ratio (SIR), 798
Signal-to-noise ratio (SNR), 272, 449, 463, 469, 552, 626
Signal-to-quantization error ratio, 352, 354, 364, 387
Simplex transmission, 523
Simplex cyclic codes, 875–878
Singular-value decomposition (SVD), 155
Slope overload distortion, 389
Soft output Viterbi algorithm (SOVA), 921–924
Source coding, 433
Spectral efficiency, 458, 460, 463, 468, 503, 782, 991
Spectral factorization, 53
Spread spectrum systems, 795–826
applications of, 807
chip equalizer, 818
direct sequence, 795
frequency hopping, 804
symbol equalizer, 819
synchronization, 1074
Soft input decoding, 900, 914
Spreading
techniques, 795
factor, 795, 1074
Standard
ADSL, 1148
DECT, 1182
Ethernet, 1152, 1156, 1263–1273
FDDI, 1155
HDSL, 1147
HIPERLAN, 1185
IS-136, 1177
JDC, 1180
modem, 1146–1147
SHDSL, 1148
speech and audio, 434
Token Ring, 1155

VDSL, 1148
video, 435
Sufficient statistic, 440, 681, 816, 1007
Symbol error probability, 463, 469, 483, 508, 565, 604, 682
Synchronization, 1027–1082
Syndrome, 839–841, 871, 889, 895, 965
System
bandwidth, 265, 590, 1171
baseband equivalent, 549, 775
causal, 18
continuous time, 13
discrete time, 17, 556, 1003
identification, 239–254

T1 carrier, 524
Tanner graph, 947
girth, 952
Telecommunications Industry Association (TIA), 286
Telephone channel, 318–322, 697, 703
Token Ring, 1155
Tomlinson–Harashima precoding, 1009
Transform
D, 903
discrete cosine (DCT), 214–215, 433, 790
discrete Fourier (DFT), 19, 206, 214, 433
fast Fourier (FFT), 20
Fourier, 14–17
Hilbert, 37–38
z, 18
Transmission lines, 274–291
Trellis coded modulation (TCM), 967–997

Union bound, 455
Universal mobile telecomunication service (UMTS), 1170, 1181
Uplink, 803, 1156, 1158, 1175
User code, 796, 803

Vector quantization (VQ), 417–432
Vocoder, 341, 414
Voltage controlled oscillator (VCO), 1030, 1042, 1189, 1252

Wiener–Khintchine theorem, 46
Window
Hann, 82
raised cosine or Hamming, 82
rectangular, 78, 82, 239
triangular or Bartlett, 82
Window closing, 86
Wireless local area networks (WLANs), 1162–1163, 1170, 1185

Yule–Walker equations, 97

Zero-forcing equalizer
LE-ZF, 619, 648–649
DFE-ZF, 649, 651, 655