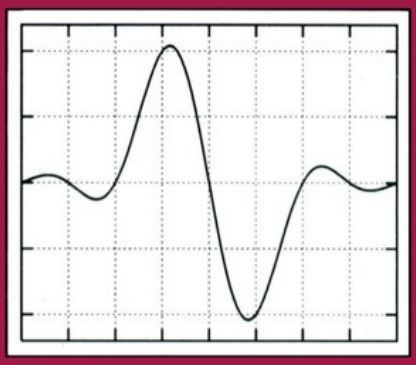
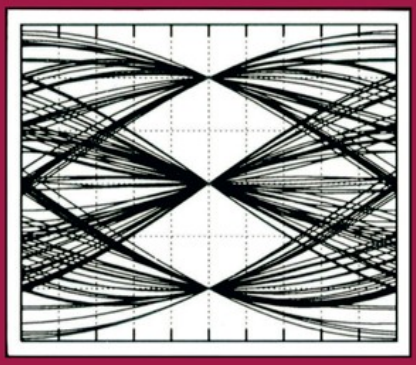
Digital Baseband Transmission and Recording



Jan W.M. Bergmans



Springer-Science+Business Media, B.V.

DIGITAL BASEBAND TRANSMISSION AND RECORDING

DIGITAL BASEBAND TRANSMISSION AND RECORDING

Jan W.M. Bergmans

Philips Research

Eindhoven, The Netherlands

SPRINGER-SCIENCE+BUSINESS MEDIA, B.V.

A C.I.P. Catalogue record for this book is available from the Library of Congress

ISBN 978-1-4419-5164-9 ISBN 978-1-4757-2471-4 (eBook)
DOI 10.1007/978-1-4757-2471-4

Printed on acid-free paper

All Rights Reserved
© 1996 Springer Science+Business Media Dordrecht
Originally published by Kluwer Academic Publishers in 1996
No part of the material protected by this copyright notice may be reproduced or
utilized in any form or by any means, electronic or mechanical,
including photocopying, recording or by any information storage and
retrieval system, without written permission from the copyright owner.

CONTENTS

PREFACE **xi**

1 MATHEMATICAL PRELIMINARIES **1**

1.1 Introduction 1
1.2 Deterministic signals and sequences 1
1.3 Stochastic signals and sequences 14
1.4 Probability density functions 25
1.5 Averages 29
1.6 Discrete-time minimum-phase functions 31
1.7 Lagrange multipliers 35
1.8 Further reading 37
APPENDIX 1A: The Fourier transform for continous-time signals 37
APPENDIX 1B: The Fourier transform for discrete-time signals 40
APPENDIX 1C: Spectral characteristics of minimum-phase functions 42
PROBLEMS 43
REFERENCES 46

2 EXAMPLES OF DIGITAL BASEBAND TRANSMISSION SYSTEMS **47**

2.1 Introduction 47
2.2 Digital subscriber lines 47
2.3 Digital magnetic recording 55
2.4 Digital optical recording 74
2.5 Further reading 87
APPENDIX 2A: Nonlinear bit-shifts due to bandwidth limitations of the write path 88
APPENDIX 2B: Nonlinear bit-shifts and write precompensation 90
APPENDIX 2C: Additive models for transition jitter 92
APPENDIX 2D: Nonlinear ISI induced by domain bloom 95
PROBLEMS 96
REFERENCES 98

3 CHARACTERISTICS OF DIGITAL BASEBAND TRANSMISSION **105**

3.1 Introduction 105

3.2 System model and nomenclature 106
3.3 Optimum modulation and demodulation for an ideal low-pass channel 115
3.4 Capacity of an ideal low-pass channel with white Gaussian noise 117
3.5 Capacity of a category of noisy dispersive channels 119
3.6 Effect of excess bandwidth 120
3.7 A hypothetical maximum-likelihood receiver 122
3.8 Noise whitening 125
3.9 Matched-filter bound 126
3.10 Equivalent discrete-time channel models 129
3.11 Performance of maximum-likelihood receivers 132
3.12 Further reading 137
APPENDIX 3A: Signal-to-noise ratios of NRZ and Bi-Phase on a Lorentzian channel 138
APPENDIX 3B: Optimum distribution of transmit power across frequency 142
APPENDIX 3C: Maximum-likelihood detection 143
APPENDIX 3D: Performance of the maximum-likelihood sequence detector in the presence of colored Gaussian noise 144
PROBLEMS 147
REFERENCES 150

4 BASEBAND MODULATION CODES 153
4.1 Introduction 153
4.2 Intersymbol interference, Nyquist-1 functions and eye patterns 155
4.3 Effect of modulation code on intersymbol interference 158
4.4 Effect of modulation code on signal-to-noise ratios 163
4.5 Effect of modulation code on timing recovery 166
4.6 Effect of modulation code on adaptation 170
4.7 Code mechanization 172
4.8 Run-length-limited codes 174
4.9 Modulation codes with spectral zeros 185
4.10 Linear modulation codes 195
4.11 Nonlinear codes with spectral zeros 201
4.12 Further reading 210
APPENDIX 4A: Analysis of baseline wander 211
APPENDIX 4B: Scramblers and pseudorandom sequences 212
APPENDIX 4C: Effect of coding on signal-to-noise ratios 215
PROBLEMS 220
REFERENCES 224

5 LINEAR EQUALIZATION 231
5.1 Introduction 231
5.2 Zero-forcing linear equalizer 232
5.3 ZF LE with minimum noise enhancement 237
5.4 Minimum mean-square error linear equalizer 240
5.5 Linear equalization for run-length-limited codes 246
5.6 Resistance to channel parameter variations 249
5.7 Transmit filters 250
5.8 Implementation issues 253
5.9 Further reading 257
APPENDIX 5A: Effect of bias on receiver performance 257
APPENDIX 5B: Structure of the MMSE transmit filter 259
PROBLEMS 260
REFERENCES 263

6 PARTIAL-RESPONSE AND DECISION-FEEDBACK EQUALIZATION 265
6.1 Introduction 265
6.2 Partial-response equalization 266
6.3 Decision-feedback equalization 277
6.4 Tomlinson/Harashima precoding 283
6.5 In-loop noise whitening 286
6.6 Decision-feedback equalization for run-length-limited codes 287
6.7 Further reading 291
APPENDIX 6A: Derivation of the discrete-time MMSE DFE 291
APPENDIX 6B: MMSE FSE and DFE of finite length 292
APPENDIX 6C: Optimum $d = 1$ DFE 294
PROBLEMS 295
REFERENCES 298

7 VITERBI DETECTION 301
7.1 Introduction 301
7.2 Dynamic programming 302
7.3 Two-state Viterbi detection 304
7.4 Effect of transmission code and channel on trellis diagram 306
7.5 Characteristics of Viterbi detection 311
7.6 Channel memory truncation by means of prefiltering 315
7.7 Sequence feedback 324
7.8 Set partitioning 331
7.9 Restricted-delay detection 335

7.10 Further reading 344
APPENDIX 7A: Viterbi detection for charge-constrained codes 344
APPENDIX 7B: Effects of a restricted detection delay 350
APPENDIX 7C: Performance of VD with prefilter 356
APPENDIX 7D: Performance of VD with sequence feedback 358
APPENDIX 7E: Exploitation of run-length constraints 360
PROBLEMS 363
REFERENCES 368

8 ADAPTIVE RECEPTION 373
8.1 Introduction 373
8.2 Structure of adaptive receivers 375
8.3 Automatic gain control 382
8.4 Non-data-aided adaptive equalization 398
8.5 Zero-forcing equalizer adaptation 400
8.6 Minimum mean-square error equalizer adaptation 412
8.7 Comparison of ZF and MMSE adaptation 420
8.8 Adaptive detection 421
8.9 Implementation example 427
8.10 Further reading 429
APPENDIX 8A: Analysis of the non-data-aided closed-loop AGC of Fig. 8.9 430
APPENDIX 8B: Effect of a nonlinear voltage-controlled amplifier on adaptation properties 432
APPENDIX 8C: Decoupled ZF adaptation 432
APPENDIX 8D: Behavior of single-tap ZF MMSI loop 433
APPENDIX 8E: Properties of the MMSE adaptation loop 435
PROBLEMS 438
REFERENCES 445

9 BASICS OF TIMING RECOVERY 451
9.1 Introduction 451
9.2 Timing-recovery schemes 452
9.3 Tracking 461
9.4 Acquisition 465
9.5 Aided acquisition 469
9.6 Acquisition limits 472
9.7 Further reading 476
APPENDIX 9A: All-digital timing-recovery scheme of Fig. 9.7 476

APPENDIX 9B: Phase- and frequency-locked loops for a sinusoid embedded in noise 482
APPENDIX 9C: Data-aided maximum-likelihood one-shot phase estimator 489
APPENDIX 9D: One-shot maximum-likelihood frequency estimator 492
PROBLEMS 494
REFERENCES 496

10 A CATALOG OF TIMING-RECOVERY SCHEMES 499
10.1 Introduction 499
10.2 Maximum-likelihood timing recovery 500
10.3 Timing recovery based on threshold crossings 508
10.4 Nonlinear spectral line methods 524
10.5 Early-late timing recovery 533
10.6 MMSE and sampled-derivative timing recovery 535
10.7 Zero-forcing timing recovery 544
APPENDIX 10A: Analysis of maximum-likelihood timing recovery 562
APPENDIX 10B: Analysis of timing recovery based on threshold crossings 568
APPENDIX 10C: Analysis of square-law timing recovery 571
APPENDIX 10D: Analysis of minimum mean-square error timing recovery 573
APPENDIX 10E: Analysis of zero-forcing timing recovery 575
PROBLEMS 577
REFERENCES 587

11 PHASE-LOCKED LOOPS 591
11.1 Introduction 591
11.2 PLL structure 591
11.3 Linear analysis of first-order PLL 594
11.4 Linear analysis of high-gain second-order PLL 602
11.5 Linear analysis of general second-order PLL 609
11.6 Nonlinear PLL behavior 610
11.7 Design example 612
11.8 Further reading 614
APPENDIX 11A: Equivalent discrete-time model of cascade of analog loop filter and VCO 615
APPENDIX 11B: Effect of loop delay on stability 618
PROBLEMS 620
REFERENCES 623

INDEX 625

PREFACE

The advent of the information age has spurred the development of communication technologies for digital transmission and storage of voice, audio, video and data signals. Application areas of these technologies range in bit rate from bits per second to gigabits per second, and span such diverse media as radio links, telephone lines, coaxial cable, magnetic disk and tape, optical disc, and optical fiber. In all of these systems, data may or may not be modulated onto a carrier wave before being transmitted or stored. The adjectives 'carrier-modulated' and 'baseband' are commonly used to distinguish between these two possibilities. Digital baseband transmission is used, for example, in narrowband ISDN, in optical fiber transmission, and in virtually all digital storage systems.[1] Key ingredients are modulation coding, equalization, detection, adaptation and timing recovery. It is the objective of this book to give a coherent, in-depth and up-to-date overview of these and related ingredients. The book is an outgrowth of an internal course at Philips Electronics, Eindhoven, The Netherlands. It is considerably broader in scope than books that focus primarily on a single transmission ingredient, like [6], [31], [21], [18], [10], [35], [16], [23], [28] and [24]. Even so the treatment extends beyond that of classical textbooks on digital communications, like [22], [25], [32], [1], [17], [19] and [15]. Special emphasis is given to transmission impairments that are characteristic of digital baseband transmission, such as linear and nonlinear intersymbol interference, random jitter of recorded transitions, and dynamic fluctuations of the time base and the characteristics of the transmission channel.

To the best of my knowledge this is the first book to deal exclusively with digital baseband transmission, and a considerable fraction of the presented material has not appeared in book form before. Some material (notably parts of Chapters 4, 7, 8 and 9) appears here in print for the first time. Even so there is inevitably a certain overlap with the many excellent text books on information theory, coding, adaptation, synchronization and digital communication, like [1]-[37]. The reader is likely to benefit from prior exposure to one or more of these works. Conversely, much of the material presented here is probably also of interest to the 'carrier-modulated' engineering community.

An attempt has been made to make the material accessible to a wide audience. To this end, the development of the subject matter is almost entirely self-contained, and mathematical formalism is de-emphasized wherever possible in favour of an exposition of the underlying concepts. The reader is assumed to have an elementary background in mathematics and signal analysis, roughly at the level of an undergraduate engineering curriculum. Required knowledge beyond this level is summarized in Chapter 1 and concerns mainly certain aspects of spectral analysis. The presented material should be

[1] Since storage is merely the transportation of information from one point in time to another, it is, in essence, also a form of transmission.

of interest to students in electrical engineering as well as to practicing engineers. The entire book can be covered in a two-semester course. For a one-semester course it will be necessary to skip parts of most chapters in accordance with their relative importance.

Some subjects had to be skipped in order to restrict the size of the book to reasonable limits. Omitted subjects include channel characterization techniques, servo signal processing, word synchronization, error protection, and implementation issues. It is hard to over-estimate the practical importance of the latter topic and we would be remiss not to refer the reader to such excellent treatises as [15, Section 10.1] and [38, Chapter 10] (both on digital implementation) as well as [39], [40] and [41] (on analog implementation). References to these and other omitted topics are listed in the 'Further Reading' sections of each chapter. The reference lists are extensive and are as much as possible up to date.

Chapter 1 discusses the mathematical tools and concepts that are basic to later chapters. Covered are the temporal and spectral effects of filtering, sampling and modulation on both deterministic and random signals; probability density functions; the arithmetic, geometric and harmonic averages of amplitude spectra; properties of discrete-time minimum-phase functions; and constrained optimization via Lagrange multipliers. Prior knowledge that is assumed for this chapter as well as for the rest of the book is an elementary familiarity with calculus, linear systems theory, Fourier analysis, and probability and stochastic processes.

By way of introduction and motivation, Chapter 2 presents brief overviews of three application areas of digital baseband transmission, namely digital subscriber lines, digital magnetic recording, and digital optical recording. The treatment of transmission techniques in later chapters is somewhat biased towards these three areas, and in particular towards digital recording. Appendices of the chapter are devoted to various types of bit-shifts that arise in the recording process.

Chapter 3 outlines the basic topology of a digital baseband transmission system, introduces elementary notions and terminology, and discusses several fundamental limits to transmission performance, such as channel capacity, the matched-filter bound, and the performance of the maximum-likelihood sequence detector.

Chapter 4 is devoted to baseband modulation codes. It discusses their *raison d'être* and their impact on such transmission aspects as intersymbol interference, signal-to-noise ratios, timing recovery, and adaptation. It also discusses two important categories of codes, namely run-length-limited codes and codes that introduce one or more spectal zeros. The latter category is subdivided into linear and nonlinear codes. This distinction is not often made but has important consequences for receiver design and transmission performance. Fundamentals of these categories are described and for each of them a small catalog of worked examples is presented.

Chapters 5 and 6 deal with equalization. Equalization is a bare necessity -as opposed to a minor luxury- in many baseband transmission systems, and the depth and scope of our treatment reflect this importance. Chapter 5 is concerned with linear equalization subject to both zero-forcing and minimum mean-square error optimality criteria. Also addressed are the equalization of run-length-limited codes, the design of transmit filters, optimization of equalizer resistance to channel parameter variations,

and implementation-related issues. Chapter 6 provides a similar coverage of partial-response and decision-feedback equalization. In both chapters, optimum equalizer performances are described as spectral averages of the folded signal-to-noise ratio of the channel and the spectrum of the encoded data. These canonical descriptions provide direct insight into performances of the equalizers *vis a vis* each other and the matched-filter bound, and into the impact of the modulation code on attainable performances.

Recent years have witnessed a rapidly growing interest in near-optimum reception techniques. Many of these are based in some way or another on the Viterbi detector (VD). Chapter 7 attempts to provide a cohesive overview of these techniques. After developing the basic form of the VD and exploring some of its characteristics, it discusses and illustrates four approaches that are aimed at lowering VD complexity at a minor expense to performance. These approaches are based on prefiltering, sequence feedback, set partitioning, and restriction of the detection delay, respectively. They are largely complementary in nature and are often quite effective. Among the presented examples is Ferguson's detector for partial-response systems, which is widely used in digital magnetic recording. The VD first found its way into data transmission systems as an effective technique for dealing with intersymbol interference, and Chapter 7 focuses primarily on this objective. A more recent and expanded objective involves exploitation of any redundancy that is introduced by the modulation code. This topic is discussed in a more restricted fashion, with emphasis on run-length-limited codes and codes that introduce a zero at DC.

Many baseband transmission channels exhibit temporal or piece-by-piece variations that must be dealt with adaptively in the data receiver. Chapter 8 provides an overview of these adaptation techniques. Emphasis is on such 'down-to-earth' topics as automatic gain and slope control, adaptive equalization with digital as well as analog equalizer structures, adaptation of various detector parameters, and more generally on adaptation techniques with a maximum simplicity. Instead of moving beyond the classical LMS adaptation algorithm into the realm of 'advanced' adaptation schemes, the chapter provides an in-depth treatment of zero-forcing (ZF) adaptation, which is often preferred to LMS because of its simplicity.

Joint adaptation of multiple parameters is most conveniently described in terms of matrix notation and some parts of Chapter 8, unlike all other chapters in this book, assume that the reader has an elementary knowledge of matrix theory.

The last three chapters of the book are concerned with timing recovery. The purpose of timing recovery is to define the instants at which decisions are taken. This task becomes more difficult and at the same time more critical to transmission performance as the system becomes more efficient in terms, for example, of modulation code or bandwidth. This is one reason why classical schemes, in which the timing is derived directly from the received signal, are gradually being replaced by more powerful schemes of the data-aided type. The timing-recovery scheme usually takes the form of a delay-locked or phase-locked loop and at the beginning of transmission this loop must be brought into lock. In many instances the time for the loop to acquire lock is an undesirable overhead that should be minimized. Various techniques have evolved to accomplish this.

The importance of timing recovery seems to warrant the extensive treatment in this book. Chapter 9 is concerned with basics. It presents a classification of timing-recovery schemes, discusses the impact of various system parameters on timing recovery, presents fundamental limits on the speed of acquisition, and discusses several approaches to improve acquisition speed of a 'bare' scheme towards these limits.

Chapter 10 presents a rather extensive catalog of schemes. The catalog is subdivided into several categories of schemes and to establish a baseline it first covers the optimum approach, called maximum-likelihood timing recovery. Further categories are presented in a more or less chronological order, progressing from classical ones to data-aided categories of the minimum mean-square error and zero-forcing type. The latter two categories are presented in such a manner as to expose their close relation to the adaptation techniques of Chapter 8. The catalog focuses on what sets distinct schemes apart, namely the timing-error detector which serves to extract timing-error information from the incoming signal. The common trait of most schemes is that they take the form of a discrete-time phase-locked loop (PLL). This trait is studied in Chapter 11. This final chapter discusses basic properties of the discrete-time PLL and relates them to those of the more usual continuous-time PLL.

Throughout the book, examples serve the dual purpose of illustrating the principal results and connecting the theory to practical applications. In all chapters the emphasis is on concepts rather than mathematics. For this reason some of the more mathematical subjects have been relegated to appendices. Each chapter contains a problem section that is meant to deepen and expand the readers understanding of the subject matter. These sections are best regarded as an integral portion of the text.

The present book would not have been written without my long-term association with Philips Research, Eindhoven, The Netherlands, and without the influence and help of many friends and colleagues. In particular I would like to thank Bob Barnes, Steve Brittenham, Theo Claasen, Kevin Fisher, Charlie Gamble, Piet van Gerwen, Henk Hollmann, Morishi Izumita, Guy Kerpen, Seiichi Mita, Saeed Rajput, Hans Voorman, Frans Willems, and HoWai Wong-Lam for interactions which have inspired parts of the material presented here, and Ad van den Enden, Henk Hollmann, Nuno Ramalho, Hugo Veenstra, Niek Verhoeckx, Jan van Vlerken, HoWai Wong-Lam and Peter van der Wurf for reviewing parts of the text. Also, I would like to express my gratitude to the management of Philips Research for giving me the opportunity and permission to publish this book. Finally, I would like to thank my Polish treasure for her support and endurance.

Jan W.M. Bergmans,

Warsaw, August 1995.

REFERENCES

[1] S. Benedetto, E. Biglieri, and V. Castellani, *Digital Transmission Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[2] W.R. Bennett and J.R. Davey, *Data Transmission*. New York: McGraw-Hill, 1965.

[3] J.A.C. Bingham, *Theory and Practice of Modem Design*. New York: Wiley, 1988.

[4] R.E. Blahut, *Principles and Practice of Information Theory*. Menlo Park, CA: Addison-Wesley, 1987.

[5] R.E. Blahut, *Digital Transmission of Information*. Menlo Park, CA: Addison-Wesley, 1990.

[6] K.W. Cattermole, *Principles of Pulse Code Modulation*. London: Iliffe Books, 1969.

[7] A.P. Clark, *Principles of Digital Data Transmission* (2nd ed.). New York: Halstead Press - Wiley, 1983.

[8] A.P. Clark, *Adaptive Detectors for Digital Modems*. London: Pentech Press, 1989.

[9] T.M. Cover and J.A. Thomas, *Elements of Information Theory*. New York: John Wiley, 1991.

[10] C.F.N. Cowan and P.M. Grant, *Adaptive Filters*. Englewood Cliffs, New Jersey: Prentice-Hall, 1985.

[11] K. Feher, *Digital Communications*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[12] K. Feher (ed.), *Advanced Digital Communication Systems and Signal Processing Techniques*. Englewood Cliffs, N.J.: Prentice-Hall, 1987.

[13] L.E. Franks, *Signal Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1969.

[14] R.G. Gallager, *Information Theory and Reliable Communication*. New York: John Wiley & Sons, 1968.

[15] R.D. Gitlin, S.B. Weinstein, and J.F. Hayes, *Data Communication Principles*. New York: Plenum, 1991.

[16] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1986.

[17] S. Haykin, *Digital Communications*. New York: Wiley, 1988.

[18] M.L. Honig and D.G. Messerschmitt, *Adaptive Filters: Structures, Algorithms, and Applications*. Boston: Kluwer Academic Publishers, 1984.

[19] E.A. Lee and D.G. Messerschmitt, *Digital Communications*. Boston, MA: Kluwer, 1988.

[20] Y.W. Lee, *Statistical Theory of Communication*. New York: Wiley, 1960.

[21] W.C. Lindsey, *Synchronization Systems in Communications*. Englewood Cliffs, NJ: Prentice-Hall, 1972.

[22] R.W. Lucky, J. Salz, and E.J. Weldon, Jr., *Principles of Data Communication*. New York: McGraw-Hill, 1968.

[23] H. Meyr and G. Ascheid, *Synchronization in Digital Communications - Volume I*. New York: Wiley, 1990.

[24] J. Moon and L.R. Carley, *Sequence Detection for High-Density Storage Channels*. Dordrecht: Kluwer Academic Publishers, 1992.

[25] J.G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1983 (2nd ed. 1989).

[26] J.G. Proakis and J.A. Salehi, *Communication Systems Engineering*. Englewood Cliffs, NJ: Prentice-Hall, 1994.

[27] M.S. Roden, *Digital and Data Communication Systems*. Englewood-Cliffs, NJ: Prentice-Hall, 1982.

[28] K.A. Schouhamer Immink, *Coding Techniques for Digital Recorders*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[29] M. Schwartz, *Information Transmission, Modulation, and Noise*. New York: McGraw-Hill, 1959 (4th ed. 1990).

[30] K.S. Shanmugan, *Digital and Analog Communication Systems*. New York: Wiley, 1979.

[31] J.J. Stiffler, *Theory of Synchronous Communications*. Englewood Cliffs, NJ: Prentice-Hall, 1971.

[32] H. Taub and D.L. Schilling, *Principles of Communication Systems*. New York: McGraw-Hill, 1986.

[33] A.J. Viterbi, *Principles of Coherent Communications*. New York: McGraw-Hill, 1966.

[34] A.J. Viterbi and J.K. Omura, *Principles of Digital Communication and Coding*. New York: McGraw-Hill, 1979.

[35] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, New Jersey: Prentice-Hall, 1985.

[36] J.M. Wozencraft and I.M. Jacobs, *Principles of Communication Engineering*. New York: Wiley, 1965.

[37] R.E. Ziemer and W. Peterson, *Digital Communications and Spread Spectrum*. New York: MacMillan, 1985.

[38] A.W.M. van den Enden and N.A.M. Verhoeckx, *Discrete-Time Signal Processing - An Introduction*. Hemel Hempstead, Hertfordshire (UK): Prentice-Hall, 1989.

[39] P.R. Gray and R.G. Meyer, *Analysis and Design of Analog Integrated Circuits*. New York: John Wiley & Sons, 1977 (2nd ed. 1984).

[40] R. Gregorian and G.C. Temes, *Analog MOS Integrated Circuits for Signal Processing*. New York: John Wiley & Sons, 1986.

[41] Y.P. Tsividis and J.O. Voorman (Eds.), *Integrated Continuous-Time Filters*. New York: IEEE Press, 1993.

1

MATHEMATICAL PRELIMINARIES

1.1 INTRODUCTION

Digital transmission and recording systems all convey discrete-time data sequences across channels with a continuous-time nature. Conversions between the discrete-time and continuous-time domains are key operations in such systems, as are linear filtering operations in both domains. It is, therefore, appropriate that we set out with an introduction to some of the mathematical concepts and tools that are central to all of these transformations. In doing so we will assume that the reader has a basic working knowledge of the theory of signal analysis in both domains. For this reason we recapitulate this theory only cursorily, aiming in part to familiarize the reader with our notation. The borderline between both domains, on the other hand, is relatively under-exposed in most textbooks on signal analysis and will therefore receive more detailed attention. Section 1.2 is concerned with deterministic signals and sequences, with an emphasis on the spectral effects of linear filtering, sampling and linear modulation. A similar account is given in Section 1.3 for stochastic sequences and signals. The remaining sections are concerned with various other subjects that are of interest to later chapters. These are probability density functions (Section 1.4), the arithmetic, geometric and harmonic averages of spectra (Section 1.5), properties of discrete-time minimum-phase functions (Section 1.6), and the method of Lagrange multipliers (Section 1.7). Appendices 1A and 1B provide a brief survey of properties of the Fourier transforms of continuous-time signals and discrete-time sequences, along with a summary of basic Fourier transform pairs.

1.2 DETERMINISTIC SIGNALS AND SEQUENCES

1.2.1 Fourier transform definitions

Let $h(t)$ be a continuous-time signal where t denotes time in seconds, and denote by E_h the energy of $h(t)$, i.e. $E_h = \int_{-\infty}^{\infty} h^2(t)dt$. If E_h is finite then $h(t)$ has a well-defined

Fourier transform

$$H(\omega) \triangleq \int_{-\infty}^{\infty} h(t)e^{-j\omega t}dt \qquad \text{(FTC)}. \tag{1.1}$$

This is a complex-valued function of the angular frequency ω. We refer to (1.1) as the Fourier transform for continuous-time signals (FTC).

Let f_k be a discrete-time sequence of sampling rate $1/T$. Individual symbols f_k are typically represented as numbers on a digital computer. For this reason it makes little sense to assign any physical dimension to f_k. Even so, we can assign to f_k a dimensionless energy $E_f \triangleq \sum_{k=-\infty}^{\infty} f_k^2$. Just as for the physical signal $h(t)$, f_k will have a well-defined Fourier transform provided that its energy is finite. The Fourier transform for discrete-time sequences (FTD) of f_k is defined as

$$F(e^{j\omega T}) \triangleq \sum_{k=-\infty}^{\infty} f_k e^{-jk\omega T} \qquad \text{(FTD)}. \tag{1.2}$$

Instead of $e^{j\omega T}$ we could also have used the argument ω for F. The argument $e^{j\omega T}$ is somewhat more complicated but has the advantage that it continuously reminds us of the fact that F is a periodic function of ω with period $\omega_0 \triangleq 2\pi/T$. Inverse Fourier transformation allows $h(t)$ and f_k to be recovered according to

$$h(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} H(\omega)e^{j\omega t}d\omega \qquad \text{(IFTC)} \tag{1.3}$$

and

$$f_k = \frac{1}{\omega_0}\int_{-\omega_0/2}^{\omega_0/2} F(e^{j\omega T})e^{jk\omega T}d\omega \qquad \text{(IFTD)}\,. \tag{1.4}$$

The functions $h(t)$ and $H(\omega)$ form a Fourier transform pair, and so do f_k and $F(e^{j\omega T})$. We represent this symbolically by

$$h(t) \circ\!\!-\!\!\circ H(\omega) \qquad \text{and} \qquad f_k \circ\!\!-\!\!\circ F(e^{j\omega T}). \tag{1.5}$$

Appendices 1A and 1B summarize elementary properties of both Fourier transforms and list basic Fourier transform pairs.

While several text books on signal processing use the angular frequency ω as the argument of the Fourier transform (see, for example, [10]), several other books use frequencies f in Hertz. Conversion between f and ω is as simple as $\omega = 2\pi f$ and the choice of argument is, therefore, mainly a matter of taste. Appendix 1A is entirely cast in terms of f and the reader may wish to compare the above FTC and IFTC definitions with those of the Appendix. In forthcoming chapters we will often find it convenient to normalize frequencies relative to a given symbol rate $1/T$. A normalized frequency $\Omega = 1$ indicates a frequency equal to $1/T$, i.e.

$$\Omega \triangleq fT = \frac{\omega T}{2\pi}. \tag{1.6}$$

Exercise 1.1 Show that FTC and IFTC may be recast in terms of normalized frequencies as

$$H(\Omega) \triangleq \int_{-\infty}^{\infty} h(t)e^{-j2\pi\Omega t/T}dt \quad \text{(FTC)} \tag{1.7}$$

and

$$h(t) = \frac{1}{T}\int_{-\infty}^{\infty} H(\Omega)e^{j2\pi\Omega t/T}d\Omega \quad \text{(IFTC)}. \tag{1.8}$$

In similar vein, the angular frequency ω in the FTD and IFTD can be replaced by a frequency f in Hertz or a normalized frequency $\Omega = fT = \omega T/(2\pi)$. Appendix 1B is entirely cast in terms of Ω.

Exercise 1.2 Show that FTD and IFTD may be recast in terms of normalized frequencies as

$$F(e^{j2\pi\Omega}) \triangleq \sum_{k=-\infty}^{\infty} f_k e^{-j2\pi\Omega k} \quad \text{(FTD)} \tag{1.9}$$

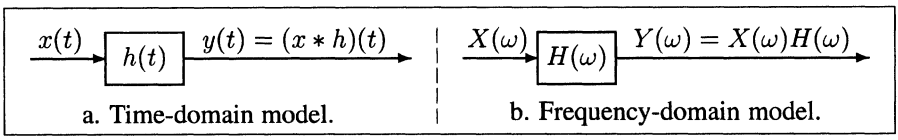
and

$$f_k = \int_{-0.5}^{0.5} F(e^{j2\pi\Omega})e^{j2\pi\Omega k}d\Omega \quad \text{(IFTD)} \tag{1.10}$$

1.2.2 Continuous-time filtering

Linear filtering operations are basic to digital transmission and occur both in the discrete-time and in the continuous-time domain. An operation is said to be *linear* if an input signal ax_1+bx_2 produces an output ay_1+by_2 where a and b are arbitrary constants. Here x_1 and x_2 are arbitrary filter inputs (continuous-time for a continuous-time filter and discrete-time for a discrete-time filter), and y_1 and y_2 are the corresponding outputs.

In the chapters ahead we will be mainly concerned with *time-invariant* filters.[1] Such filters are, crudely speaking, characterized in that the manner in which they respond to their input does not change over time. Thus, if the input of a time-invariant filter is delayed by a certain amount then the output is delayed by the same amount but does not otherwise change. We first consider the continuous-time case.



a. Time-domain model. b. Frequency-domain model.

Figure 1.1: Continuous-time linear filter.

In Fig. 1.1.a, a continuous-time signal $x(t)$ is applied to a linear time-invariant filter with *impulse response* $h(t)$. The filtered signal $y(t)$ is the linear convolution of $x(t)$ and

[1] An exception is formed by Chapter 8. That chapter deals with adaptive filters, which have a slowly time-varying nature.

$h(t)$. This convolution is defined as

$$y(t) \triangleq \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau. \tag{1.11}$$

In the sequel we denote linear convolution by the symbol '$*$', and adopt the notation $y(t) = (x * h)(t)$ as a compact equivalent for (1.11). The term 'impulse response' for $h(t)$ originates from the fact that $h(t)$ will appear at the output of the filter in response to an impulse at $t = 0$ in the form of Dirac's delta function $\delta(t)$. This function represents an infinitely narrow and infinitely high impulse of unit area at $t = 0$, and may be defined as $\delta(t) = \lim_{\varepsilon \to 0} f_\varepsilon(t)$ where

$$f_\varepsilon(t) \triangleq \begin{cases} \frac{1}{2\varepsilon}, & |t| < \varepsilon, \\ 0, & \text{else.} \end{cases} \tag{1.12}$$

Dirac's delta function has the *sieve property*

$$\int_{-\infty}^{\infty} x(t)\delta(t)dt = x(0) \tag{1.13}$$

for any function $x(t)$ that is continuous in $t = 0$. Also

$$\delta(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{j\omega t}d\omega. \tag{1.14}$$

This property leads directly to the FTC. A discussion of these and other properties may be found in [10, App. 3-C].

Exercise 1.3 Use the sieve property (1.13) to confirm that $h(t)$ will appear at the output of the filter in response to $\delta(t)$.

A filter is said to be *stable* if it responds to any input signal $x(t)$ of finite amplitude by an output $y(t)$ of finite amplitude. It is said to be *causal* when its response at any instant t is fully determined by the present and past values of the input signal. A stable filter has $\int_{-\infty}^{\infty} |h(t)|dt < \infty$. A causal filter has $h(t) = 0$ for all $t < 0$.

Transfer function The spectral effect of a convolution is a multiplication:

$$Y(\omega) = X(\omega)H(\omega) \text{ for all } \omega \tag{1.15}$$

(see Fig. 1.1.b). Here $Y(\omega)$, $X(\omega)$ and $H(\omega)$ denote the Fourier integrals of $y(t)$, $x(t)$ and $h(t)$, respectively. One commonly calls $H(\omega)$ the *transfer function* of the filter. Since $H(\omega)$ is a complex-valued function of frequency, it can also be written as $H(\omega) = A(\omega)\exp[j\phi(\omega)]$ where $A(\omega) \triangleq |H(\omega)|$ is the *transfer magnitude* or *amplitude characteristic* of the filter and

$$\phi(\omega) \triangleq \arctan\frac{\text{Im}[H(\omega)]}{\text{Re}[H(\omega)]} \tag{1.16}$$

is the *phase characteristic*. The *group delay characteristic* is uniquely determined by the phase characteristic according to

$$\tau(\omega) \triangleq -\frac{d\phi(\omega)}{d\omega}. \tag{1.17}$$

The group delay characterizes, coarsely speaking, the delay in seconds that an incoming signal of frequency ω experiences when propagating through the filter. This interpretation should not be taken too literally. For example, while the group delay of a causal filter is usually positive it can become negative across a restricted band of frequencies, and a negative delay in the physical sense is clearly precluded by the causality of the filter (see also Problems 1.1 and 1.2).

Poles and zeros The transfer function of continuous-time filters is sometimes expressed in terms of the complex-valued Laplace variable p. A treatment of the Laplace transform is beyond the scope of this text and may be found, for example, in [7] and [16]. For our purposes it suffices to know that the choice $p = j\omega$ reduces the Laplace transform to a Fourier transform. Realizable analog filters have a transfer function of the form

$$H(p) = \frac{b_0 + b_1 p + \ldots + b_N p^N}{1 - a_1 p - \ldots - a_M p^M} \tag{1.18}$$

where $b_0, \ldots, b_N$ and $a_1, \ldots, a_M$ are arbitrary real-valued coefficients and M and N are nonnegative integers such that $N \leq M$. The filter is said to have *order* M. We can rewrite (1.18) as

$$H(p) = -\frac{b_N}{a_M} \frac{\prod_{i=1}^{N}(p - z_i)}{\prod_{j=1}^{M}(p - p_j)}. \tag{1.19}$$

The complex-valued quantities z_i and p_j are called the *zeros* and *poles* of $H(p)$, respectively. Zeros and poles are either real-valued or occur in complex conjugate pairs. They are important for several reasons.

1. Except for a scaling factor the pole/zero constellation fully determines $H(p)$.

2. For a filter to be stable, all poles must be in the left half of the p-plane, i.e. $\mathrm{Re}(p_j)$ must be negative for all $j \in \{1, \ldots, M\}$.

3. For a filter to have minimum phase, all zeros must be in the left half of the p-plane, i.e. $\mathrm{Re}(z_i)$ must be negative for all $i \in \{1, \ldots, N\}$. (A filter is said to have minimum phase if, given the amplitude characteristic, the group delay $\tau(\omega)$ has the lowest possible value for *all* frequencies.)

4. For a filter to be a stable 'all-pass' network, all poles must be in the left half of the p-plane and all zeros in the right half. Furthermore, poles and zeros must occur in

pairs that have mirror symmetry about the imaginary axis. (An all-pass network, also called phase shifter, is characterized in that $|H| = 1$ for all ω.)

5. Pole-zero plots of filters connected in cascade can be overlaid on top of each other. If a pole and a zero coincide, they cancel one another. By implication, a high-order filter can be realized as a cascade of first-order and second-order sections. The second-order sections realize the complex conjugate pairs of poles and zeros, while the first-order sections realize real-valued poles and zeros.

6. The amplitude and phase characteristics of a filter can be estimated literally at a glance from the pole/zero constellation. By way of illustration Fig. 1.2 depicts the pole-zero plot of a second-order filter. Crosses represent poles and circles represent zeros.

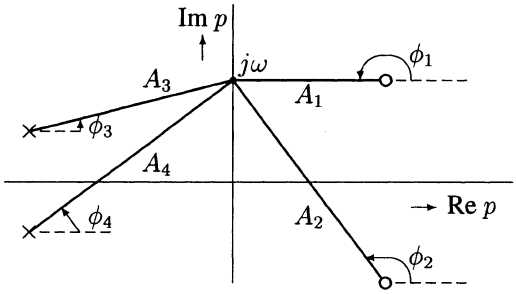


Figure 1.2: Graphical determination of amplitude and phase characteristics for a second-order filter.

For simplicity we take $b_N/a_M = -1$. With (1.19) we then have $A(\omega) = A_1A_2/(A_3A_4)$ and $\phi(\omega) = \phi_1 + \phi_2 - \phi_3 - \phi_4$. Each pole and zero, therefore, has the largest effect in a frequency range that corresponds to a part of the vertical axis nearest to that pole or zero.

In data receivers, analog filters often serve as a prefilter for noise suppression or as part of the equalizer. Properties, design and VLSI realization of analog filters are consequently of considerable interest to digital transmission. We refer to [17] and to the excellent survey of Tsividis and Voorman [13] for further reading in this context.

1.2.3 Discrete-time filtering

In Fig. 1.3.a, a discrete-time sequence a_k is applied to a linear filter with impulse response f_k. The filtered sequence b_k is the linear convolution of a_k and f_k:

$$b_k \triangleq \sum_{i=-\infty}^{\infty} a_i f_{k-i}. \tag{1.20}$$

We denote this compactly as $b_k = (a * f)_k$. The term 'impulse response' for f_k originates from the fact that the filter will respond with output f_k to an impulse in the form

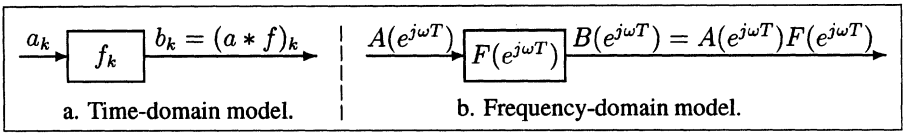


Figure 1.3: Discrete-time filter.

of the Kronecker delta function δ_k. This function is defined by

$$\delta_k \triangleq \begin{cases} 1, & k = 0, \\ 0, & \text{else.} \end{cases} \tag{1.21}$$

By analogy to the continuous-time case, a *stable* discete-time filter has $\sum_{k=-\infty}^{\infty} |f_k| < \infty$, and a *causal* one has $f_k = 0$ for $k < 0$.

Transfer function In frequency-domain notation we may write

$$B(e^{j\omega T}) = A(e^{j\omega T})F(e^{j\omega T}) \tag{1.22}$$

where $B(e^{j\omega T})$, $A(e^{j\omega T})$ and $F(e^{j\omega T})$ denote the discrete-time Fourier transforms of b_k, a_k and f_k, respectively (see Fig. 1.3.b). We call $F(e^{j\omega T})$ the transfer function of the filter. Just as for continuous-time filters, $F(e^{j\omega T})$ determines the amplitude characteristic $A(e^{j\omega T}) = |F(e^{j\omega T})|$, the phase characteristic $\phi(e^{j\omega T}) \triangleq \arctan\{\mathrm{Im}[H(e^{j\omega T})]/\mathrm{Re}[H(e^{j\omega T})]\}$, and the group delay characteristic $\tau(e^{j\omega T}) \triangleq -d\phi(e^{j\omega T})/d\omega$ of the filter. The filter is said to have *linear phase* if $\tau(e^{j\omega T})$ is constant.

Poles and zeros The transfer function of discrete-time filters is sometimes expressed in terms of the complex-valued variable z or the delay operator $\mathcal{D} = z^{-1}$. A treatment of the z-transform is beyond the scope of this text and may be found, for example, in [3]. For our purposes it suffices to know that z^{-1} and $\mathcal{D}$ may be interpreted as a delay of one sampling interval T, and that the choice $z = e^{j\omega T} = e^{j2\pi\Omega}$ reduces the z-transform to a Fourier transform. In z-transform notation, realizable discrete-time filters all have a transfer function of the form

$$F(z) = \frac{b_0 + b_1 z^{-1} + \ldots + b_N z^{-N}}{1 - a_1 z^{-1} - \ldots - a_M z^{-M}} \tag{1.23}$$

where $b_0, \ldots, b_N$ and $a_1, \ldots, a_M$ are arbitrary real-valued coefficients and M and N are nonnegative integers. Unlike analog filters, there is no requirement that M be at least equal to N. In fact, even for $M = 0$ we can choose N at liberty and this leads to a class of filters whose impulse response has a strictly finite duration of $N + 1$ symbol intervals. Such filters are said to be of the FIR (Finite Impulse Response) type. Filters with $M > 0$, by comparison, are usually of the IIR (Infinite Impulse Response) type. IIR filters may be realized in various manners and one possible structure is shown in Fig. 1.4.

FIR filters are conveniently realized in the form of a *transversal filter* or *tapped delay line* (Fig. 1.5). FIR filters can be designed to have an exactly linear phase characteristic. Such a characteristic can, in general, only be approximated with IIR filters. The 'direct-form 2' filter of Fig. 1.4 is among the most straightforward realizations of the transfer function (1.23) but is relatively sensitive to quantization effects, especially for high filter orders. A detailed account of these important effects may be found, for example, in [3].

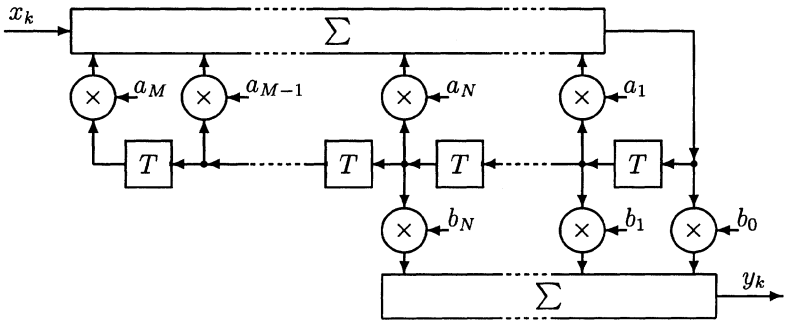


Figure 1.4: Infinite Impulse Response filter with 'direct-form 2' structure.

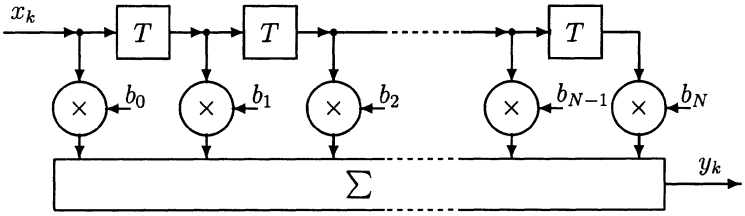


Figure 1.5: Transversal filter.

We can rewrite (1.23) as

$$H(z) = b_0 \frac{\prod_{i=1}^{N}(z - z_i)}{\prod_{j=1}^{M}(z - p_j)} z^{M-N}. \tag{1.24}$$

The complex-valued quantities z_i and p_i are called the *zeros* and *poles* of $H(z)$, respectively. Just as for analog filters, poles and zeros are either real-valued or occur in complex conjugate pairs. Their location determines various filter properties.

1. Except for a scaling factor b_0 the pole/zero constellation fully determines the transfer function.

2. For a filter to be stable, all poles must be within the unit circle $|z| = 1$.

3. For a filter to have minimum phase, all zeros must be within the unit circle. (A filter is said to have minimum phase if, given the amplitude characteristic, the group delay $\tau(e^{j\omega T})$ has the lowest possible value for *all* frequencies; see Section 1.6 for more details.)

4. For a filter to be a stable 'all-pass' network, all poles must be within the unit circle and all zeros outside it. Furthermore, poles and zeros must occur in pairs that are 'reflected' in the unit circle, i.e. for every pole $p_j = re^{j\phi}$ with $0 \leq r < 1$ there must be a zero $z_j = r^{-1}e^{j\phi}$. (An all-pass network, also called phase shifter, is characterized in that $|F| = 1$ for all ω.)

5. Pole-zero plots of filters connected in cascade can be overlaid on top of each other. If a pole and a zero coincide, they cancel one another. By implication, the transfer function (1.24) may alternatively be realized as a cascade of first- and second-order sections. The first-order sections account for the real-valued poles and zeros, and the second-order sections account for the complex conjugate pairs of poles and zeros. Such a cascade realization is often less sensitive to quantization effects than direct-form realizations like those of Figs. 1.4 and 1.5.

In digital transmission systems, discrete-time filters often form part of modulation encoders, transmit filters and equalizers. The properties, design and digital realization of these filters are therefore of considerable interest to digital transmission. These subjects are discussed, among many other books, in [11], [6] and [3].

Example 1.1 The transversal filter of Fig. 1.5 is normally implemented in the discrete-time domain. Even so, a continuous-time implementation is possible and here the filter has impulse response

$$c(t) = \sum_{k=0}^{N} c_k \delta(t - kT) \tag{1.25}$$

where c_k represents the kth filter coefficient and $\delta(t)$ is Dirac's delta-function. This leads to a transfer function

$$C(\omega) = \int_{-\infty}^{\infty} c(t)e^{-j\omega t}dt = \sum_{k=0}^{N-1} c_k \int_{-\infty}^{\infty} \delta(t - kT)e^{-j\omega t} = \sum_{k=0}^{N-1} c_k e^{-j\omega kT}. \tag{1.26}$$

The last equality is a direct consequence of the sieve property (1.13). Now the last part of (1.26) is just the FTD of the filter coefficients c_k, i.e. the transfer function of the discrete-time transversal filter. This shows that the transfer function of a T-spaced transversal filter is domain-independent.

Exercise 1.4 Does this conclusion remain valid when the tap spacing is not an integer multiple of the sampling period T?

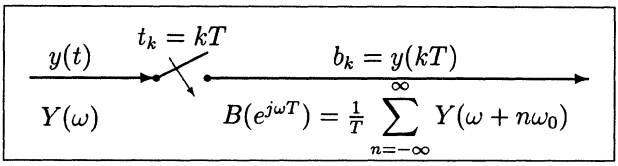


Figure 1.6: Time- and frequency domain effect of a synchronous sampling operation.

1.2.4 Sampling

In Fig. 1.6, a continuous-time signal $y(t)$ is sampled under the control of a clock at periodic instants $t_k = kT$, $k = \ldots, -2, -1, 0, 1, 2, \ldots$. The sampler output is a discrete-time sequence $b_k = y(kT)$. The spectral effect of sampling is expressed by Shannon's sampling theorem [4]

$$B(e^{j\omega T}) = \frac{1}{T}\sum_{n=-\infty}^{\infty} Y(\omega + n\omega_0), \; \omega_0 = \frac{2\pi}{T}. \tag{1.27}$$

Fig. 1.7 provides an illustration for a low-pass signal $y(t)$ with two different sampling rates $1/T$.

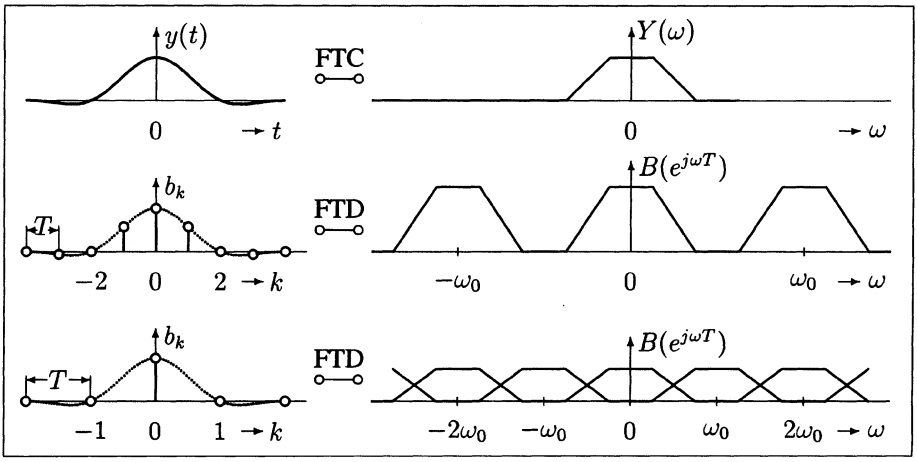


Figure 1.7: Illustration of the spectral effects of sampling.

The left half of Fig. 1.7 depicts $y(t)$ and two sampled versions, with sampling frequencies that differ a factor of 2. The spectral effect of sampling is that $Y(\omega)$ is cut into slices of width ω_0, which are overlaid on top of each other and repeated periodically with period ω_0. In the middle portion of Fig. 1.7 the sampling rate $1/T$ is high enough to avoid any overlap between slices. In the bottom portion, by comparison, slices are

twice as narrow and overlap does occur. This is called *aliasing*. An immediate consequence of aliasing is that $y(t)$ can no longer be reconstructed from b_k. In the absence of aliasing, by comparison, $y(t)$ can in principle be reconstructed without error via a linear pulse modulator that suppresses all translates of Y (see Section 1.2.5).

The scaling factor $1/T$ in the right hand side of (1.27) causes the amplitude of B to increase linearly with the sampling rate. In the absence of aliasing, $TB(e^{j\omega T}) = Y(\omega)$ for $|\omega| \leq \omega_0/2$ and the energies E_b and E_y of b_k and $y(t)$ are related by $T^2 E_b = E_f$. This 'energy conservation property' continues to apply if the sampling phase is changed, i.e. if the sampling instants are $t_k = kT + \tau$ where τ is an arbitrary delay.

Exercise 1.5 Show that (1.27) may be re-expressed in terms of normalized frequencies as

$$B(e^{j2\pi\Omega}) = \frac{1}{T} \sum_{n=-\infty}^{\infty} Y(\Omega + n). \tag{1.28}$$

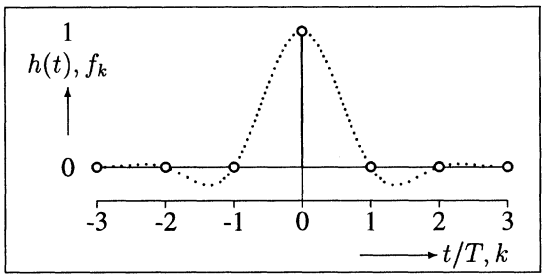


Figure 1.8: Nyquist-1 pulse $h(t)$ (dotted) and its sampled version f_k (open circles).

Example 1.2 A function $h(t)$ is said to be a *Nyquist-1 pulse* if its sampled version $f_k = h(kT)$ amounts to a Kronecker delta-function δ_k. A pulse of this type is shown in Fig. 1.8. From (1.9) we see that f_k has FTD $F(e^{j2\pi\Omega}) = 1$ for all Ω, and with (1.28) it follows that $h(t)$ will be a Nyquist-1 pulse if and only if

$$\frac{1}{T} \sum_{n=-\infty}^{\infty} H(\Omega + n) = 1 \text{ for all } \Omega. \tag{1.29}$$

This is Nyquist's celebrated first criterion for distortionless transmission [5]. Aliasing is permitted but the aliased spectrum must be flat, as in the bottom portion of Fig. 1.7. Chapter 5 will reveal that the Nyquist-1 criterion plays an important role in zero-forcing linear equalization.

1.2.5 Linear pulse modulation

The process of transforming a discrete-time data sequence a_k into a continuous-time signal $s(t)$ is generally known as *modulation*. In radio-frequency transmission systems, modulation involves the creation of a sinusoidal wave whose amplitude and/or frequency is determined by a_k. For frequency modulation the modulator output is a nonlinear function of the input. Amplitude modulation, by comparison, is usually (though

not necessarily) a linear operation, i.e. an input αa_k will produce an output $\alpha s(t)$ irrespective of α. In baseband transmission systems we are mainly concerned with *linear pulse modulation* or *pulse amplitude modulation* (PAM). Here the data sequence a_k is carried by pulses of a fixed form rather than by a sinusoidal carrier wave (Fig. 1.9).

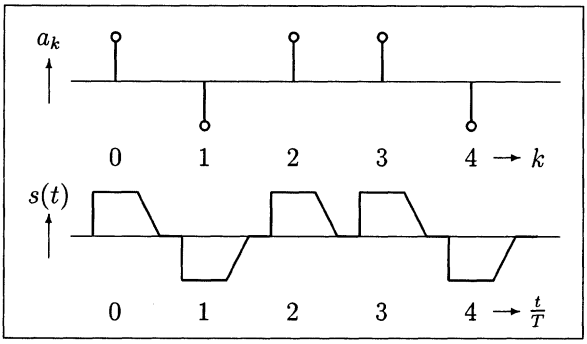


Figure 1.9: Linear pulse modulation of a binary data sequence a_k.

Each digit a_k is replaced by a pulse $a_k c(t - kT)$. Here $c(t)$ is the *symbol response* of the modulator and T is the temporal spacing of digits a_k. The modulator output $s(t)$ is the sum of all pulses, i.e.

$$s(t) \triangleq \sum_{k=-\infty}^{\infty} a_k c(t - kT). \tag{1.30}$$

We may recognize $c(t)$ as the waveform that appears at the output of the modulator when the input is a Kronecker delta function ($a_k = \delta_k$). In effect, the operation described by (1.30) is somewhere in between the continuous-time convolution of (1.11) and the discrete-time convolution of (1.20). For this reason we call it a *hybrid convolution*. A system-theoretical description of the linear pulse modulator as a hybrid system may be found in [14].

From now on we shall usually say 'linear modulation' where we mean 'linear pulse modulation'. Confusion cannot arise because we will not be concerned with other forms of modulation. The Fourier integral $C(\omega)$ of the symbol response $c(t)$ may be regarded as the transfer function of the modulator and is sometimes called *shape factor*. Not surprisingly, the Fourier integral $S(\omega)$ of $s(t)$ satisfies

$$S(\omega) = A(e^{j\omega T})C(\omega), \tag{1.31}$$

i.e. a hybrid convolution translates spectrally into a multiplication of a discrete-time Fourier transform (FTD) and a Fourier integral (FTC). Since $A(e^{j\omega T})$ is periodic in ω with period ω_0, $S(e^{j\omega T})$ basically exhibits the same periodicity except for the multiplicative shape factor $C(\omega)$ that is due to the pulse modulator. This quasi-periodicity restricts the class of spectra $S(\omega)$ that can be produced as a function of the data sequence

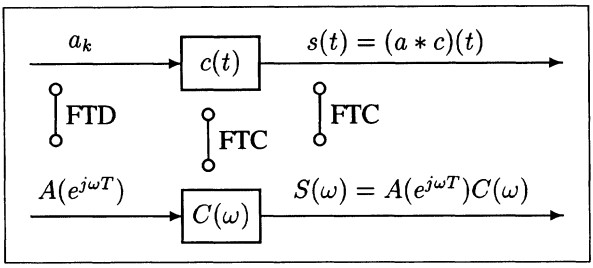


Figure 1.10: A linear pulse modulator transforms a discrete-time data sequence a_k into a continuous-time output $s(t)$ that is transmitted or recorded. The effect of the modulator can be described by a *hybrid convolution.*

a_k. In effect, only a single 'slice' of width ω_0 can be defined at liberty and this slice will fully determine the remainder of $S(\omega)$.

Exercise 1.6 Prove (1.31).

Exercise 1.7 Consider the binary waveform a_k of Fig. 1.9. Assume that $c(t)$ is a rectangular pulse of width αT. How many amplitude levels does $s(t)$ assume for $\alpha = 0.5$, 1 and 2? For which of these values of α is there interference between successive pulses $a_k c(t - kT)$?

Both the linear pulse modulator and the sampler are linear but time-variant. Their cascade, however, can be time-invariant.

Example 1.3 Fig. 1.11 shows the cascade of a linear pulse modulator and a sampler. This cascade is desired to be non-distorting, i.e. the output b_k of the sampler is desired to equal the input sequence a_k of the modulator. Clearly

$$b_k = s(kT) = \sum_{i=-\infty}^{\infty} a_i c(kT - iT). \tag{1.32}$$

We see that $b_k = a_k$ if and only if $c(kT) = \delta_k$, i.e. if and only if $c(t)$ is a Nyquist-1 pulse as in Example 1.2.

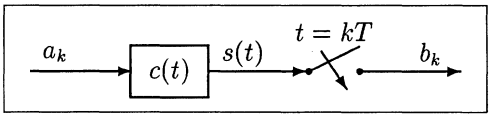


Figure 1.11: Cascade of linear pulse modulator and sampler.

Example 1.4 Fig. 1.12 shows the cascade of a sampler and a linear pulse modulator. The modulator is meant as a reconstruction filter, i.e. its output $\tilde{y}(t)$ is desired to equal the sampler input $y(t)$. This is, in general, only possible if no aliasing occurs, i.e. if $y(t)$ has no spectral content above half of the sampling frequency $1/T$. Subject to this condition the spectrum $B(e^{j\omega T})$ of b_k is a periodic repetition of that of $y(t)$ (cf. the middle portion of Fig. 1.7). To retain only the

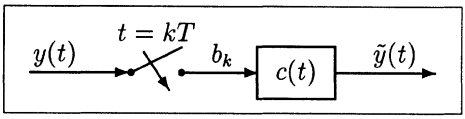


Figure 1.12: Cascade of sampler and linear pulse modulator.

baseband component while rejecting the aliases we need a shape factor

$$C(\omega) = \begin{cases} T, & |\omega| < \frac{\omega_0}{2}, \\ 0, & \text{else.} \end{cases} \tag{1.33}$$

The gain T in the passband is needed to compensate for the gain factor $1/T$ that emerges in the sampling operation (compare (1.28)). The symbol response $c(t)$ of an ideal reconstruction filter is the inverse Fourier transform of $C(\omega)$, i.e.

$$c(t) = \frac{\sin \pi t/T}{\pi t/T}. \tag{1.34}$$

This is a sinc-pulse of infinite duration. In practice a finite-length approximation must be used.

The cascade of Fig. 1.11 is time-invariant provided only that sampling occurs at a fixed phase. The spectral content of a_k and the exact symbol response $c(t)$ are of no concern. By comparison, the cascade of Fig. 1.12 is time-invariant under the conditions of Example 1.4, but it is time-variant in many other cases (see Problems 1.9 and 1.11).

1.3 STOCHASTIC SIGNALS AND SEQUENCES

Digital transmission systems are often hampered by noise. The transmitted data and the noise share at least one property: both are not precisely known beforehand, i.e. both have a *stochastic* or *random* nature. All that is known beforehand is that they belong to some class or *ensemble*. For example, if the data is produced via a binary DC-free modulation code, then we know beforehand that only two data levels are admissible and that valid data sequences are all DC-free. Ideally the transmission system functions properly across the complete data and noise ensembles, i.e. for all possible data sequences and noise signals. We will see later that system performance depends not just on the power of the noise but also on how this power is distributed across frequency. This distribution may be described by a *power spectral density* function (PSD). Of similar importance is the PSD of the data sequence. For the definition and calculation of PSDs we must describe the data and noise in terms of *random processes*. This is the subject of Sections 1.3.1 and 1.3.2, which pertain to discrete-time and continuous-time processes, respectively. Just as for deterministic sequences and signals, stochastic sequences and signals are subject to linear transformations within and between the discrete-time and continuous-time domains. The statistical effects of these transformations are reviewed in Sections 1.3.3 to 1.3.6. The treatment is rather compact and assumes that the reader is familiar with such concepts of probability theory as random variables and probabilities. A more detailed treatment may be found in [9] and [10].

1.3.1 Discrete-time random processes

Let us consider the collection of all ternary DC-free sequences of length $2N + 1$. Individual members of this ensemble are denoted a_k^i where k is a time index and i is a member index. By way of illustration Fig. 1.13 depicts four members or *sample functions* of the $N = 3$ ensemble.

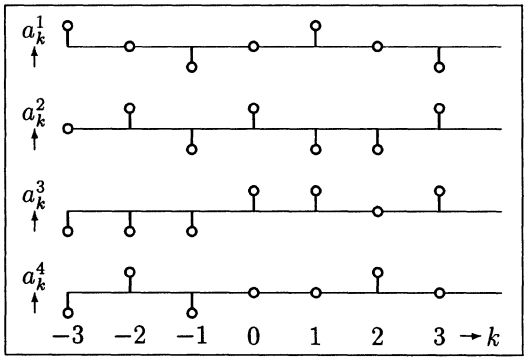


Figure 1.13: Four sample functions of the ensemble of ternary DC-free sequences of length 7.

The ensemble is the collection of all possible sequences a_k^i. The sequence number i is a *random variable* that depends on the message that is to be conveyed. This message is not known beforehand, but it is generally known how likely each possible message is. The ensemble along with the probabilities of the contained sample functions is called a *discrete-time random process*. Observation of the process at instant k yields a random variable that we denote as a_k. For simplicity of notation we will also denote the random process as a_k. It will be evident from the context which interpretation is meant.

Stationary processes Statistical properties of random sequences are often independent of the moment at which observation of the sequence begins. For example, the average of N consecutive samples may be insensitive to the precise time at which the first sample is taken. A process is said to be *stationary* if the statistical properties of the random variable a_k are independent of k. Thus a stationary process is fully characterized by the statistical properties of a_k at a single and arbitrary instant k_0.

Example 1.5 We can associate to each sample function a_k^i a mean

$$\overline{a}^i \triangleq \frac{1}{2N+1} \sum_{k=-N}^{N} a_k^i. \tag{1.35}$$

The mean $\overline{a}$ of the process is the ensemble average of $\overline{a}^i$ and may be expressed as

$$\overline{a} = \frac{1}{2N+1} \sum_{k=-N}^{N} E[a_k] \tag{1.36}$$

where $E[.]$ denotes averaging across the ensemble. For a stationary process, $E[a_k]$ will not depend on k and therefore $\overline{a} = E[a_{k_0}]$ irrespective of N, where k_0 is an arbitrary time instant.

Power To each sample function a_k^i we can associate an *energy* $E_a^i \triangleq \sum_{k=-N}^{N}(a_k^i)^2$. We should note that E_a^i is not an energy in the physical sense of the word, because one clearly cannot heat up a resistor with digits. For this reason it is also not *a priori* obvious how the *power* of a_k^i is to be defined.

- We could normalize E_a^i on a *per-digit* basis. This gives rise to a dimensionless *per-digit power* $P_a^i \triangleq E_a^i/(2N+1)$. In practice we usually deal with ensembles for which N is not bounded and in this case we must define

$$P_a^i \triangleq \lim_{N\to\infty} \frac{1}{2N+1} E_a^i. \tag{1.37}$$

- By analogy to continuous-time signals, we could also define power as the 'energy' per second. This leads to the definition

$$\tilde{P}_a^i \triangleq \lim_{N\to\infty} \frac{1}{(2N+1)T} E_a^i = \frac{1}{T} P_a^i \tag{1.38}$$

  where T denotes the temporal spacing of digits a_k. This power has dimension s^{-1} and is of particular interest in situations where data sequences are converted to continuous-time format and vice versa.

Unless noted otherwise we will use the per-digit definition of (1.37). The power P_a of the process a_k may then be defined as the ensemble-average of P_a^i. A stationary process has power $P_a = E[a_k^2]$.

Power spectral density We are now in a position to define the power spectral density of the process. To this end we first observe that each sample function a_k^i has a Fourier transform

$$A^i(e^{j\omega T}) = \sum_{k=-N}^{N} a_k^i e^{-jk\omega T}. \tag{1.39}$$

The magnitude squared of $A^i(e^{j\omega T})$ describes the distribution of *energy* of a_k^i across frequency, and the function

$$\mathcal{A}^i(e^{j\omega T}) \triangleq \lim_{N\to\infty} \frac{1}{2N+1} |A^i(e^{j\omega T})|^2 \tag{1.40}$$

describes the distribution of per-digit power. The power spectral density $\mathcal{A}(e^{j\omega T})$ of the process is the ensemble average of $\mathcal{A}^i(e^{j\omega T})$, i.e.

$$\mathcal{A}(e^{j\omega T}) \triangleq \lim_{N\to\infty} \frac{1}{2N+1} E[|A^i(e^{j\omega T})|^2]. \tag{1.41}$$

Exercise 1.8 Assume that all sample functions a_k^i in the ensemble are DC-free. Prove that $\mathcal{A}(e^{j\omega T}) = 0$ at DC ($\omega = 0$).

Wiener-Kinchine relation for stationary processes Upon using (1.39) in (1.41) we find after a change of variables that

$$\mathcal{A}(e^{j\omega T}) = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{k=-N}^{N} \sum_{n=-N-k}^{N-k} E[a_k a_{k+n}] e^{-jn\omega T}. \tag{1.42}$$

The expectation $R_{k,n}^a \triangleq E[a_k a_{k+n}]$ is known as the *discrete-time autocorrelation function* of the process. For a stationary process $R_{k,n}$ will depend only on the *time lag* n and not on the absolute time index k. This allows us to omit the subscript k so that

$$\mathcal{A}(e^{j\omega T}) = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{k=-N}^{N} \sum_{n=-N-k}^{N-k} R_n^a e^{-jn\omega T}. \tag{1.43}$$

The factor $R_n^a e^{-jn\omega T}$ is dependent of k, and practical functions R_n^a usually decay towards zero for sufficiently large $|n|$. These two facts allow us to rewrite the PSD as

$$\mathcal{A}(e^{j\omega T}) = \sum_{n=-\infty}^{\infty} R_n^a e^{-jn\omega T}. \tag{1.44}$$

Thus $\mathcal{A}(e^{j\omega T})$ is the FTD of R_n^a. Equation (1.44) is known as the discrete-time version of the Wiener-Kinchine theorem [9].

Example 1.6 A stationary process is said to be *uncorrelated* if the autocorrelation function differs from zero for $n = 0$ only. Thus $R_n^a = R_0^a \delta_n$ where δ_n is the Kronecker delta function. With (1.44) this means that $\mathcal{A}(e^{j\omega T}) = R_0^a$ for all ω, i.e. uncorrelated processes have a flat PSD.

Exercise 1.9 Prove the following properties of R_n^a: 1) $R_{-n}^a = R_n^a$ for all n; 2) $R_0^a = P_a \geq 0$; 3) $R_n^a \leq R_0^a$ for all k. What are the implications of these properties for $\mathcal{A}(e^{j\omega T})$?

Wide-sense stationarity Any process a_k for which $E[a_k]$ and $R_{k,n}^a = E[a_k a_{k+n}]$ are both independent of the absolute time index k is said to be wide-sense stationary (WSS). Wide-sense stationarity is a weaker condition than stationarity but suffices, for example, for the Wiener-Kinchine relation to be valid. A stationary process is sometimes called *strict-sense stationary* in order to stress the fact that it is not just WSS. For our purposes it will usually suffice to assume that processes are WSS.

1.3.2 Continuous-time random processes

Most of the concepts and properties of discrete-time processes carry over directly to the continuous-time domain. By way of illustration we consider a noise process $n(t)$.

By analogy to the discrete-time case, this is the ensemble of all possible noise signals along with the probability measure that is associated to the members or *sample functions* of the ensemble. The autocorrelation function $R_n(t,\tau)$ of the noise process is defined as the average of $n(t)n(t+\tau)$ across the ensemble, i.e.

$$R_n(t,\tau) \triangleq E[n(t)n(t+\tau)]. \tag{1.45}$$

The *noise power* $P_n(t)$ is the average of $n^2(t)$, i.e. $P_n(t) \triangleq E[n^2(t)] = R_n(t,0)$. In many instances $R_n(t,\tau)$, $P_n(t)$ and the mean $E[n(t)]$ are independent of the absolute time t. The process is called *wide-sense stationary* when this is so. For processes of this kind we shall omit the argument t to R_n, P_n and $\overline{n}$.

Exercise 1.10 Prove that $R_n(\tau)$ is a real-valued and even function of τ with a global maximum $P_n = R_n(0)$ for $t = 0$.

The PSD of $n(t)$ is defined as

$$\mathcal{N}(\omega) \triangleq \lim_{\theta\to\infty} \frac{1}{2\theta} E[|N_\theta(\omega)|^2]. \tag{1.46}$$

In this formula $N_\theta(\omega)$ is the Fourier integral of the finite-duration noise signal $n_\theta(t)$, defined as

$$n_\theta(t) \triangleq \begin{cases} n(t) & \text{for } |t| < \theta \text{ and} \\ 0 & \text{for } |t| > \theta. \end{cases} \tag{1.47}$$

Again we see, as we did in Section 1.3.1, that the PSD is obtained in two steps. First we take the ensemble average of $N_\theta(\omega)$, so that we do not meet problems with the FTC of a signal which may not be absolutely integrable. Then we let θ go to infinity. The resulting PSD is a real-valued, non-negative, even and non-periodic function of frequency.

The Wiener-Kinchine theorem for continuous-time processes says that the PSD of $n(t)$ forms a Fourier pair with the time-averaged version of the autocorrelation function $R_n(t,\tau)$:

$$\mathcal{N}(\omega) \circ\!\!-\!\!\circ A[R_n(t,\tau)] \tag{1.48}$$

where the operator $A[.]$ is defined as

$$A[.] \triangleq \lim_{\theta\to\infty} \frac{1}{2\theta} \int_{-\theta}^{\theta} [.]dt. \tag{1.49}$$

For a WSS process this averaging operator has no effect so that

$$\mathcal{N}(\omega) \circ\!\!-\!\!\circ R_n(\tau). \tag{1.50}$$

Exercise 1.11 Express P_n in terms of the noise PSD $\mathcal{N}(\omega)$.

White noise A WSS noise process is said to be white if its PSD is independent of frequency, i.e. if $\mathcal{N}(\omega) = N_0$ for some constant $N_0 \geq 0$. Such a process has autocorrelation function $R_n(\tau) = N_0\delta(\tau)$ and the power $P_n = R_n(0)$ is consequently infinite. This is clearly unphysical but even so white noise is often used in theoretical studies of communication systems. The reason is that many practical systems have a noise PSD that is approximately constant throughout the band in which information is transmitted. Frequencies beyond this band are rejected by a filter in the receiver and the exact value of the noise PSD in this stop-band is of no concern. This allows us to consider noise as white so as to simplify computations relative to a more physical PSD. Noise that is not white is sometimes said to be *colored.*

Cyclostationary processes A process is called *cyclostationary* with period T if $R_n(t,\tau) = R_n(t+T,\tau)$ for all t. We will see in Section 1.3.5 that a linear pulse modulator produces a process of this type when excited by a WSS discrete-time process. For a cyclostationary process it suffices to extend the averaging operation $A[.]$ across a single period and therefore

$$\mathcal{N}(\omega) \circ\!\!-\!\!\circ \frac{1}{T}\int_{-T/2}^{T/2} R_n(t,\tau)dt. \tag{1.51}$$

1.3.3 Linear filtering of WSS processes

Continuous-time random signals In Fig. 1.14, a sample function of a WSS process $x(t)$ with autocorrelation function $R_x(\tau)$ and PSD $\mathcal{X}(\omega)$ is applied to a continuous-time filter with impulse response $h(t)$ and transfer function $H(\omega)$. The random process at the filter output is denoted $y(t)$ and has autocorrelation function

$$\begin{aligned} E[y(t)y(t+\tau)] &= E[\int_{-\infty}^{\infty} x(\mu)h(t-\mu)d\mu \int_{-\infty}^{\infty} x(\rho)h(t+\tau-\rho)d\rho] \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} R_x(\rho-\mu)h(t-\mu)h(t+\tau-\rho)d\mu d\rho \\ &= \int_{-\infty}^{\infty} (R_x * h)(t+\tau-\mu)h(t-\mu)d\mu \\ &= (R_x * h * h_-)(\tau) \end{aligned} \tag{1.52}$$

where '$_-$' denotes time-reversal, i.e. $h_-(t) \triangleq h(-t)$ for all t. The second step is due to the fact that $x(t)$ is WSS. Not surprisingly, the autocorrelation function of $y(t)$ does not depend on t. The same is true for the mean of $y(t)$ and therefore $y(t)$ is also WSS.

Exercise 1.12 Express $E[y(t)]$ in terms of $\overline{x}$ and $h(t)$. Verify that $E[y(t)]$ is independent of t.

We see from (1.52) that the autocorrelation function $R_y(\tau)$ of $y(t)$ is obtained by convolving $R_x(\tau)$ by a function

$$(h * h_-)(t) = \int_{-\infty}^{\infty} h(\mu)h(t+\mu)d\mu. \tag{1.53}$$

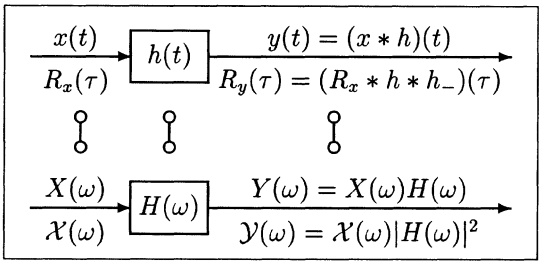


Figure 1.14: Temporal and spectral effects of a linear filter on continuous-time random signals.

This function is called the *autocorrelation function* of the filter. It satisfies the same properties as the autocorrelation function of a WSS process. In particular, $(h * h_-)(t)$ is an even function of t with a global maximum for $t = 0$. The Fourier integral $|H(\omega)|^2$ of $(h * h_-)(t)$ is called the *power transfer function* of the filter. The PSD $\mathcal{Y}(\omega)$ of $y(t)$ is the Fourier integral of $R_y(\tau)$, i.e.

$$\mathcal{Y}(\omega) = \mathcal{X}(\omega)|H(\omega)|^2. \tag{1.54}$$

Hence the PSD of a filtered WSS process is just the product of the PSD of the input process and the power transfer function of the filter.

Discrete-time random sequences In Fig. 1.15, a sample function of a discrete-time WSS process a_k with autocorrelation function R_k^a is applied to a linear filter with impulse response f_k. The output process b_k is WSS and has autocorrelation function

$$R_k^b = (R^a * f * f_-)_k. \tag{1.55}$$

The PSDs $\mathcal{B}(e^{j\omega T})$ and $\mathcal{A}(e^{j\omega T})$ of b_k and a_k are related by

$$\mathcal{A}(e^{j\omega T}) = \mathcal{B}(e^{j\omega T})|F(e^{j\omega T})|^2 \tag{1.56}$$

where $|F(e^{j\omega T})|^2$ is referred to as the power transfer function of the filter. These relations are completely analogous to those for a continuous-time filter.

1.3.4 Sampling of WSS processes

In Fig. 1.16, a continuous-time random process $n(t)$ is sampled at instants $t_k = (k+\phi)T$ where ϕ is a fixed sampling phase. If $n(t)$ is WSS then the the sampled sequence $u_k = n(t_k)$ will also be WSS. For this reason u_k has mean $\overline{u} = E[u_k] = E[n(t_k)] = \overline{n}$, i.e. sampling does not affect the mean of a WSS process. Furthermore, u_k has autocorrelation function

$$R_k^u = E[u_l u_{l+k}] = E[n(t_l)n(t_l + kT)] = R_n(kT). \tag{1.57}$$

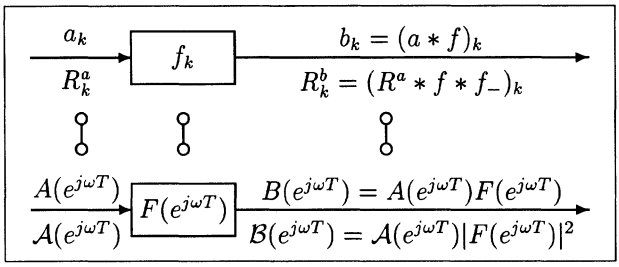


Figure 1.15: Temporal and spectral effects of a linear filter on discrete-time random sequences.

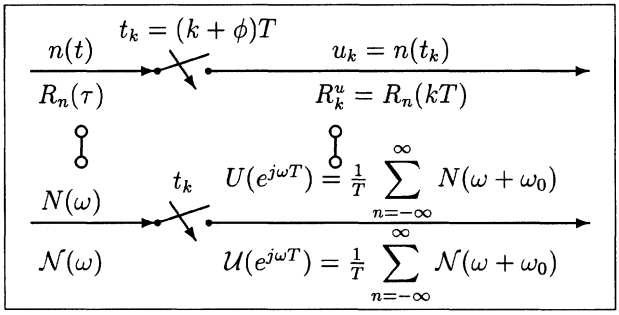


Figure 1.16: Temporal and spectral effects of sampling.

The last step is due to the fact that $n(t)$ is WSS. It follows that the autocorrelation function of u_k is just a sampled version of that of $n(t)$. The sampling phase ϕ does not come into play. The spectral effect of sampling is a folding operation according to

$$\mathcal{U}(e^{j\omega T}) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \mathcal{N}(\omega + n\omega_0), \ \omega_0 = \frac{2\pi}{T}. \tag{1.58}$$

This expression is similar to (1.27), be it that folding now occurs on a power rather than voltage basis. The per-digit power $P_u = E[u_k^2]$ amounts to

$$P_u = R_0^u = R_n(0) = P_n = E[n^2(t)] \tag{1.59}$$

i.e. the per-digit power of the sampler output is equal to the physical power at the sampler input. In spectral terms we have

$$P_u = \frac{1}{\omega_0} \int_{-\omega_0/2}^{\omega_0/2} \mathcal{U}(e^{j\omega T}) d\omega = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathcal{N}(\omega) d\omega. \tag{1.60}$$

1.3.5 PAM processes

In digital transmission systems, linear pulse modulators serve to convert a random data sequence a_k of data rate $1/T$ into a continuous-time data signal $s(t)$ that is trans-

mitted or recorded (Fig. 1.17). The random process that is produced by the modulator is sometimes called a *PAM process*. This process is, in general, not stationary but rather cyclostationary with period T. For an illustration we may re-inspect Fig. 1.9. Here $s(t)$ is zero during the last quarter of each symbol interval and nonzero elsewhere. Statistical properties of $s(t)$ are evidently periodic in t with period T. For the mean of $s(t)$ we may write

$$E[s(t)] = E[\sum_{k=-\infty}^{\infty} a_k c(t-kT)] = \sum_{k=-\infty}^{\infty} \overline{a} c(t-kT) = \overline{a}\,\overline{c}_T(t) \tag{1.61}$$

where $\overline{c}_T(t) \triangleq \sum_{k=-\infty}^{\infty} c(t-kT)$. This is, in general, a periodic function of t with period T. The same must be true for $E[s(t)]$ except when $\overline{a} = 0$. The autocorrelation function of $s(t)$ is

$$\begin{aligned} R_s(t,\tau) &= E[s(t)s(t+\tau)] = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} E[a_i a_j] c(t-iT)c(t+\tau-jT) \\ &= \sum_{i=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} E[a_i a_{k+i}] c(t-iT)c(t+\tau-iT-kT) \\ &= \sum_{k=-\infty}^{\infty} R_k^a \gamma(t,\tau-kT) \end{aligned} \tag{1.62}$$

where $\gamma(t,\tau) \triangleq \sum_{i=-\infty}^{\infty} c(t-iT)c(t+\tau-iT)$. This so-called *kernel* is a periodic function of t with period T, and the same must be true for $R_s(t,\tau)$.

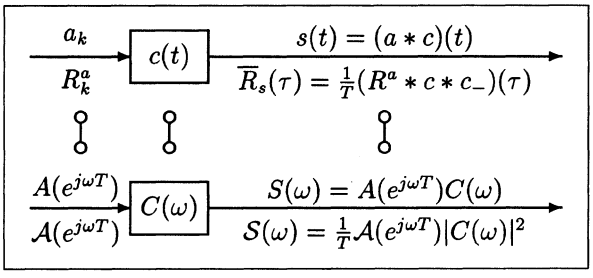


Figure 1.17: Temporal and spectral effects of linear pulse modulation.

According to the Wiener-Kinchine theorem, the PSD of $s(t)$ is the Fourier transform of the averaged autocorrelation function $\overline{R}_s(\tau) \triangleq \frac{1}{T}\int_{-T/2}^{T/2} R_s(t,\tau)dt$ (compare (1.51)). Clearly

$$\overline{R}_s(\tau) = \frac{1}{T}\sum_{k=-\infty}^{\infty} R_k^a \int_{-T/2}^{T/2} \gamma(t,\tau-kT)dt. \tag{1.63}$$

The integral in this expression may be expanded as

$$\int_{-T/2}^{T/2} \sum_{i=-\infty}^{\infty} c(t-iT)c(t+\tau-iT) = \int_{-\infty}^{\infty} c(\mu)c(\tau+\mu)d\mu = (c * c_-)(\tau). \quad (1.64)$$

Correspondingly

$$\overline{R}_s(\tau) = \frac{1}{T} \sum_{k=-\infty}^{\infty} R_k^a (c * c_-)(\tau - kT). \quad (1.65)$$

A system interpretation [9], [14] of this equation is that the averaged autocorrelation function may be obtained by applying the sequence R_k^a to a linear pulse modulator with symbol response $\frac{1}{T}(c * c_-)(t)$. This interpretation allows us to determine the PSD of $s(t)$ by direct application of the results of Section 1.2.5. The PSD is

$$\mathcal{S}(\omega) = \frac{1}{T}\mathcal{A}(e^{j\omega T})|C(\omega)|^2 \quad (1.66)$$

where $\mathcal{A}(e^{j\omega T})$ is the PSD of a_k and $C(\omega)$ is the shape factor of the modulator. By analogy to filtered random signals, we may regard $\frac{1}{T}|C(\omega)|^2$ as the power transfer function of the modulator.

At a first glance the factor $1/T$ appearing in (1.65) and (1.66) may seem somewhat puzzling. For a clarification we recall that the we have defined the power of discrete-time sequences on a per-digit basis (see Section 1.3.1). The factor $1/T$ accounts for the conversion of per-digit power into per-second power. The power transfer function $T^{-1}|C(\omega)|^2$, therefore, converts between per-symbol power and per-second power. Alternatively, if we define $\tilde{\mathcal{A}}(e^{j\omega T}) \triangleq T^{-1}\mathcal{A}(e^{j\omega T})$ as the 'per-second' PSD of a_k, then $S(\omega) = \tilde{\mathcal{A}}(e^{j\omega T})|C(\omega)|^2$, i.e. the modulator has per-second to per-second power transfer function $|C(\omega)|^2$. A similar reasoning applies to the relation between R_k^a and $\overline{R}_s(\tau)$.

Filtered cyclostationary signals In the chapters ahead we will be concerned with the situation that $s(t)$ is a transmitted data signal and that a data receiver operates on a filtered and noisy version of $s(t)$. The effect of filtering is addressed in Fig. 1.18.

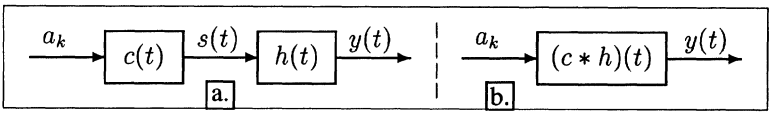


Figure 1.18: a. Cascade of linear pulse modulator with symbol response $c(t)$ and filter with impulse response $h(t)$. The output $y(t)$ of the cascade can be produced directly by a linear pulse modulator with symbol response $(c * h)(t)$.

The filter has impulse response $h(t)$. Its output $y(t)$ may be thought of as being produced directly by a linear pulse modulator with symbol response $(c * h)(t)$. The implication is that any filtered version of $s(t)$ must also be cyclostationary. This cyclostationarity is central to the reception process because decisions are taken once every

T seconds at moments that are determined by a timing recovery circuit. Cyclostationarity is also at the heart of the *eye pattern* of a data signal, which will be introduced in Chapters 2 and 3. The PSDs of $s(t)$ and $y(t)$ do not pertain to a single sampling phase but are averages across a complete symbol interval. For this reason they are of restricted importance to receiver performance.

1.3.6 Synchronous sampling of PAM processes

The clock recovery circuit in the receiver is intended to sample the received data signal in synchronism with the transmitted data bits. The archetypal situation is depicted in Fig. 1.19. The output $s(t)$ of the linear pulse modulator arrives undistorted at the receiving end of the system and is sampled at instants $t_k = (k + \phi)T$ where ϕ denotes the sampling phase. If $s(t)$ is as in Fig. 1.9 then a sampling phase $\phi = 0$ will produce a binary sampler output $b_k = a_k$, while phases $\phi \in [0.5, 0.75]$ will produce the all-zero sequence $b_k = 0$. This illustrates how critical it can be to reception that ϕ be chosen properly.

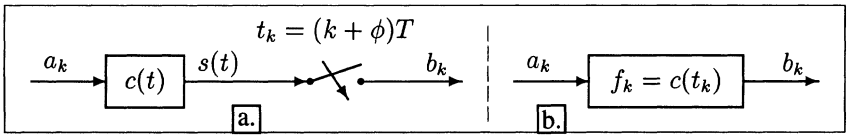


Figure 1.19: a. Cascade of linear pulse modulator and sampler. b. Equivalent discrete-time model.

The output b_k of the sampler is stationary by virtue of the fact that sampling occurs in synchronism with the transmitted data sequence. We can think of b_k as originating from a_k via a discrete-time filter with impulse response $f_k \triangleq c(t_k) = c((k + \phi)T)$ (Fig. 1.19.b). The transfer function $F(e^{j\omega T})$ of the filter is given by

$$F(e^{j\omega T}) = \frac{1}{T} \sum_{n=-\infty}^{\infty} C(\omega + n\omega_0) e^{j\phi(\omega + n\omega_0)} \tag{1.67}$$

where $C(\omega)$ is the shape factor of the modulator. Both f_k and $F(e^{j\omega T})$ depend heavily on ϕ. In the chapters ahead we will repeatedly encounter cascades akin to those of Fig. 1.19 and their equivalent discrete-time models.

Exercise 1.13 Prove (1.67). Express the PSD of b_k in terms of that of a_k.

1.3.7 Joint random processes

Random signals are often generated simultaneously at various places within a system. Each signal belongs to an ensemble and may or may not be related to any of the other signals. There are two extreme situations:

1. There is no relation whatsoever. For example, noise signals in digital transmission systems are normally produced by noise sources whose action depends in no way on the transmitted data sequence.

2. There is a deterministic relation. This is so, for example, for the input and output signals of a linear filter.

The collection of all ensembles along with the underlying joint probability function of the sample functions is referred to as a *joint random process*. A detailed treatment of such processes is beyond the scope of this text and may be found, for example, in [15, ch. 3] and [1]. For our purposes it suffices to consider the first extreme situation. (The second one has already been treated in as far as we will need it.)

Independent processes Two processes $x(t)$ and $y(t)$ are said to be statistically independent if there is no statistical relation between both (see [15, ch. 3] for a more rigorous definition). For such processes, any expectation across the joint ensemble can be split into expectations across the individual ensembles. Thus, for example, $E[x(t) + y(t)] = E[x(t)] + E[y(t)]$ and $E[x(t)y(t)] = E[x(t)]E[y(t)]$. These two properties suffice to determine the mean, power and autocorrelation function of both the sum and product of $x(t)$ and $y(t)$. A summary is provided in Table 1.1 for the situation that $x(t)$ and $y(t)$ are both WSS. The sum and product of $x(t)$ and $y(t)$ are then also WSS. The properties of Table 1.1 carry over directly to the discrete-time domain.

$z(t)$	$\overline{z}$	P_z	$R_z(\tau)$
$x(t) + y(t)$	$\overline{x} + \overline{y}$	$P_x + 2\overline{x}\,\overline{y} + P_y$	$R_x(\tau) + 2\overline{x}\,\overline{y} + R_y(\tau)$
$x(t)y(t)$	$\overline{x}\,\overline{y}$	$P_x P_y$	$R_x(\tau)R_y(\tau)$

Table 1.1: Statistical properties of the sum and product of two independent WSS processes $x(t)$ and $y(t)$.

1.4 PROBABILITY DENSITY FUNCTIONS

So far we have described a stationary random process in terms of mean, power, autocorrelation function and power spectral density. These characteristics, while important, are not complete. That is, two processes can be indistinguishable in terms of these characteristics yet be quite different in nature.

Example 1.7 A binary data sequence $a_k \in \{-1, 1\}$ can convey a single bit of information per symbol interval, while a quaternary sequence $b_k \in \{\pm 1/\sqrt{5}, \pm 3/\sqrt{5}\}$ can convey two bits. To this end, successive data symbols must be statistically independent and all symbol values in the alphabet must be equally likely. In this event a_k and b_k both have zero mean, power $P_a = P_b = 1$, autocorrelation function $R_a(k) = R_b(k) = \delta_k$ and power spectral density $\mathcal{A}(e^{j2\pi\Omega}) = \mathcal{B}(e^{j2\pi\Omega}) = 1$. Even so, they are quite different from a transmission standpoint.

This example illustrates that the characteristics defined thus far convey little information about the permissible amplitude levels of a process and their probability of occurrence. This information is often important and for discrete-valued processes a it may be characterized in terms of a *probability function* $p(a)$. The domain A of this function is the set of all possible values of a. The function value $p(a)$ is the probability that a will occur. Probabilities $p(a)$ are all non-negative and must sum to one, i.e.

$$\sum_{a \in A} p(a) = 1. \tag{1.68}$$

Probability functions relate, strictly speaking, to random variables rather than to stochastic processes. Every digit a_k of a process a is a random variable and therefore has a probability function $p(a_k)$. For a stationary process the probability functions of successive digits are all identical, and we can say that the *process* has probability function $p(a_k)$. To stress the fact that this function does not depend on the time index we will henceforth omit this index.

In the context of probability functions and probability density functions we will normally be concerned with processes that are *ergodic*. An ergodic process is one, coarsely speaking, of which all ensemble averages are equal to the corresponding time averages (see [9] for a more precise definition). This is a stronger condition than stationarity, and it allows us to regard the probabilities $p(a)$ interchangeably as time averages or as ensemble averages.

It is worth observing that $p(a)$ fully determines the mean $\overline{a}$ and power P_a of a process. Specifically, $\overline{a}$ is just the average of all possible values of a weighted by their probability of occurrence, i.e.

$$\overline{a} = \sum_{a \in A} a\, p(a). \tag{1.69}$$

Similarly, P_a is just the weighted average of all possible values of a^2, i.e.

$$P_a = \sum_{a \in A} a^2 p(a). \tag{1.70}$$

Example 1.8 The process a_k of Example 1.7 has domain $A = \{-1, 1\}$, probability function $p(a) = 0.5$ for all $a \in A$, mean $\overline{a} = 0.5[(-1) + 1] = 0$, and power $P_a = 0.5[(-1)^2 + 1^2] = 1$. The process b_k has domain $B = \{\pm 1/\sqrt{5}, \pm 3/\sqrt{5}\}$ and probability function $p(b) = 1/4$ for all $b \in B$. Its mean and power are the same as those of a.

Probability functions capture amplitude-related information that is quite orthogonal to the temporal information provided by the autocorrelation and power spectral density functions. They are readily extended to continuous-valued processes. As the number of amplitude levels increases, we eventually arrive at a continuous domain X with infinitely many values. To each value $x \in X$ we can associate a function value $p(x)$ that can be interpreted as a *probability density*. Any sub-interval X_0 of X has a well-defined probability

$$\Pr(X_0) = \int_{x \in X_0} p(x)dx. \tag{1.71}$$

For $p(x)$ to be a valid probability density function (PDF) it is necessary that all function values are non-negative and that the total probability mass amounts to unity, i.e.

$$\Pr(X) = \int_{x \in X} p(x)dx = 1. \tag{1.72}$$

By analogy to probability functions, $p(x)$ fully determines the mean and power of x according to

$$\overline{x} = \int_{x \in X} x\, p(x)dx \tag{1.73}$$

and

$$P_x = \int_{x \in X} x^2 p(x)dx. \tag{1.74}$$

Since $\overline{x}$ and P_x are just the expected values of x and x^2, they are also denoted $E[x]$ and $E[x^2]$, respectively. Probability mass is centered about $\overline{x}$ and the location of this center affects $E[x^2]$. To undo this influence one may consider the *variance* σ_x^2 of x. This is just the power of the centered version of x, i.e.

$$\sigma_x^2 \triangleq \int_{x \in X} (x - \overline{x})^2 p(x)dx. \tag{1.75}$$

The variance of a discrete-valued process is defined analogously. Clearly $E[x^2] = \sigma_x^2 + \overline{x}^2$, i.e. the power of a process cannot be smaller than its variance.

We proceed to discuss two popular density functions and some of their properties.

Uniform PDF This function has a finite domain $X = [\alpha \ldots \beta]$ and all values $x \in X$ are equally likely (Fig. 1.20). In view of the normalization (1.72) we have

$$p(x) = \begin{cases} \dfrac{1}{\beta - \alpha} & \text{for } x \in X \text{ and} \\ 0 & \text{otherwise.} \end{cases} \tag{1.76}$$

The mean, power and variance of the underlying process amount to

$$\overline{x} = \frac{\alpha + \beta}{2},\ P_x = \frac{\alpha^2 + \beta^2 + \alpha\beta}{3},\ \text{and } \sigma_x^2 = \frac{(\alpha - \beta)^2}{3}. \tag{1.77}$$

In the particular event that $p(x)$ is symmetric about 0 (i.e. that $\alpha = -\beta$) we have $\overline{x} = 0$ and $P_x = \sigma_x^2 = \alpha^2/3$.

Gaussian PDF This function has an infinite domain $X = \mathbb{R}$ and is given by

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{(x - \overline{x})^2}{2\sigma^2}\} \tag{1.78}$$

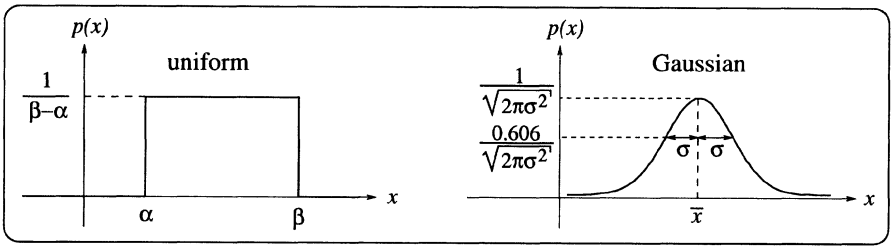


Figure 1.20: Two probability density functions.

where $\overline{x}$ and σ^2 denote the mean and variance of x (Fig. 1.20). Gaussian distribution functions (also called normal) are often encountered and noise processes in particular are often found or assumed to have a Gaussian PDF.

The importance of the Gaussian PDF is largely rooted in the *central limit theorem.* This is actually a group of related theorems which are collected under a single name. These theorems indicate, coarsely speaking, that the PDF of a sum of N random components tends towards Gaussian as N increases, provided only that the N components are statistically independent. This is even true (with few exceptions) if the N underlying PDFs are not Gaussian. Noise is often a sum of many small and unrelated disturbance components and is as such often Gaussian or approximately so. A linear filter basically forms the sum (or integral) of a number of input components so that it tends to produce an output PDF that is 'more Gaussian' than that of its input. (It is, in fact, simple to relate the PDF of a filter output mathematically to that of its input [9], but for our purposes this relation will not be of interest.)

In later chapters we will sometimes be concerned with the probability that noise x exceeds a given threshold x_t. From (1.71), this probability may be written

$$\Pr(x \geq x_t) = \int_{x_t}^{\infty} p(x)dx = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{x_t}^{\infty} \exp\{-\frac{(x-\overline{x})^2}{2\sigma^2}\}dx. \tag{1.79}$$

This integral cannot be evaluated analytically but may be re-expressed as

$$\Pr(x \geq x_t) = Q(\frac{x_t - \overline{x}}{\sigma}) \tag{1.80}$$

where $Q(x)$ is the so-called Q function, defined as

$$Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_{x}^{\infty} e^{-\frac{y^2}{2}} dy. \tag{1.81}$$

This standard function is depicted in Fig. 1.21 along with two analytical approximations. It vanishes exponentially beyond $x \simeq 2$. In this region $Q(x) \simeq (2\pi x^2)^{-1/2} \exp\{-x^2/2\}$ to good approximation.

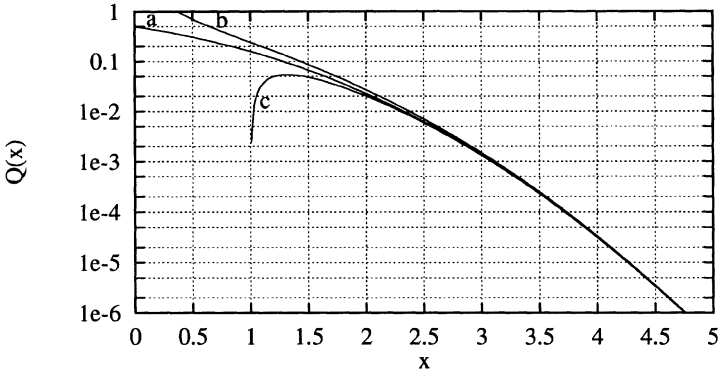


Figure 1.21: The Q-function (a) and two analytical approximations: $(2\pi x^2)^{-1/2}\exp\{-x^2/2\}$ (b) and $(2\pi x^2)^{-1/2}(1-x^{-2})\exp\{-x^2/2\}$ (c).

1.5 AVERAGES

In later chapters we will see that the optimum performances of various data receivers are governed by certain spectral averages. In the present section we introduce these averages in a general setting and discuss some of their properties. By way of introduction we first consider averages of a sequence of N positive numbers $a_1, \ldots, a_N$ (Table 1.2). The arithmetic average A is the most common one and if we talk about an average without any further specification then we will mean A. Denote by $a_{\max}$ and $a_{\min}$ the largest and smallest number in the sequence, respectively. Then

$$a_{\max} \geq A \geq G \geq H \geq a_{\min} \tag{1.82}$$

with equality holding if and only if all numbers are equal ($a_1 = a_2 = \ldots = a_N$). Both G and H can become ill-defined if any of the numbers $a_1, \ldots, a_N$ is negative. The harmonic average H is heavily determined by $a_{\min}$.

Average	
Arithmetic	$A = \frac{1}{N}[a_1 + a_2 + \ldots + a_N]$
Geometric	$G = \sqrt[N]{a_1 a_2 \ldots a_N} = \exp\{\frac{1}{N}[\ln a_1 + \ldots + \ln a_N]\}$
Harmonic	$H = \{\frac{1}{N}[\frac{1}{a_1} + \frac{1}{a_2} + \ldots + \frac{1}{a_N}]\}^{-1}$

Table 1.2: Three averages of a sequence of N numbers $a_1, a_2, \ldots, a_N$.

Example 1.9 Let $N = 2$, $a_1 = 1$ and $a_2 = 9$. Then $a_{\max} = 9$, $A = 5$, $G = 3$, $H = 1.8$ and $a_{\min} = 1$. Indeed $a_{\max} > A > G > H > a_{\min}$, while H is not much larger than $a_{\min}$.

The definitions of Table 1.2 are readily extended to functions of frequency. Let $F(e^{j\omega T})$ be an arbitrary real-valued and non-negative function of ω. This function is periodic in ω and the fundamental interval has length $\omega_0 \triangleq 2\pi/T$. Let us assume that we sample F at N uniformly spaced frequencies throughout this interval. The arithmetic average of these samples is an unweighted sum and in the limit as $N \to \infty$ becomes an integral

$$\overline{F}^A \triangleq \frac{1}{\omega_0}\int_{-\omega_0/2}^{\omega_0/2} F(e^{j\omega T})d\omega \qquad \text{(arithmetic average).} \tag{1.83}$$

Along the same lines we arrive at the other two definitions:

$$\overline{F}^G \triangleq \exp\{\frac{1}{\omega_0}\int_{-\omega_0/2}^{\omega_0/2} \ln F(e^{j\omega T})d\omega\} \qquad \text{(geometric average)} \tag{1.84}$$

and

$$\overline{F}^H \triangleq [\frac{1}{\omega_0}\int_{-\omega_0/2}^{\omega_0/2} F^{-1}(e^{j\omega T})d\omega]^{-1} \qquad \text{(harmonic average).} \tag{1.85}$$

Again the arithmetic average is the most common one and for this reason we will usually write $\overline{F}$ where we mean $\overline{F}^A$. Denote by $F_{\max}$ and $F_{\min}$ the maximum and minimum value of $F(e^{j\omega T})$, respectively. Then

$$F_{\max} \geq \overline{F} \geq \overline{F}^G \geq \overline{F}^H \geq F_{\min} \tag{1.86}$$

with equality holding if and only if F is independent of frequency. Just as for discrete-time sequences, $\overline{F}^H$ is heavily determined by $F_{\min}$. Unlike the discrete-time case, $\overline{F}^G$ can be strictly positive even when F vanishes at one or more frequencies.

Example 1.10 Let $F(e^{j\omega T}) = 4\sin^2(\omega T)$. Then the above averages are readily expressed in terms of standard integrals and it is found that $\overline{F} = 2$, $\overline{F}^G = 1$ and $\overline{F}^H = 0$. These averages are depicted along with F in Fig. 1.22.

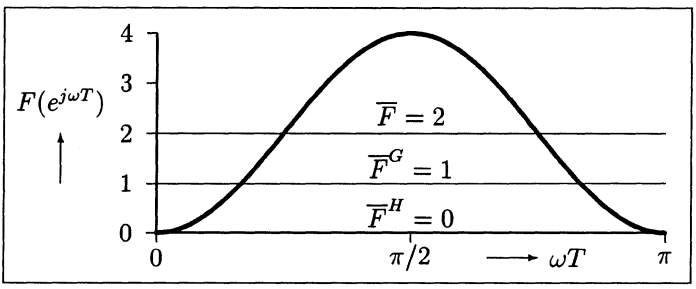


Figure 1.22: The function $F(e^{j\omega T}) = 4\sin^2(\omega T)$ and three averages.

$\overline{F}$ is a factor of 2 smaller than $F_{\max}$. $\overline{F}^G$ is distinctly nonzero even though F vanishes at both $\omega T = 0$ and $\omega T = \pi$. $\overline{F}^H$ coincides with $F_{\min}$.

Conceptually, $\overline{F}$, $\overline{F}^G$ and $\overline{F}^H$ are driven apart from each other by 'nonflatness' of F. If F does not depend much on frequency then all averages are more or less equal. Conversely, $\overline{F}^H$ drops far below $\overline{F}$ whenever F exhibits deep depressions or spectral zeros. This is much less true for $\overline{F}^G$. In fact, for functions F of interest to us, $\overline{F}^G$ is rarely if ever less than half of $\overline{F}$.

1.6 DISCRETE-TIME MINIMUM-PHASE FUNCTIONS

Minimum-phase functions are of basic importance to various receiver types, including the decision feedback equalizer (Chapter 6) and certain types of Viterbi detector (Chapter 7). In this section we discuss and exemplify various properties of these functions. Attention is restricted to those properties that are of interest to later chapters. More detailed expositions may be found in [6, sec. 7.2] and [10, ch. 2].

We first give a definition. Consider a discrete-time filter with impulse response g_k and denote the z-transform of g_k by $G(z)$. The filter is said to have minimum phase when $G(z)$ has its poles and zeros all inside the unit circle. We call g_k a minimum-phase function if $G(z)$ meets this condition. Any minimum-phase filter is causal and stable. A direct implication is that $g_k = 0$ for $k < 0$.

Inverse The inverse $G^{-1}(z)$ of $G(z)$ also has all poles and zeros inside the unit circle. Thus $G^{-1}(z)$ characterizes a causal, stable and minimum-phase filter. It follows that the action of a minimum-phase filter can be fully undone by a second minimum-phase filter. Since $G^{-1}(z)G(z) = 1$, the cascade of both filters will not introduce any delay. This is unlike non-minimum-phase filters, whose effect can only be undone at the cost of a nonzero overall delay.

Energy concentration Consider an arbitrary stable and causal filter with impulse response f_k and transfer function $F(e^{j\omega T})$. Let g_k be the minimum-phase function with the same amplitude characteristic as F, i.e. $|G(e^{j\omega T})| = |F(e^{j\omega T})|$ for all ω. (We will see that a minimum-phase function is fully determined by its amplitude characteristics and therefore there is only one such function g_k.) Then for any sequence e_k and any K

$$\sum_{k=0}^{K}(e * g)_k^2 \geq \sum_{k=0}^{K}(e * f)_k^2. \tag{1.87}$$

This property states, in essence, that signal energy is 'transported' with the smallest possible delay across a filter with given amplitude characteristic by choosing the filter to have minimum phase. For this reason minimum-phase filters are sometimes called *minimum-delay* filters. In the particular event that $e_k = \delta_k$ we find that

$$\sum_{k=0}^{K} g_k^2 \leq \sum_{k=0}^{K} f_k^2. \tag{1.88}$$

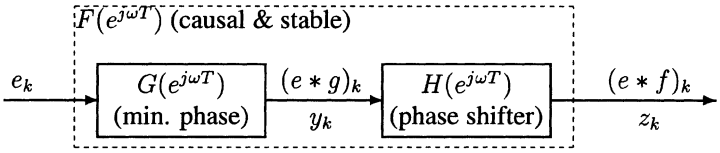


Figure 1.23: Any causal and stable filter with transfer function $F(e^{j\omega T})$ may be thought of as the cascade of a minimum-phase filter with transfer function $G(e^{j\omega T})$ and a causal and stable phase-shifter with transfer function $H(e^{j\omega T})$.

Thus, among all causal functions with a prescribed amplitude characteristic, the minimum-phase function has its energy maximally concentrated near the time origin.

The proof of (1.87) is quite instructive. Let h_k be the impulse response of a filter with transfer function $H(e^{j\omega T}) \triangleq F(e^{j\omega T})/G(e^{j\omega T})$ for all ω. This filter has the following properties:

1. it is *stable* (the reason is that its poles are determined by the poles of $F(z)$ and the zeros of $G(z)$, which are all within the unit circle);
2. it is *causal* (see Problem 1.19);
3. it is an *all-pass* filter or *phase shifter*, i.e. $|H(e^{j\omega T})| = 1$ for all ω.

This justifies the model of Fig. 1.23. Let y_k be a sequence with energy $E_y = \sum_{k=-\infty}^{\infty} y_k^2$. By Parseval's relation,

$$E_y = \frac{1}{\omega_0} \int_{-\omega_0/2}^{\omega_0/2} |Y(e^{j\omega T})|^2 d\omega \tag{1.89}$$

where $\omega_0 = 2\pi/T$ and where $Y(e^{j\omega T})$ is the Fourier transform of y_k. When y_k is applied to a phase shifter with transfer function $H(e^{j\omega T})$, the output sequence z_k has energy

$$E_z = \frac{1}{\omega_0} \int_{-\omega_0/2}^{\omega_0/2} |Y(e^{j\omega T})H(e^{j\omega T})|^2 d\omega = E_y, \tag{1.90}$$

i.e. the phase shifter preserves all signal energy. Now assume that y_k vanishes outside the interval $\{0, \ldots, K\}$. Then $E_y = \sum_{k=0}^{K} y_k^2$. By causality of the phase shifting network, z_k vanishes for $k < 0$. Conversely, z_k is not necessarily zero for $k > K$. Hence $E_z \leq \sum_{k=0}^{K} z_k^2$. We arrive at (1.87) upon identifying y_k and z_k as $(e * g)_k$ and $(e * f)_k$, respectively. □

Example 1.11 Assume that $F(z) = (2 + z^{-1})(2z^{-1} + 1) = 2 + 5z^{-1} + 2z^{-2}$. This function has two zeros $z_1 = -0.5$ and $z_2 = 2$. Clearly z_2 is outside the unit circle and therefore f_k is not a minimum-phase function. The decomposition according to Fig. 1.23 is $F(z) = G(z)H(z)$ with

$$G(z) = (2 + z^{-1})^2 = 4 + 4z^{-1} + z^{-2} \quad \text{and} \quad H(z) = \frac{2z^{-1} + 1}{2 + z^{-1}}. \tag{1.91}$$

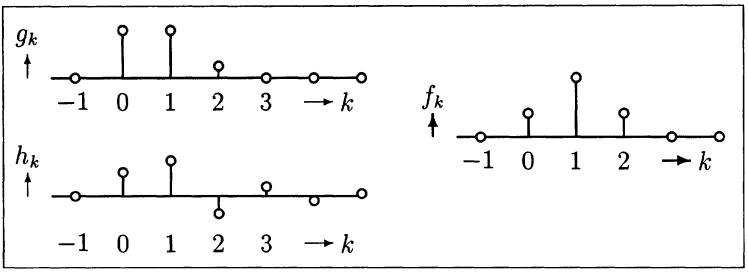


Figure 1.24: Impulse responses g_k, h_k and f_k for $F(z) = (2 + z^{-1})(2z^{-1} + 1)$.

Corresponding impulse responses g_k, h_k and f_k are depicted in Fig. 1.24.

The function $G(z)$ has two identical zeros $z_{1,2} = -0.5$. Both are within the unit circle and therefore g_k is a minimum-phase function. Furthermore, since $H(z)$ corresponds to a phase shifter, g_k and f_k have the same amplitude characteristics. Evidently g_k has its energy more concentrated near the time origin ($k = 0$) than f_k.

Minimum-phase property Consider an arbitrary causal and stable filter with transfer function $F(e^{j\omega T})$ and phase characteristic $\phi_f(e^{j\omega T})$. The *phase lag* of the filter is defined as the negative of the phase, i.e. as $-\phi_f(e^{j\omega T})$. Problem 1.20 reveals that the phase lag of the phase shifter in Fig. 1.23 is non-negative for all ω. The implication is that $-\phi_g(e^{j\omega T}) \leq -\phi_f(e^{j\omega T})$, i.e. that the minimum-phase filter has the smallest possible phase lag among all filters with a prescribed amplitude characteristic. It is this property that justifies the adjective 'minimum-phase'.

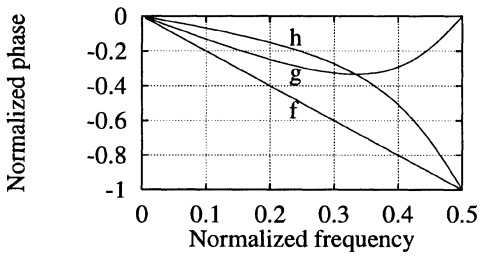


Figure 1.25: Normalized phase $\phi(e^{j\omega T})/\pi$ versus normalized frequency $\omega T/(2\pi)$ for the functions f_k, g_k and h_k of Example 1.11.

Example 1.12 Fig. 1.25 depicts the phase characteristics of the functions f_k, g_k and h_k of Example 1.11. Indeed $\phi_h(e^{j\omega T}) < 0$ for all ω and $-\phi_g(e^{j\omega T}) < -\phi_f(e^{j\omega T})$ for all ω.

Relation between amplitude and phase characteristics The amplitude characteristic of a minimum-phase filter fully determines the phase characteristic and vice versa.

More in particular, if $G(e^{j\omega T})$ is the transfer function of a minimum-phase filter, then the *attenuation characteristic*, defined as $X(e^{j\omega T}) \triangleq \ln |G(e^{j\omega T})|$, is the Hilbert transform of the phase characteristic $\phi_g(e^{j\omega T})$. Appendix 1C provides a proof of this property and outlines and efficient numerical procedure for computing $\phi(e^{j\omega T})$ and $G(e^{j\omega T})$ when $|G(e^{j\omega T})|$ is prescribed.

Recursive computation of a minimum-phase function with prescribed amplitude characteristics Let $X(e^{j\omega T}) = \ln |G(e^{j\omega T})|$ be the prescribed attenuation characteristic of a minimum-phase function g_k with Fourier transform $G(e^{j\omega T})$, and denote the inverse Fourier tranform of $X(e^{j\omega T})$ by x_k. Then g_k obeys the recursion

$$g_0 = e^{x_0} \tag{1.92}$$

and

$$g_k = \frac{2}{k}\sum_{i=0}^{k-1}(k-i)x_{k-i}g_i \text{ for } k \geq 1. \tag{1.93}$$

Proof: See [2, p. 161]. □

Averages Since x_k is the inverse Fourier transform of $\ln |G(e^{j\omega T})|$ we may write (1.92) as

$$g_0 = \exp\{\frac{1}{\omega_0}\int_{-\omega_0/2}^{\omega_0/2} \ln |G(e^{j\omega T})| d\omega\}. \tag{1.94}$$

This shows that g_0 is the *geometric average* of $|G(e^{j\omega T})|$. In the nomenclature of Section 1.5 we may write $g_0 = \overline{|G|}^G$. A direct implication is that $g_0^2 = \overline{|G|^2}^G$, i.e. the energy of the first sample of g_k is the geometric average of $|G(e^{j\omega T})|^2$. The total energy $E_g \triangleq \sum_{k=0}^{\infty} g_k^2$ cannot be smaller than g_0^2 and may be expressed in terms of $|G(e^{j\omega T})|^2$ with the aid of Parseval's identity. It follows that

$$E_g = \overline{|G|^2}^A \geq \overline{|G|^2}^G \tag{1.95}$$

where $\overline{|G|^2}^A$ denotes the arithmetic average of $|G(e^{j\omega T})|^2$ as defined in Section 1.5. The equality sign in (1.95) applies if and only if $|G|$ is independent of ω, i.e. if $G(e^{j\omega T})$ has no amplitude distortion. The ratio $g_0^2/E_g = \overline{|G|^2}^A/\overline{|G|^2}^G$ is the fraction of all energy of g_k that is concentrated in the first sample g_0. We will see in Chapter 6 that the performance of the decision feedback equalizer is governed by a ratio of this kind.

Example 1.13 Consider the function $G(z) = (2+z^{-1})^2$ of Example 1.11. The averages $\overline{|G|^2}^G$ and $\overline{|G|^2}^A$ may be computed in two ways.

1. We can perform frequency-domain integration as in (1.83) and (1.84). For the function $G(z)$ at hand both averages can be recast in terms of standard integrals and with some effort is found that $\overline{|G|^2}^G = 16$ and $\overline{|G|^2}^A = 33$.

2. We can exploit the fact that $G(z)$ has minimum phase, whence $\overline{|G|^2}^G = g_0^2$. Furthermore, by Parseval's identity, $\overline{|G|^2}^A = E_g$. From Fig. 1.24 we see directly that $\overline{|G|^2}^G = 4^2 = 16$ and $\overline{|G|^2}^A = 4^2 + 4^2 + 1^2 = 33$.

1.7 LAGRANGE MULTIPLIERS

In the design of data transmission systems one often encounters optimization problems in which a function of multiple variables is to be optimized subject to one or more constraints on the variables. These constrained optimization problems can usually be solved via the method of Lagrange multipliers. This method is named after its discoverer, the French mathematician Joseph Louis Lagrange (1736-1813). We shall merely describe and exemplify the approach without deriving it formally. The interested reader is referred to standard text books on engineering mathematics (see, for example, [12, Section 3.10]) for a more thorough treatment.

We first consider the simplest possible situation to which the method is applicable, namely a function $f(x, y)$ of two variables x and y, which is to be optimized subject to a single constraint $g(x, y) = 0$. To accommodate this constraint we could, in principle, first try to solve the constraint equation for y in terms of x. This would yield a function $y = \phi(x)$, and we could then maximize or minimize the single-variable function $f(x, \phi(x))$ by locating the zeros of its derivative. Unfortunately, it may not be practical nor even doable to solve the constraint equation explicitly for y in terms of x. The method of Lagrange multipliers gets around this problem via a so-called *Lagrangian* L that combines the function and the constraint according to

$$L \triangleq f(x, y) + \lambda g(x, y). \tag{1.96}$$

The scaling factor λ, which is as yet undetermined, is said to be a *Lagrange multiplier*. The crux of the method of Lagrange multipliers is the fact that the constrained optimum of f is a stationary point of L [12, Section 3.10]. Thus, in order to identify this optimum we differentiate L with respect to x and y and require both derivatives to vanish. This yields the two conditions

$$f_x(x, y) + \lambda g_x(x, y) = 0 \text{ and } f_y(x, y) + \lambda g_y(x, y) = 0 \tag{1.97}$$

where the subscripts $_x$ and $_y$ denote partial derivatives with respect to x and y, respectively. The constraint $g(x, y) = 0$ provides us with a third condition, and we can attempt to solve the three conditions for the three unknowns x, y and λ. The points (x, y) that we find are the only possible locations for the extrema of f subject to the constraint $g(x, y) = 0$. The associated value of λ is usually not of much interest. In order to spot the maximum and minimum values of f we calculate the values of f at each of the solution points.

The method just sketched will work properly provided only that g_x and g_y do not vanish simultaneously at the constrained optimum. In this rare event, the graph $g(x, y)$ usually fails to be a smooth curve at the location (x, y) of the optimum.

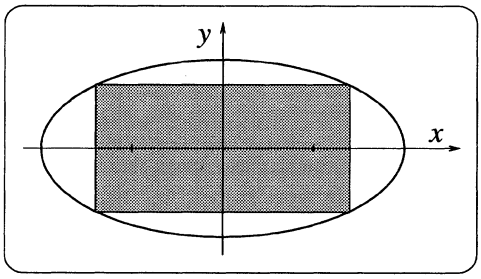


Figure 1.26: Ellipse that encloses a rectangle.

Example 1.14 Consider the ellipse $x^2 + 4y^2 = 4$ as shown in Fig. 1.26. We wish to fit a rectangle within the ellipse in such a manner that the area of the rectangle is maximized. For simplicity we assume that the sides of the rectangle are parallel to the coordinate axes. The area is then given by $f(x, y) = 4xy$ where (x, y) denotes the upper right corner $(x, y \geq 0)$. This corner must evidently touch the ellipse so that we have the constraint

$$g(x, y) = x^2 + 4y^2 - 4 = 0. \tag{1.98}$$

The Lagrangian $L = f + \lambda g$ has partial derivatives $L_x = 4y + 2\lambda x$ and $L_y = 4x + 8\lambda y$. These derivatives will vanish if and only if

$$2y + \lambda x = 0 \text{ and } x + 2\lambda y = 0. \tag{1.99}$$

It is clear that neither $x = 0$ nor $y = 0$ gives the maximum area. Hence we may divide the first equation by x and the second one by $2y$ to obtain

$$\frac{2y}{x} = \frac{x}{2y} = -\lambda. \tag{1.100}$$

We forget λ, and see that $x^2 = 4y^2$. Together with (1.98) and the fact that x and y must both be positive this implies that $x = \sqrt{2}$ and $y = 1/\sqrt{2}$. The corresponding area $f(x, y)$ amounts to 4.

It is worth noting that the function $f(x, y) = 4xy$ has a maximum of 4 at both $(\sqrt{2}, 1/\sqrt{2})$ and $(-\sqrt{2}, -1/\sqrt{2})$, and a minimum of -4 at $(\sqrt{2}, -1/\sqrt{2})$ and $(-\sqrt{2}, 1/\sqrt{2})$. The Lagrange multiplier method actually locates all four of these constrained optima.

It is straightforward to extend the technique to functions of more than two variables. For a function $f(x_1, \ldots, x_N)$ of N variables and a constraint $g(x_1, \ldots, x_N) = 0$ we obtain a Lagrangian $L = f + \lambda g$ whose partial derivatives with respect to $x_1, \ldots, x_N$ must all vanish. Together with the constraint $g(x_1, \ldots, x_N) = 0$ this yields $N + 1$ conditions that can be used to find the $N + 1$ unknowns λ and $x_1, \ldots, x_N$. This technique will work properly provided only that the partial derivatives of g with respect to $x_1, \ldots, x_N$ do not vanish simultaneously at the location of the constrained optimum.

Accommodation of multiple constraints is possible by increasing the number of multipliers. For example, in order to optimize a function $f(x_1, \ldots, x_N)$ subject to

two constraints $g(x_1, \ldots, x_N) = 0$ and $h(x_1, \ldots, x_N) = 0$ we form the Lagrangian $L = f + \lambda_1 g + \lambda_2 h$ where λ_1 and λ_2 are Lagrange multipliers. Again we require the derivatives of L with respect to the N variables to vanish. Together with the two constraints this yields $N+2$ conditions that can be used to find the $N+2$ unknowns λ_1, λ_2 and $x_1, \ldots, x_N$. This method will work properly provided that the gradients of g and h with respect to the variables do not vanish simultaneously at the location of the constrained optimum, and that the the gradients are not linearly dependent at this location.

1.8 FURTHER READING

The above survey has covered most mathematical concepts that are basic to subsequent chapters but is neither rigorous nor complete. For brevity we have not addressed various concepts that are relevant to only some subjects in later chapters. These will either be discussed where appropriate or assumed as prior knowledge. Those wishing to pursue the subject matter of this chapter in greater detail will find a wealth of material in the literature [1], [3], [4], [6], [7], [2], [9], [10], [8, Chapter 1], [15, Chapter 2,3] [16].

APPENDIX 1A

The Fourier transform for continous-time signals

This appendix provides an overview of properties of the Fourier transform for continuous-time signals (FTC). Attention is restricted to aperiodic signals, which have a continuous spectrum. Periodic signals, by comparsion, have a discrete spectrum that may be described by a Fourier series expansion (see, for example, [3, 10]). Signals of this type are of restricted interest to digital transmission and for this reason we do not treat the Fourier series here.

Throughout this appendix we express frequencies in Hertz and denote these by the variable f. The angular frequency ω is related to f by $\omega = 2\pi f$, and the normalized frequency Ω of Section 1.2 is related to f by $\Omega = fT$ where T denotes the sampling interval.

1A.1 Definitions

Let $x(t)$ be a continuous aperiodic signal of finite energy where t denotes time in seconds. Then the Fourier transform $X(f)$ of $x(t)$ is well-defined and is given by

$$X(f) \triangleq \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} dt. \quad \text{(FTC)} \tag{1A.1}$$

The inverse transform is given by

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} df. \quad \text{(IFTC)} \tag{1A.2}$$

We shall say that $x(t)$ and $X(f)$ form a Fourier pair. This is denoted $x(t) \circ\!\!-\!\!\circ X(f)$.

Let $x_1(t)$ and $x_2(t)$ be arbitrary non-periodic continuous-time signals with Fourier integrals $X_1(f)$ and $X_2(f)$. We denote by $(x_1 * x_2)(t)$ the convolution of $x_1(t)$ and $x_2(t)$, i.e.

$$(x_1 * x_2)(t) \triangleq \int_{-\infty}^{\infty} x_1(\tau)x_2(t-\tau)d\tau. \tag{1A.3}$$

Then

$$(x_1 * x_2)(t) \circ\!\!-\!\!\circ X_1(f)X_2(f). \tag{1A.4}$$

Convolution is *commutative* $[(x_1 * x_2)(t) = (x_1 * x_2)(t)]$, *associative* $[(x_0 * (x_1 * x_2))(t) = ((x_0 * x_1) * x_2)(t)]$, and *distributive* $[(x_0 * (x_1 + x_2))(t) = (x_0 * x_1)(t) + (x_0 * x_2)(t)]$.

1A.2 Symmetry properties of the FTC

$x(t)$	$X(f)$
real-valued	real part even, imaginary part odd
real-valued and even	real-valued and even
real-valued and odd	imaginary and odd
imaginary	real part odd, imaginary part even
imaginary and even	imaginary and even
imaginary and odd	real-valued and odd
real and imaginary parts even	real and imaginary parts even
real and imaginary parts odd	real and imaginary parts odd
real part even, imaginary part odd	real-valued
real part odd, imaginary part even	imaginary

Note: A function is said to be even when it is symmetric about the origin and odd when it is antisymmetric.

1A.3 Other properties of the FTC

Linearity	$ax_1(t) + bx_2(t)$	$\circ\!\!-\!\!\circ$ $aX_1(f) + bX_2(f)$
Time reversal	$x(-t)$	$\circ\!\!-\!\!\circ$ $X(-f)$
Conjugation	$x^*(t)$	$\circ\!\!-\!\!\circ$ $X^*(-f)$
	$x^*(-t)$	$\circ\!\!-\!\!\circ$ $X^*(f)$
Reciprocity	$X(t)$	$\circ\!\!-\!\!\circ$ $x(-f)$
Time shift	$x(t-t_0)$	$\circ\!\!-\!\!\circ$ $X(f)e^{-j2\pi f t_0}$
Frequency shift	$x(t)e^{j2\pi f_0 t}$	$\circ\!\!-\!\!\circ$ $X(f-f_0)$
Time/frequency scaling	$x(at)$	$\circ\!\!-\!\!\circ$ $\frac{1}{\lvert a\rvert}X(\frac{f}{a})$

Convolution	$(x_1 * x_2)(t) \circ\!\!-\!\!\circ X_1(f)X_2(f)$
Multiplication	$x_1(t)x_2(t) \circ\!\!-\!\!\circ (X_1 * X_2)(f)$
Differentiation	$\frac{d^n}{dt^n}x(t) \circ\!\!-\!\!\circ (j2\pi f)^n X(f)$
Parseval's identity	$\int_{-\infty}^{\infty} x_1(t)x_2^*(t)dt = \int_{-\infty}^{\infty} X_1(f)X_2^*(f)df$
	$\int_{-\infty}^{\infty} \lvert x(t)\rvert^2 dt = \int_{-\infty}^{\infty} \lvert X(f)\rvert^2 df$
Poisson's sum formula	$\sum_{k=-\infty}^{\infty} x(t+kT) = \frac{1}{T}\sum_{n=-\infty}^{\infty} X(\frac{n}{T})e^{j2\pi nt/T}$

1A.4 Some basic FTC pairs

Function	$x(t)$	$X(f)$
Delta function	$\delta(t)$	1
Sequence of delta pulses	$\sum_{k=-\infty}^{\infty} \delta(t-kT)$	$\frac{1}{T}\sum_{n=-\infty}^{\infty} \delta(f-\frac{n}{T})$
Phasor	$e^{j2\pi f_0 t+\theta}$	$e^{j\theta}\delta(f-f_0)$
Unit step	$\begin{cases} 0, & t \le 0, \\ 1, & t \ge 0 \end{cases}$	$\frac{1}{j2\pi f} + \frac{1}{2}\delta(f)$
Rectangular pulse	$\begin{cases} 1, & \lvert t\rvert \le \frac{T}{2}, \\ 0, & \text{else} \end{cases}$	$T\frac{\sin \pi fT}{\pi fT}$
Triangular pulse	$\begin{cases} 1-\frac{\lvert t\rvert}{T}, & \lvert t\rvert \le T, \\ 0, & \text{else} \end{cases}$	$T\frac{\sin^2 \pi fT}{(\pi fT)^2}$
Causal exponential pulse ($\mathrm{Re}(a) > 0$)	$\begin{cases} 0, & t < 0, \\ \frac{t^{n-1}}{(n-1)!}e^{-at}, & t \ge 0 \end{cases}$	$\frac{1}{(a+j2\pi f)^n}$
Lorentzian pulse ($t_0 > 0$)	$\frac{1}{1+(\frac{2t}{t_{50}})^2}$	$\frac{\pi t_{50}}{2}e^{-\pi\lvert f\rvert t_{50}}$
Gaussian pulse ($\mathrm{Re}(a) > 0$)	$\sqrt{\frac{a}{\pi}}e^{-at^2}$	$e^{-(\pi f)^2/a}$

APPENDIX 1B

The Fourier transform for discrete-time signals

This appendix provides an overview of properties of the Fourier transform for discrete-time signals (FTD). Attention is restricted to aperiodic signals, which have a continuous spectrum. Periodic signals, by comparison, have a discrete spectrum that may be described by a Fourier series expansion (see, for example, [3, 10]). Signals of this type are of restricted interest to digital transmission and for this reason we do not treat the Fourier series here.

Throughout this appendix we use a normalized measure of frequency Ω. Normalization is such that $\Omega = 1$ corresponds to the sampling rate $1/T$. The frequency f (in Hertz) is related to Ω by $f = \Omega/T$, and the angular frequency ω by $\omega = 2\pi\Omega/T$.

1B.1 Definitions

Let x_k be an non-periodic sequence where k denotes the sample index. The Fourier transform $X(e^{j2\pi\Omega})$ of x_k is defined by

$$X(e^{j2\pi\Omega}) \triangleq \sum_{k=-\infty}^{\infty} x_k e^{-j2\pi\Omega k} \quad \text{(FTD)}. \tag{1B.1}$$

If x_k is absolutely summable, i.e. if $\sum_{k=-\infty}^{\infty} |x_k| < \infty$, then $X(e^{j2\pi\Omega})$ is a finite and continuous function of Ω. It can be observed that $X(e^{j2\pi\Omega})$ is periodic in Ω with period 1. For this reason x_k is fully determined by the behaviour of $X(e^{j2\pi\Omega})$ across an interval of length 1. The *fundamental interval* is the interval of length 1 that is actually used to reconstruct x_k. In the sequel we shall use the interval $[-0.5, 0.5]$ for this purpose. For this interval the inverse Fourier transform is given by

$$x_k = \int_{-0.5}^{0.5} X(e^{j2\pi\Omega}) e^{j2\pi\Omega k} d\Omega. \quad \text{(IFTD)} \tag{1B.2}$$

We shall say that x_k and $X(e^{j2\pi\Omega})$ form a Fourier pair. This is denoted $x_k \circ\!\!-\!\!\circ X(e^{j2\pi\Omega})$.

Let x_k^a and x_k^b be arbitrary sequences with Fourier transforms $X^a(e^{j2\pi\Omega})$ and $X^b(e^{j2\pi\Omega})$. We denote the convolution of x_k^a and x_k^b by $(x^a * x^b)_k$, i.e.

$$(x^a * x^b)_k \triangleq \sum_{i=-\infty}^{\infty} x_i^a x_{k-i}^b. \tag{1B.3}$$

Convolution translates spectrally into a multiplication, i.e.

$$(x^a * x^b)_k \circ\!\!-\!\!\circ X^a(e^{j2\pi\Omega}) X^b(e^{j2\pi\Omega}). \tag{1B.4}$$

Convolution is *commutative* $[(x^a * x^b)_k = (x^a * x^b)_k]$, *associative* $[(x^a * (x^b * x^c))_k = ((x^a * x^b) * x^c)_k]$, and *distributive* $[(x^a * (x^b + x^c))_k = (x^a * x^b)_k + (x^a * x^c)_k]$.

1B.2 Symmetry properties of the FTD

These are fullly identical to those for the FTC (see Section 1A.2).

1B.3 Other properties of the FTD

Linearity	$\alpha x_k^a + \beta x_k^b \circ\!\!-\!\!\circ\ \alpha X^a(e^{j2\pi\Omega}) + \beta X^b(e^{j2\pi\Omega})$
Time reversal	$x_{-k} \circ\!\!-\!\!\circ\ X(e^{-j2\pi\Omega})$
Conjugation	$x_k^* \circ\!\!-\!\!\circ\ X^*(e^{-j2\pi\Omega})$
	$x_{-k}^* \circ\!\!-\!\!\circ\ X^*(e^{j2\pi\Omega})$
Time shift	$x_{k-k_0} \circ\!\!-\!\!\circ\ X(e^{j2\pi\Omega})e^{-j2\pi k_0\Omega}$
Frequency shift	$x_k e^{j2\pi k\Omega_0} \circ\!\!-\!\!\circ\ X(e^{j2\pi(\Omega-\Omega_0)})$
Convolution	$(x^a * x^b)_k \circ\!\!-\!\!\circ\ X^a(e^{j2\pi\Omega})X^b(e^{j2\pi\Omega})$
Multiplication	$x_k^a x_k^b \circ\!\!-\!\!\circ\ \frac{1}{2\pi}\int_{-0.5}^{0.5} X^a(e^{j2\pi\mu})X^b(e^{j2\pi(\Omega-\mu)})d\mu$
Parseval's identity	$\sum_{k=-\infty}^{\infty} x_k^a x_k^{b*} = \int_{-0.5}^{0.5} X^{a*}(e^{j2\pi\Omega})X^b(e^{j2\pi\Omega})d\Omega$
	$\sum_{k=-\infty}^{\infty} \lvert x_k\rvert^2 = \int_{-0.5}^{0.5} \lvert X(e^{j2\pi\Omega})\rvert^2 d\Omega$

1B.4 Some basic FTD pairs

Function	x_k	$X(e^{j2\pi\Omega})$
Delta function	δ_k	1
Sequence of delta pulses	$\sum_{m=-\infty}^{\infty} \delta_{k-mN}$	$\frac{1}{NT}\sum_{n=-\infty}^{\infty} \delta(\Omega - \frac{n}{N})$
Unit step	$\begin{cases} 0, & k < 0, \\ 1, & k \geq 0 \end{cases}$	$\frac{1}{1-e^{-j2\pi\Omega}} + \frac{1}{2}\sum_{k=-\infty}^{\infty} \delta(\Omega - k)$
Rectangular pulse	$\begin{cases} 1, & \lvert k\rvert \leq N, \\ 0, & \text{else} \end{cases}$	$\frac{\sin \pi\Omega(2N+1)}{\sin \pi\Omega}$
Sinc pulse	$\frac{\sin 2\pi n\Omega_1}{2\pi n\Omega_1}$	$\begin{cases} \frac{1}{2\Omega_1}, & \lvert\Omega\rvert \leq \Omega_1, \\ 0, & \Omega_1 \leq \lvert\Omega\rvert \leq 0.5. \end{cases}$

Causal power series ($\lvert a\rvert < 1$)	$\begin{cases} 0, & k < 0, \\ a^k, & k \geq 0 \end{cases}$	$\dfrac{1}{1 - ae^{-j2\pi\Omega}}$
	$\begin{cases} 0, & k < 0, \\ (k+1)a^k, & k \geq 0 \end{cases}$	$\dfrac{1}{(1 - ae^{-j2\pi\Omega})^2}$
	$\begin{cases} 0, & k < 0, \\ \frac{(k+m-1)!}{k!(m-1)!}a^k, & k \geq 0 \end{cases}$	$\dfrac{1}{(1 - ae^{-j2\pi\Omega})^m}$
Damped sinusoid ($\lvert a\rvert < 1$)	$\begin{cases} 0, & k < 0, \\ a^k \sin(k\zeta + \psi), & k \geq 0 \end{cases}$	$\dfrac{\sin(\psi)e^{j4\pi\Omega} + a\sin(\zeta - \psi)e^{j2\pi\Omega}}{e^{j4\pi\Omega} - 2a\cos(\zeta)e^{j2\pi\Omega} + a^2}$

Note: The Kronecker and Dirac delta functions are denoted by δ_k and $\delta(.)$, respectively.

APPENDIX 1C

Spectral characteristics of minimum-phase functions

Consider a minimum-phase function g_k with Fourier transform $G(e^{j\omega T})$, attenuation characteristic $X(e^{j\omega T}) \triangleq \ln|G(e^{j\omega T})|$ and phase characteristic $\phi_g(e^{j\omega T})$. We shall show that $X(e^{j\omega T})$ and $\phi_g(e^{j\omega T})$ are a Hilbert transform pair and that $\phi_g(e^{j\omega T})$ may be computed from the inverse Fourier transform x_k of $X(e^{j\omega T})$ according to

$$\phi_g(e^{j\omega T}) = -\sum_{k=1}^{\infty} 2x_k \sin(k\omega T). \tag{1C.1}$$

Proof: For g_k to be a minimum-phase function, $G(z)$ should have its poles and zeros all inside the unit circle. The function $\ln G(z)$ must, therefore, be analytic in the region $|z| \geq 1$. Consequently, $\ln G(e^{j\omega T}) = X(e^{j\omega T}) + j\phi_g(e^{j\omega T})$ has a causal and well-behaved inverse Fourier transform γ_k, i.e. $\ln G(e^{j\omega T})$ may be written as

$$\ln G(e^{j\omega T}) = \sum_{k=0}^{\infty} \gamma_k e^{-jk\omega T}. \tag{1C.2}$$

The real and imaginary parts of this expression are

$$X(e^{j\omega T}) = \sum_{k=0}^{\infty} \gamma_k \cos k\omega T \tag{1C.3}$$

and

$$\phi_g(e^{j\omega T}) = -\sum_{k=1}^{\infty} \gamma_k \sin k\omega T. \tag{1C.4}$$

This verifies that $X(e^{j\omega T})$ and $\phi_g(e^{j\omega T})$ form a Hilbert transform pair. To see how $\phi_g(e^{j\omega T})$ may be computed from $X(e^{j\omega T})$, we note that $X(e^{j\omega T})$ is the logarithm of a real-valued, symmetric function of frequency and thereby itself real-valued and symmetric. Thus its inverse Fourier transform x_k is real-valued and symmetric, i.e. $x_{-k} = x_k \in \mathbb{R}$ for all k, so that

$$X(e^{j\omega T}) = x_0 + 2\sum_{k=0}^{\infty} x_k \cos k\omega T. \tag{1C.5}$$

By comparing (1C.3) and (1C.5) we conclude that $\gamma_0 = x_0$ and $\gamma_k = 2x_k$ for $k \geq 1$. Upon using this result in (1C.4) we obtain (1C.1). □

Numerical procedure for computing $G(e^{j\omega T})$ from $|G(e^{j\omega T})|$ We may regard $j\phi_g(e^{j\omega T})$ as the Fourier transform of an antisymmetric sequence y_k that is derived from x_k according to $y_k = -x_{-k}$ for $k < 0$, $y_0 = 0$, and $y_k = x_k$ for $k > 0$. This suggests the following computational procedure to determine a minimum-phase transfer function $G(e^{j\omega T})$ with prescribed amplitude characteristics $|G(e^{j\omega T})|$:

1. Compute $X(e^{j\omega T}) = \ln |G(e^{j\omega T})|$ at a set of discrete frequencies $\omega_i T = 2\pi i/N$ for $i = 0, 1, \ldots, N-1$.

2. Compute an approximation $\tilde{x}_k$ of x_k by taking the inverse FFT of $\{X(e^{j\omega_i T})\}$. For sufficiently large N this approximation will be very good.

3. Construct a signal

$$\tilde{y}_k = \begin{cases} -\tilde{x}_{-k} & \text{for } k < 0, \\ 0 & \text{for } k = 0, \text{ and} \\ \tilde{x}_k & \text{for } k > 0. \end{cases} \tag{1C.6}$$

4. Compute an approximation $j\tilde{\phi}_g(e^{j\omega_i T})$ of $j\phi_g(e^{j\omega T})$ at the frequencies $\omega_i T$ from $\tilde{y}_k$ by means of an FFT of length N.

5. An approximation $\tilde{G}(e^{j\omega_i T})$ of $G(e^{j\omega T})$ at the discrete frequencies $\omega_i T$ is formed as $\tilde{G}(e^{j\omega_i T}) = \exp[X(e^{j\omega_i T}) + j\tilde{\phi}_g(e^{j\omega_i T})]$.

PROBLEMS

1.1 Consider the first-order analog filter with transfer function $H(\omega) = 1/[1 + j\omega/\omega_c]$ where ω_c is a real-valud and non-negative constant. Determine the amplitude, phase and group delay characteristics of this filter. What type of filter is this? How can ω_c be interpreted? How many zeros and poles does the filter have and where are they located?

1.2 A second-order analog filter of particular interest is the *resonance circuit.* This circuit has transfer function

$$H(\omega) = \frac{1}{Q\omega_0} \frac{j\omega}{1 + \frac{j}{Q}\frac{\omega}{\omega_0} - (\frac{\omega}{\omega_0})^2} \tag{1P.1}$$

where Q is called the *quality factor* and ω_0 is called the *resonance frequency.* Determine the amplitude, phase and group delay characteristics of this circuit. How can ω_0 and Q be interpreted? How many zeros and poles does the filter have and where are they located?

1.3 Use the FTC of a sequence of Delta functions (see Section 1A.4) to prove Shannon's sampling theorem.

1.4 Reconsider Fig. 1.6 for the situation that sampling occurs at the instants $t_k = (k + \phi)T$ where ϕ is a fixed phase shift normalized in symbol intervals T. Generalize (1.28) to this situation.

1.5 Assume that the sampler in Fig. 1.6 is *oversampled* such that it closes at the instants $t = nT/L$, where L is an integer oversampling factor. Generalize (1.28) to this situation.

1.6 Show that the linear pulse modulator of Fig. 1.10 can equivalently be described as a cascade of two building blocks: 1) a discrete-time to continuous-time converter, and 2) a continuous-time filter with impulse response $h(t)$. What mathematical operation does the discrete-time-to-continuous-time converter perform?

1.7 Consider the cascade of a continuous-time T-spaced transversal filter with coefficients c_k and a sampler that closes at the instants $t = kT$. Show that the output of the cascade does not change if the order of both building blocks is reversed.

1.8 Prove (1.31).

1.9 Consider the cascade of Fig. 1.11. Show by means of examples that this cascade is, in general, not time-invariant when a) the sampler causes aliasing, and b) $C(\Omega)$ is not bandlimited to half of the sampling frequency.

1.10 Use Poisson's summation formula (see Section 1A.3) to show that

$$\sum_{k=-\infty}^{\infty} x_1(t_1 + kT)x_2(t_2 + kT) = \frac{1}{T} \sum_{n=-\infty}^{\infty} e^{j2\pi n t_1/T} \int_{-\infty}^{\infty} X_2(\mu)X_1(n-\mu)e^{j2\pi\mu(t_2-t_1)/T} d\mu. \tag{1P.2}$$

Subject to which condition is $X_2(\mu)X_1(n-\mu)$ zero irrespective of μ for all $n \neq 0$? How does (1P.2) simplify in this case?

1.11 Use the results of Problem 1.10 to derive a necessary and sufficient condition on the Fourier transforms of $y(t)$ and $c(t)$ for the cascade of Fig. 1.12 to be time-invariant.

1.12 Let N be an arbitrary integer and let

$$x_k^N \triangleq \begin{cases} x_{k/N}, & k = lN, l \text{ any integer}, \\ 0, & \text{otherwise}. \end{cases} \tag{1P.3}$$

Show that $x_k^N \circ\!\!-\!\!\circ X(e^{j2\pi N\Omega})$. This is the counterpart for discrete-time sequences of the time-frequency scaling property of the FTC (see Section 1A.3).

1.13 Prove (1.60).

1.14 Consider a linear pulse modulator with stationary random input a_k and symbol response $c(t) = 1$ for $|t| < T/2$ and 0 else.

1. Show that the mean $E[s(t)]$ of the modulator output $s(t)$ does not depend on t.
2. Determine the autocorrelation function $R_s(t, \tau)$ of $s(t)$ for the situation that a_k has flat power spectral density $\mathcal{A}e^{j\omega T} = 1$. Do this in the form of a two-dimensional picture that shows the altitude lines of $R_s(t, \tau)$ as a function of t and τ.
3. Determine the averaged autocorrelation function $R_s(\tau) = \frac{1}{T}\int_{-T/2}^{T/2} R_s(t, \tau)d\tau$ and the power spectral density $\mathcal{S}(\omega)$ of $s(t)$.

1.15 Consider the cascade of Fig. 1.19 and assume that a_k is stationary with autocorrelation function R_k^a. Determine the autocorrelation function of b_k.

1.16 In Exercise 1.13 we have determined the power spectrum of a sampled PAM process. In this problem we investigate to what extent this result depends on the cyclostationarity of the PAM process. To this end, assume for the sake of the argument that the output $s(t)$ of the modulator is stationary rather than cyclostationary, but that its PSD is still given by (1.66). Determine the PSD of the sampled process $b_k = s(kT)$ and compare this PSD with the one determined in Exercise 1.13.

1.17 Prove the properties of Table 1.1.

1.18 Consider a stationary random process $n(t)$ with mean $\overline{n}$ and autocorrelation function $R_n(\tau)$. Assume that a new process $x(t)$ is produced according to $x(t) = s(t)n(t)$ where $s(t)$ is a given deterministic function of finite duration and finite energy.

1. Is $x(t)$ stationary?
2. Determine the mean, the power and the autocorrelation function of $x(t)$.
3. Assume that a random variable z is obtained from $x(t)$ by integration across an infinite interval (i.e. $z = \int_{-\infty}^{\infty} x(t)dt$). Determine the mean $\overline{z} \triangleq E[z]$ and variance $\sigma_z^2 \triangleq E[z^2]$ of z.
4. Consider the special situation that $n(t)$ is uncorrelated and has autocorrelation function $R_n(\tau) = N_0\delta(t)$. Show that $\sigma_z^2 = N_0 E_s$ where $E_s \triangleq \int_{-\infty}^{\infty} s^2(t)dt$ denotes the energy of $s(t)$.

1.19 Let g_k be a causal sequence (i.e. $g_k = 0$ for $k < 0$) with $g_0 \neq 0$. Assume that g_k is convolved with a sequence h_k and that the resulting sequence f_k is also causal. Prove that h_k must also be causal. Hint: assume that h_k is not causal and show that this will cause f_k to be nonzero for at least one instant $k < 0$.

1.20 In this problem we show that the phase $\phi(e^{j\omega T})$ of any causal and stable phase shifter is non-positive for all ω. Since any phase shifter can be decomposed as a cascade of first-order and second-order phase shifters, it suffices to consider these two building blocks.
First-order phase shifter: Consider the transfer function $H(z) = (z^{-1} - a)/(1 - az^{-1})$ where a is real-valued and $|a| < 1$. Show that $H(z)$ characterizes a stable and causal all-pass filter. Determine the phase characteristic $\phi_h(e^{j\omega T}) \triangleq \text{Arg}[H(e^{j\omega T})]$ and show that ϕ_h is non-positive for all ω.
Second-order phase shifter: Consider the transfer function

$$H(z) = \frac{z^{-1} - a^*}{1 - az^{-1}} \frac{z^{-1} - a}{1 - a^* z^{-1}} \tag{1P.4}$$

where a is complex-valued with $|a| < 1$. Show that $H(z)$ characterizes a stable and causal all-pass filter. Determine the phase characteristic $\phi_h(e^{j\omega T}) \triangleq \text{Arg}[H(e^{j\omega T})]$ and show that ϕ_h is non-positive for all ω.

REFERENCES

[1] W.B. Davenport and W.L. Root, *Random Signals and Noise*. New York: McGraw-Hill, 1958.

[2] J.L. Doob, *Stochastic Processes*. New York: Wiley, 1967.

[3] A.W.M. van den Enden and N.A.M. Verhoeckx, *Discrete-Time Signal Processing - An Introduction*. Prentice-Hall International, U.K.: 1989.

[4] A.J. Jerri, 'The Shannon Sampling Theorem - Its Various Extensions and Applications: A Tutorial Review', *Proc IEEE*, Vol. 65, pp. 1565-1596, 1977.

[5] H. Nyquist, 'Certain Topics in Telegraph Transmission Theory', *AIEE Trans.*, Vol. 47, pp. 617-644, Apr. 1928.

[6] A.V. Oppenheim and R.W. Schafer, *Digital Signal Processing*. Englewood Cliffs, N.J.: Prentice-Hall, 1975.

[7] A.V. Oppenheim, A.S. Willsky and I.T. Young, *Signals and Systems*. Englewood Cliffs, N.J.: Prentice-Hall, 1983.

[8] J.G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1983 (2nd ed. 1989).

[9] A. Papoulis, *Probability, Random Variables and Stochastic Processes*. New York: McGraw-Hill, 1965.

[10] A. Papoulis, *Signal Analysis*. New York: McGraw-Hill, 1984.

[11] L. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Englewood Cliffs, N.J.: Prentice-Hall, 1975.

[12] I.S. Sokolnikoff and R.M. Redheffer, *Mathematics of Physics and Modern Engineering*. New York: McGraw-Hill, 1958.

[13] Y.P. Tsividis and J.O. Voorman (Eds.), *Integrated Continuous-Time Filters*. New York: IEEE Press, 1993.

[14] P. van der Wurf, 'Statistical Analysis of Synchronous Random Pulse Trains by Means of Hybrid Correlation Functions', in *Proc.* 11th *Symposium on Information Theory in the Benelux*, Noordwijkerhout, The Netherlands, Oct. 25/26, 1990, pp. 148-154.

[15] J.M. Wozencraft and I.M. Jacobs, *Principles of Communication Engineering*. New York: Wiley, 1965.

[16] R.E. Ziemer, W.H. Tranter and D.R. Fannin, *Signals and Systems: Continuous and Discrete*. New York: MacMillan Publishing Co., 1983.

[17] A.I. Zverev, *Handbook of Filter Synthesis*. New York: Wiley, 1967.

2

EXAMPLES OF DIGITAL BASEBAND TRANSMISSION SYSTEMS

2.1 INTRODUCTION

Digital transmission and recording systems invariably transport information across a band of restricted width. The center frequency of this band may be high with respect to the band width, as it is in radio transmission. It may also be of the same order of magnitude or even zero, as in cable transmission and digital recording. In the first case the information must be modulated on to a carrier in order to convert it into the passband. In the second case this step is, in principle, not needed. The adjective 'baseband' serves to indicate that no carrier modulation is used. Instead, baseband transmission systems typically use a modulation code to match the characteristics of the data to those of the channel. At the receiving end of the system, equalization, detection, timing recovery and adaptation techniques are combined to recover the data. In the forthcoming chapters we discuss all these subjects. By way of introduction and motivation, the present chapter gives a survey of three important application areas of digital baseband transmission, namely digital subscriber lines, digital magnetic recording and digital optical recording. The aim is to convey a flavor of the origin, nature and importance of the various transmission impairments that arise, to exemplify some of the techniques that exist to prevent and/or counteract them, and to show that the principal types of impairment are common to all systems. For future reference we also model some of the impairments analytically. We should stress that the present survey is only meant as an introduction to the various application areas. As such it is restricted both in scope and in depth. Section 2.5 lists several references that provide much more complete treatments.

2.2 DIGITAL SUBSCRIBER LINES

In the course of the twentieth century, analog telephony has become so widespread that almost every home in the Western world is now connected to the *local copper plant*, i.e. the maze of copper wires that link each telephone set to the nearest telephone exchange.

Signal cables in the copper plant were originally designed to handle analog voice signals with a bandwidth of about 4 kHz. This restricted bandwidth is ill matched to the needs of the information age, in which various types of digital information (computer data, digitized voice, audio and video) vie for customer access. To be able to transport all of this information, the copper plant will need to be replaced by a high-bandwidth infrastructure made up of such media as optical fiber and coax cable. The gigantic scale of the copper plant will render this transition both costly and slow. In the mean time, various electronic techniques are being deployed in order to exploit the capacity of the existing copper plant as well as possible. The basic approach is to terminate each local wire at both ends with a *transmission unit* that 'pumps' digital information across the wire (Fig. 2.1). These units together convert the analog wire into a Digital Subscriber Line (DSL). DSLs are at the heart of the emerging Integrated Services Digital Network (ISDN), which could be viewed as the digital counterpart of the Plain Old Telephony Service (POTS), with various digital services added.[1]

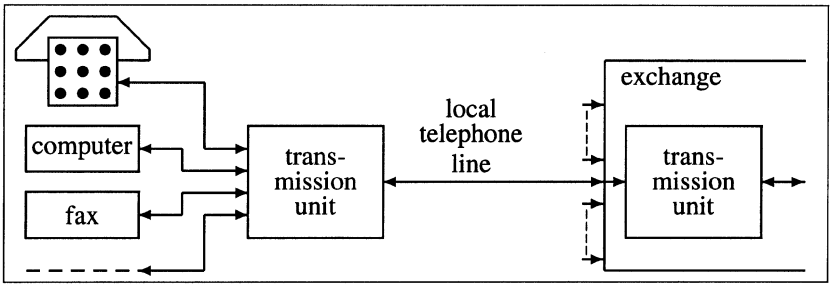


Figure 2.1: Bidirectional two-wire transmission in the local telephone network.

Data rates for 'basic ISDN access' are in the order of 160 kb/s. Transmission technology for DSLs at these data rates is now mature [6], [10]. The recent past has witnessed a rapid evolution towards higher data rates. Major approaches are designated high-bit-rate digital subscriber line (HDSL) and asymmetric digital subscriber line (ADSL) [13], [8], [7], [9]. HDSL uses two wire-pairs to transport 1.544 Mb/s in both directions. ADSL uses one pair to transport 6.144 Mb/s to the subscriber on top of a 640 kb/s bi-directional channel and a bi-directional POTS channel [9]. In both cases, transmission distances are restricted to a so-called *carrier serving area* (CSA) of about 4000 metres. Substantially higher bit rates than those in HDSL and ADSL can be delivered by a 'fiber-to-the-curb' network in which a single optical fiber serves multiple homes, and in which only the last few hundred meters of each connection consist of copper wire. A *very high bit-rate digital subscriber line* (VDSL) is being defined to bridge this final drop. Downstream data rates in VDSL are projected to reach approximately 60 Mb/s. Finally, data rates upwards of 100 Mb/s across unshielded twisted-pair cables

[1] ISDN has become a buzzword that means different things to different people, and we will not attempt to give a precise definition here.

are pursued in various local networking applications, including ATM, FDDI, and fast Ethernet [3], [5], [4].

2.2.1 Narrowband ISDN

Transmission units for narrowband ISDN transport some 160 kb/s worth of information in both directions simultaneously. They must provide a highly reliable digital connection across any wire pair with no need for manual adjustments. To this end, various transmission techniques are combined so as to overcome the principal impediments to reliable transmission [6], [15], [10]. We now make a brief guided tour.

Echos To enable simultaneous bidirectional transmission across the single wire pair available to each subscriber, hybrid coupling circuits at each end of the connection separate incoming and outgoing data signals. Fig. 2.2 pertains to a single end of the connection. The other end is, in principle, identical. Because of component tolerances, perfect separation of incoming and outgoing signals is, in general, not feasible. Furthermore, reflections on the line cause an additional portion of the transmitted signal to leak back into the local receiver. On long connections, these echos may be some 20 dB larger than the strongly attenuated desired signal component s. Echo cancellation techniques serve to overcome this problem.

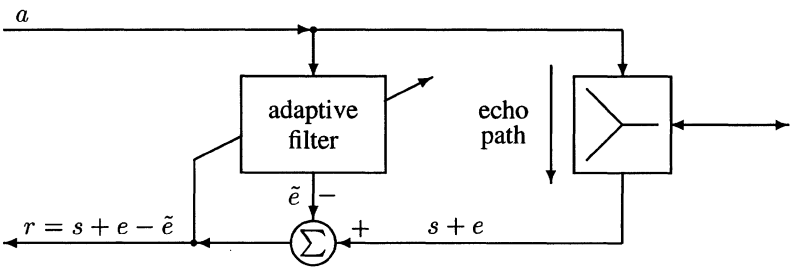


Figure 2.2: Echo cancellation.

An adaptive filter is used to form a replica $\tilde{e}$ of the echo e, which is subtracted from the received signal $s + e$. Ideally, only the desired signal component s remains after cancellation. The filter may be controlled on the basis of the residual signal $r = s + e - \tilde{e}$ by means of the LMS adaptation algorithm or derivatives thereof (see Chapter 8). Adaptivity is required because the echo depends, among other factors, on the length, gauge and temperature of the wire pair and on component values in the hybrid circuit, none of which are precisely known beforehand. While the bulk of the echo depends in a linear fashion on the transmitted data a, there may also be nonlinear components as a result, for example, of signal unbalance of the transmitter or saturation in the signal transformer that forms part of the hybrid coupling circuit. Because of the high degree of echo suppression that is required, it is generally unacceptable to leave nonlinear echo

components uncompensated. For this reason the adaptive filter in Fig. 2.2 may need to be able to generate the principal nonlinear echo components on top of the linear ones.

Intersymbol interference In baseband data transmission, information is usually conveyed via the amplitude of transmitted pulses. The simplest possible example is the non-return-to-zero (NRZ) modulation code. Here binary information of data rate $1/T_s$ is transmitted in the form of rectangular pulses of width $T = T_s$ and amplitude ± 1 (Fig. 2.3). Spectrally, most of the power of NRZ is concentrated in the band $[-1/(2T_s), 1/(2T_s)]$. Wire pairs in the local copper plant have a useful bandwidth that is inversely proportional to their length and that can be as small as tens of kiloHertzes. At data rates $1/T_s$ of around 160 kb/s, the power spectrum of NRZ clearly does not 'fit' in this bandwidth. To mitigate this mismatch, transmission units for narrowband ISDN use a modulation code called 2B1Q. This code maps pairs of bits (2B) into single quaternary symbols (1Q). The duration T of these symbols is twice as large as for NRZ ($T = 2T_s$), i.e. the bandwidth of the transmitted signal is twice as small. A disadvantage of 2B1Q is that the spacing between data levels is much smaller than for NRZ. This reduced spacing stems from legal and physical restrictions, which pose limits to the power and/or amplitude of the transmitted signal. In Fig. 2.3, the 2B1Q signal is taken to have the same amplitude as the NRZ signal. As a result, the spacing between adjacent data levels is 3 times smaller than for NRZ, and the margin against various disturbances (such as crosstalk and noise) has decreased accordingly.

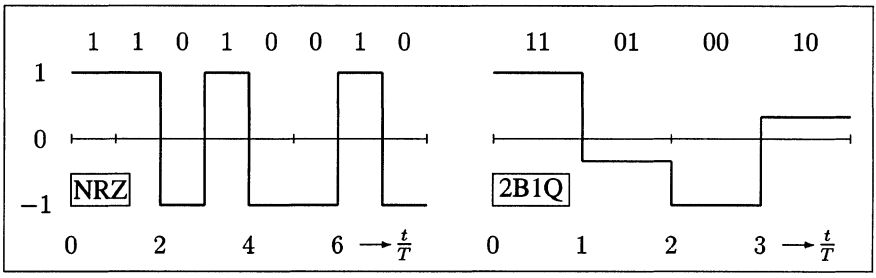


Figure 2.3: Comparison of NRZ and 2B1Q.

Even with 2B1Q, the bandwidth of the transmitted data signal still exceeds that of the wire pairs, especially on long connections. As a result, individual transmitted pulses become dispersed. This is illustrated in Fig. 2.4 for a typical signal cable (diameter 0.4 mm, capacitivity 46 nF/km).

As the length of the cable increases, pulses are increasingly attenuated and become ever longer. The increased pulse duration causes neighboring pulses to overlap and to interfere with each other. This effect is called intersymbol interference (ISI). The severeness of ISI is often gauged via an *eye pattern* or *eye diagram*. Such a diagram appears when a data signal is displayed on an oscilloscope that is synchronized to the data clock. Because of the synchronization, the traces of the signal are overlaid on the screen

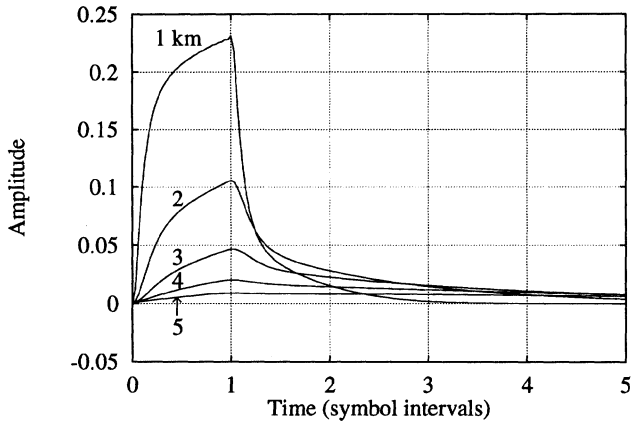


Figure 2.4: Response $h(t)$ of typical signal cable to a single rectangular pulse of width $T = 2T_s$ (with $1/T_s = 160$ kbit/s).

of the oscilloscope in a phase-aligned manner. The shape and size of the resulting 'eye' provides an indication of the margin of the system against various disturbances. By way of illustration Fig. 2.5 depicts eye patterns at the output of some of the cable segments of Fig. 2.4. Only for the shortest cable (1 km) are the 4 transmitted data levels clearly discernible. Beyond that length, the eye patterns silt up completely as a result of ISI.

The ISI of Fig. 2.5 has a linear nature because it emerges as a result of linear filtering of a 2B1Q signal. ISI can, however, also have significant nonlinear components. This is so, for example, in digital magnetic and optical recording (see Sections 2.3.4 and 2.4.2).

To deal with ISI, ISDN transmission systems make use of a so-called decision-feedback equalizer (Fig. 2.6). Basic to the DFE is the observation that the pulses $h(t)$ of Fig. 2.4 all have a minimum-phase-like nature, i.e. that they are causal and have their energy concentrated near the time origin. Thus the amplitude and polarity of an incoming pulse can be reliably detected from its initial part. On the basis of this symbol decision, a copy of the tail of the pulse can then be synthesized by means of a feedback filter and subtracted from the incoming signal. In effect, this procedure truncates $h(t)$ to a length of only one symbol interval, thereby eliminating ISI altogether. The procedure only works, of course, when the impulse response of the FBF matches the tail of $h(t)$. This is accomplished in practice by making the filter adaptive (see Chapter 8). A potential problem of the DFE is that erroneous decisions may propagate via the FBF to cause subsequent decision errors. Fortunately, this problem is typically only minor. In Chapter 6 we discuss decision feedback equalization to greater detail.

It is worth mentioning that pulses $h(t)$ do not always have the desirable minimum-phase-like shape of Fig. 2.4. Deviations of this shape are mainly due to *bridged taps*. These are open-circuit wire pairs, which are connected in shunt with working wire pairs.

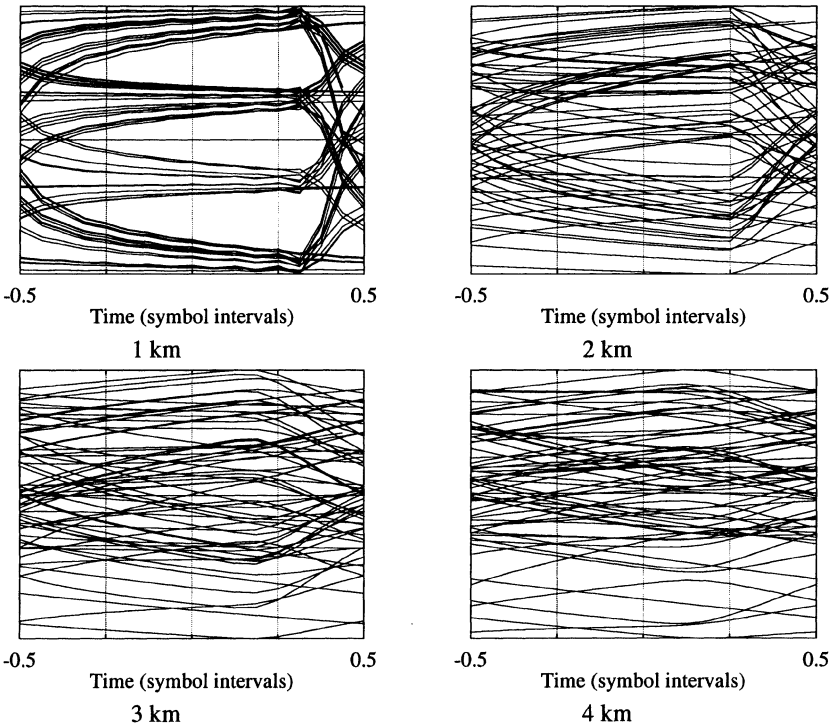


Figure 2.5: Eye patterns for systems with 2B1Q modulation code operating over some of the signal cables of Fig. 2.4. No other disturbances besides ISI are present.

They are intended to provide flexibility for future additions and changes in service demands. Bridged taps induce signal reflections that may cause 'bumps' and other irregularities in the waveforms of Fig. 2.4. As a result, DFE performance may degrade unless a prefilter is added that adaptively corrects for these irregularities.

Crosstalk Signal cables in the local telephone network often comprise many wire pairs, each for a different subscriber. Crosstalk is the interference that is induced via some coupling path by signals in one or more neighbouring wire pairs. In Fig. 2.7, a signal x is generated at the input of one of the wire pairs. As the signal propagates through the cable it generates two types of crosstalk in other wire pairs. The crosstalk y_n, which appears on the left, is called near-end crosstalk (NEXT). The crosstalk y_f, which appears on the right, is called far-end crosstalk (FEXT). NEXT is generally more damaging than FEXT because NEXT, unlike FEXT, does not necessarily propagate through a long loop and may thus experience a much smaller propagation loss than the desired signal component.

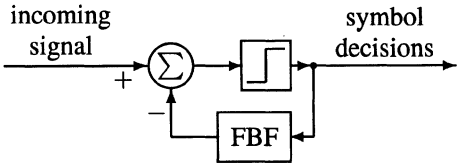


Figure 2.6: The decision-feedback equalizer contains a feedback filter (FBF) that is excited by past decisions. The FBF output signal compensates the tails of incoming pulses so as to eliminate ISI.

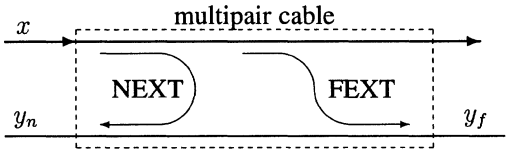


Figure 2.7: NEXT and FEXT generation in a multipair cable.

Coupling between wire pairs is predominantly a capacitive phenomenon and hence increases with frequency. For this reason it is desirable that the signals propagating through a multipair cable have a restricted bandwidth. This favors the use of a multilevel modulation code such as 2B1Q over a binary code such as NRZ. At the receiving end of the system, a DFE prefilter can be used to suppress the out-of-band crosstalk components.

Individual crosstalk components are all cyclostationary, i.e. their amplitude (or power) varies periodically with period T. The compound crosstalk disturbance may or may not be cyclostationary, depending on correlation between the individual components and on their mutual phase-alignment.

Impulse noise At irregular instants in time, pulse-like disturbances are injected into the system by a variety of man-made and environmental sources (electrical machinery, dialing pulses in analog telephony, electrostatic discharges, lightning, etc.). Together referred to as impulse noise, these disturbances cause random errors that can in principle be handled with the help of error protection techniques [11]. Unfortunately, this approach requires introduction of redundancy which increases the bandwidth of the transmitted signal and thereby increases signal attenuation, ISI and crosstalk. This is one reason why error protection is not used in narrowband ISDN. An alternative approach involves the use of smearing filters ([1], Fig. 2.8).

At the receiving end, a *desmearing filter* smears incoming pulses out in time so as to reduce their amplitude. This process inevitably distorts the data component of the received signal. To undo this effect, the data is predistorted at the transmitting end with the help of a complementary *smearing filter*. Ideally, the cascade of both filters represents a mere delay. Although they are a quite practical solution, smearing filters have not found widespread application in narrowband ISDN. Instead, one typically uses repeat-

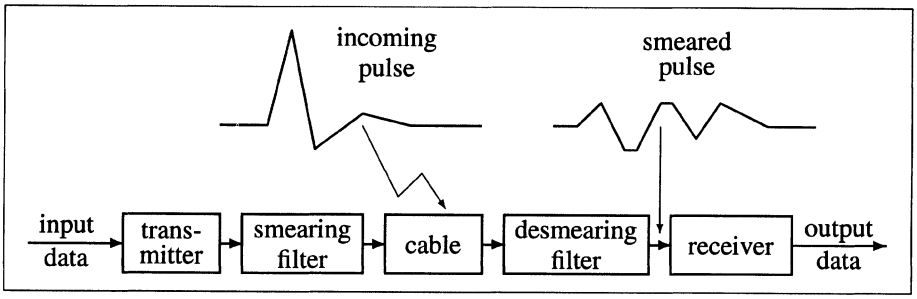


Figure 2.8: Suppression of impulse noise by means of smearing filters.

ers on lines that are hampered by impulse noise, thus ensuring that data levels remain well above the noise floor.

Additional transmission impairments Impairments not mentioned so far include thermal and other noise as well as signal reflections due to gauge changes and impedance mismatches. These impairments tend to be comparatively insignificant, and they will not be discussed here in further detail.

2.2.2 HDSL, ADSL and VDSL

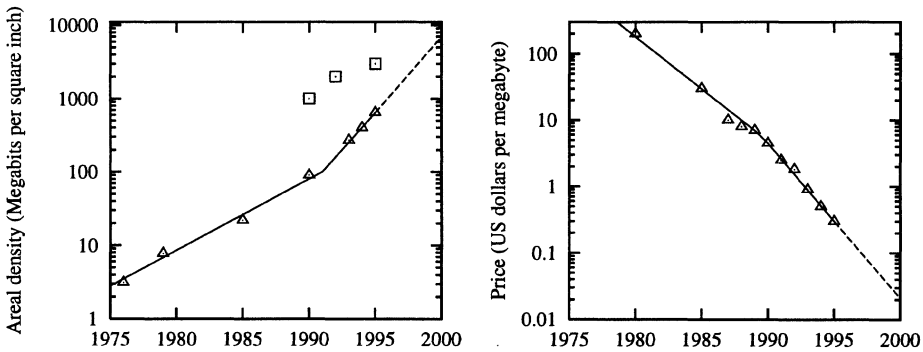
Data rates in HDSL, ADSL and VDSL are much higher than those in narrowband ISDN. This compounds all of the above transmission problems severely. The single most important contribution towards overcoming these problems is a reduction of the range of the connections to a CSA of about 4000 metres for HDSL and ADSL, and several hundreds of meters for VDSL. This reduction restricts the attenuation of the data signal, thereby improving worst-case signal-to-disturbance ratios. The use of a CSA implies that some of the remotest subscribers are beyond reach. For these subscribers, added measures must be taken, such as wire-pair selection or installation of repeaters.

A second contribution towards mitigating transmission problems involves the use of more sophisticated transmission techniques than for narrowband ISDN. This possibility is used in ADSL. Here an approach based on *discrete multicarrier modulation* has been standardized. Multicarrier modulation is basically a form of frequency-division multiplexing [2]. The available signal bandwidth is subdivided into many narrow frequency bands, and each of these is used to transmit a fraction of the user data. Modulation and demodulation are conveniently based on the fast Fourier transform, and optimization of the allocation of bits and transmit power across the available bands, depending on the signal-to-disturbance ratio in each band, permits close-to-optimum data throughputs.

2.3 DIGITAL MAGNETIC RECORDING

Magnetic recording was first described in 1888 [74], and first demonstrated in 1898 by the Danish engineer Vandemar Poulsen in the form of his 'Telegraphone' [68], for which he received the Grand Prix at the 1900 Paris World Exhibition. In Poulsen's machine, analog voice signals were stored on a steel piano wire that was wound spirally around a rotating drum. It took several decades and such important inventions as electron tubes, AC-biasing [27] and particulate media [65] before magnetic recording systems became practical enough for large-scale use.

Digital recording seems to originate after the second world war, linked to the advent of digital computers [79], [24], [66]. Early tape storage systems such as UNIVAC's UNISERVO I (1950) were followed in 1956 by the first disk drive, IBM's 350 [36], [35]. Since then, storage densities, capacities and prices per bit have improved rapidly and continuously (Fig. 2.9). To this date, the frantic paces of Fig. 2.9 continue unabated, owing in part to the massive momentum that springs from the sheer volume of the hard disk market. In recent years, digital storage on tape has branched out towards such applications as digital audio recording (DAT [60], DCC [39]) and digital recording of both standard-resolution and high-resolution video signals [32], [37]. Areal recording densities are projected to reach 10 Gbits per square inch in the early part of the 21st century, within one order of magnitude of what is currently thought to be the fundamental limit.



Areal density trends. Triangles: IBM products; squares: experiments.

Cost trends for average-capacity hard disk drives.

Figure 2.9: Evolution of areal densities and price per bit in hard disk drives.

Continuous improvements in magnetic media and recording/reproduce heads have been key factors in the above evolution. Signal processing techniques have played a much more modest and supporting role, although their level of sophistication has also increased steadily. We will sketch some of these techniques when discussing the principal impairments that arise during recording and playback (Sections 2.3.4 and 2.3.5). An in-depth treatment may be found in later chapters.

2.3.1 Magnetic recording technology

Fig. 2.10 depicts some of the basic elements of a magnetic tape recorder. The tape consists of a nonmagnetic substrate on which a layer of magnetic particles is deposited. It is wound on a reel and is transported past a magnetic head. The head is a gapped toroid with windings around a core portion away from the gap. The relative motion of tape past the head permits writing or reading, and causes a translation between the temporal domain of write and replay signals and the spatial domain of magnetization patterns on tape. For ease of handling the tape reels are often enclosed in a cassette. A capstan is used to accurately control the tape-to-head velocity.

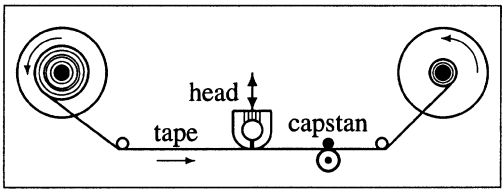


Figure 2.10: Magnetic tape recorder.

Bit rates in Fig. 2.10 depend on the physical length of bits on tape and on the tape-to-head velocity. Practical bit lengths are roughly in the order of 0.1 to 1 μm, while simple tape handling mechanisms can achieve tape speeds in the order of 0.1 to 1 m/s. Bit rates beyond 0.1 to 10 Mb/s are, therefore, difficult to achieve with the configuration of Fig. 2.10. There are basically two ways to break this bottleneck.

1. Several tracks can be written and read in parallel. This requires multi-track heads as applied, for example, in computer tape drives [36] and in S-DAT and DCC digital audio recorders [60], [39].

2. A rotating head can be used. The prevailing approach is called *helical scanning* (Fig. 2.11). Here the tape is wrapped helically around a relatively large drum. The tape runs slowly and the drum rotates rapidly. The tracks written by the rotating heads in the drum lie at a shallow angle to the tape. This system combines a low tape consumption with a large tape-to-head velocity. Examples include R-DAT [78], the 8-mm video system with digital sound, and professional digital recording systems for both standard-definition and high-definition video [60, 54, 25, 37].

An illustrative comparison of these two approaches for digital audio applications may be found in [60]. A principal disadvantage of all tape-based systems is their inability to provide random access. Disk-based systems provide relief in this respect. Of greatest practical importance are hard disk drives of the Winchester type. Central to Winchester technology are a closed head-disk assembly (HDA) and the fact that the head is kept flying above the disk on a thin film of air. The HDA includes one or more disks mounted on a common rotating spindle (Fig. 2.12).

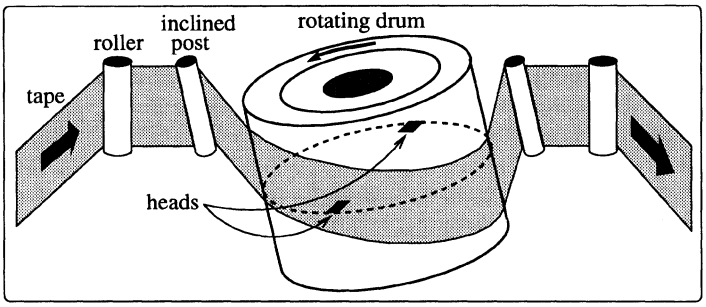


Figure 2.11: Helical scanning mechanism.

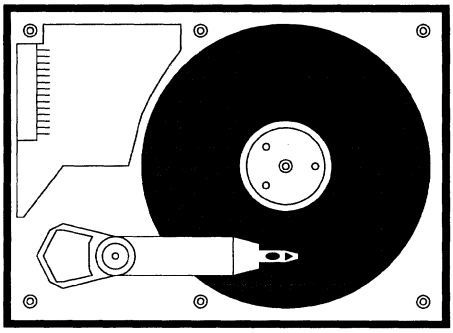


Figure 2.12: Schematic view of the interior of a hard disk drive.

An assembly of record/playback heads is attached to a common actuator which provides access to the recorded tracks of information. Heads are separated from the disk by an extremely thin film of air which is generated by the motion of the disk under the bearing surface of the head. The very close proximity of the head to the disk is a prerequisite for high information densities, and has been compared to a Boeing 747 skimming only meters above the earth. Life time of the heads is restricted by the wear that occurs during start and stop operations, when the head takes off from the disk or lands on it. Recent years have witnessed a transition from inductive read/write heads to heads with an inductive write element and a magneto-resistive (MR) read element. The excellent and ever-improving sensitivity of MR heads has been a key factor in the accelerated density improvements after 1991 in Fig. 2.9. While MR heads behave similarly to inductive read heads in several respects, there are also important differences (see Section 2.3.5).

Capacity improvements in both tape- and disk-based systems tend to occur along three axes simultaneously.

1. The physical length of bits on tape or disk is reduced. This requires better media and heads, and -to a lesser extent- more powerful signal processing techniques.

Hard disk drives nowadays use mostly *zone-density recording*. Here the disk is divided into radial recording zones of constant bit rate so as to keep recording density nearly fixed across the disk. This scheme stores about two times more bits per platter than *constant bit-rate* recording, where bit rates are constant across the disk and densities decrease towards the outer radius.

2. The width of tracks of bits is reduced. This also requires better media and heads. Furthermore, track-following mechanisms must become more accurate. For example, advanced video tape recorders use pilot tones that are embedded in adjacent tracks to permit accurate track following [44]. An alternative approach is *sectorized servo tracking* or *embedded servo*, where bursts of servo information are interspersed between data sectors in a time-multiplex fashion. This technique is used in hard disk drives to sustain track widths of only a few micrometers [33].

   It is worth noting that track widths in magnetic recording tend to be significantly larger than those in optical recording, even though recording densities are more or less at par. The reason is that 'optical bits' are more or less square, while 'magnetic bits' have an aspect ratio in the order of 10. This large ratio eases tracking requirements to some extent.

3. Tape thickness is reduced so that longer tapes fit inside the same cassette. In similar vein, more disk platters are added in a given enclosure at the cost of added hardware and a decreased spacing between platters.

2.3.2 Magnetic recording

Fig. 2.13 illustrates the translation of temporal signals to a recorded pattern and vice versa. A signal current is applied to the windings of the recording head. This current magnetizes the head and causes a flux pattern that follows the head path and fringes from the head core due to the presence of the air gap. The gap is adjacent to the medium and the fringing head flux magnetizes the medium longitudinally, i.e. in the direction of tape motion. Via a different head construction it is, in principle, also possible to magnetize the medium vertically, i.e. perpendicular to the direction of tape motion. Perpendicular recording has attracted much scientific attention [40], [41], [58], [42] but has not yet found widespread application.

The recorded magnetization pattern will, in general, not precisely resemble the input signal waveform since magnetic recording is fundamentally a nonlinear process. However, if the input is a binary signal of sufficient amplitude, then the tape will be magnetized to saturation in one of two opposite directions, and to a first-order approximation the magnetization pattern will be a spatial copy of the incoming signal. By comparison, if more than two data levels were recorded then nonlinearities would be much more problematic and signal-to-disturbance ratios would decrease significantly. These two reasons explain in part why practical digital recording systems invariably use binary saturation recording.

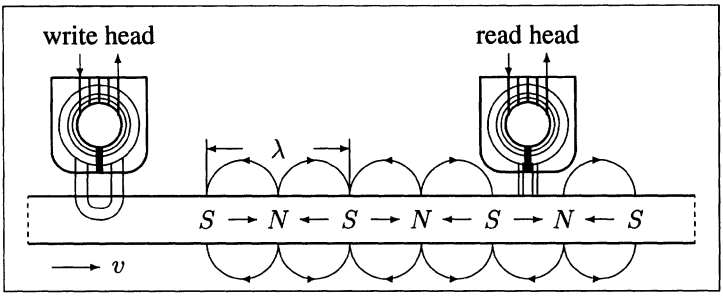


Figure 2.13: Illustration of the recording and replay process. Write and read heads are both of the inductive type.

If the input signal is a square wave with fundamental frequency f and the head-to-medium velocity is denoted v, then the recorded magnetization pattern will have a fundamental wavelength $\lambda = v/f$. A single period of the square wave may be regarded as a succession of two NRZ bit cells of width $\lambda/2$ and opposite polarity. The frequency f may, therefore, be equated with a bit rate $1/T = 2f = 2v/\lambda$.

As indicated in Fig. 2.13, a spatially varying magnetization pattern will produce external fields. When the tape passes the read head, these fields provide a flux source that circulates though the head core. The flux Φ, which threads the N windings, induces a temporal output voltage $V = -Nd\Phi/dt = -Nvd\Phi/dx$. This voltage increases linearly with tape speed; the differentiation causes the head to reject DC. It should be emphasized that this behavior is characteristic of inductive read heads. Magneto-resistive heads sense flux directly and their output is, in principle, independent of v. In most other respects they behave quite similarly to inductive heads (see Section 2.3.5).

If the head is in full contact with the tape then the smallest details that can be resolved in the magnetization pattern are in the order of the gap width g. Resolution improves as the gap width decreases. In practice there is effectively always some distance d between the head and the tape. This distance causes the head to look sidewards in both directions, thereby limiting resolution. Furthermore, the strongest field lines, i.e. those closest to the tape, no longer reach the head and as a result the amplitude of the output signal will decrease. This heuristic argumentation suggests that recorded bits can only be recovered reliably if physical bit lengths are sufficiently large in relation to both d and g. We will later make these insights more precise.

2.3.3 Recording media

The classical medium for magnetic recording contains small needle-shaped magnetic particles dispersed in an organic binder. This composite layer is supported either by a flexible polymeric film or by a rigid disk. Ideally the particles are packed at high packing fractions, are well oriented and are uniformly dispersed. The attainment of

these goals is a demanding task. Variations in these factors can easily account for greater performance differences than the magnetic material used.

Key properties of magnetic media are their *coercivity*, *remanence*, and *energy product* (Table 2.1). Coercivity measures the field required to reverse the direction of magnetization of a recorded region in the medium. Higher-coercivity media are less likely to demagnetize, a critical design factor where tiny regions of alternating magnetization must be packed together. The remanence is the amount of residual magnetization left in the medium after the writing field has disappeared. The energy product is a measure of the magnetic energy that the medium contains per unit of volume, and is basically the product of coercivity, remanence, and the inductance constant $\mu_0 = 4\pi.10^{-7}$ H/m. A higher energy product means that magnetic particles can become smaller. Smaller particles give lower levels of media noise and permit fabrication of smoother media, which permit the head to be in closer proximity to the media.

media type	coercivity (kA/m)	remanence (kA/m)	energy product (kJ/m^3)
gamma ferric oxide	28	115	4
metal particle	120	200	30

Table 2.1: Typical values of coercivity, remanence and energy product for particulate media.

The lower limit on particle size is determined by the superparamagnetic limit, the point at which the individual particle's magnetic energy becomes comparable to random thermal energy that can spontaneously reverse its magnetization. For Cobalt-like materials it has been estimated that spherical particles with a diameter of 10 nm can still be sufficiently stable. Furthermore, in order to achieve sufficient SNR to permit reliable bit-detection it has been estimated that a single bit must be 'spread' across a plane containing some 100 particles or more. This reasoning suggests that areal recording densities can be as high as 100 bits per μm^2, i.e. around 2 orders of magnitude beyond the most advanced current-day products.

In hard disk drives, particulate media have gradually been replaced by media with continuous metal films, typically a Cobalt alloy. Magnetic tapes are still mostly particulate, but thin metal films do show up in Metal-Evaporated (ME) tape. Main advantage of thin-film media is that their packing fraction approaches 100 percents, i.e. 2 or 3 times that of particulate media. In theory, remanence and energy product improve by the same factor. In practice, demagnetization effects offset this improvement to some extent [87], but even so thin-film media tend to have excellent recording properties.

2.3.4 The write process

Fig. 2.14 attempts to convey the essence of the write process. The head field is in the opposite direction to the initial medium magnetization. In a circular region near the gap, the field exceeds the coercivity and the magnetization reverses. Beyond the trailing edge of this 'write bubble', the head field can no longer change the magnetization

pattern and the pattern solidifies. To a first-order approximation we may, therefore, regard this edge as the point where transitions are actually being written.

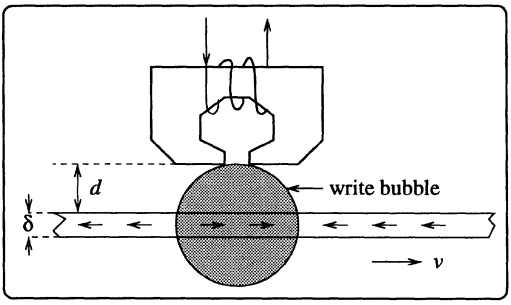


Figure 2.14: Illustration of the write process.

Ideally the recorded magnetization pattern is an exact spatial copy of the binary write waveform. In practice various anomalies cause a detraction from this ideal. We discuss some of the major imperfections as well as some partial remedies.

Bandwidth limitations of the write path The size of the write bubble is proportional to the write current and would be constant over time if write current were perfectly binary and head fields would react instantly to a change of the write current. In reality, however, write currents cannot have infinitely steep transitions (see Problem 2.1). Furthermore, eddy-current effects and other frequency-dependent head losses cause the head field to be a low-pass function of the write current [88], [23], [47], [48]. The net effect of these and other bandwidth limitations is that the size of the write bubble varies dynamically.

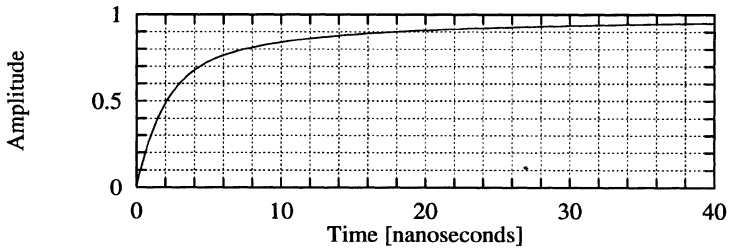
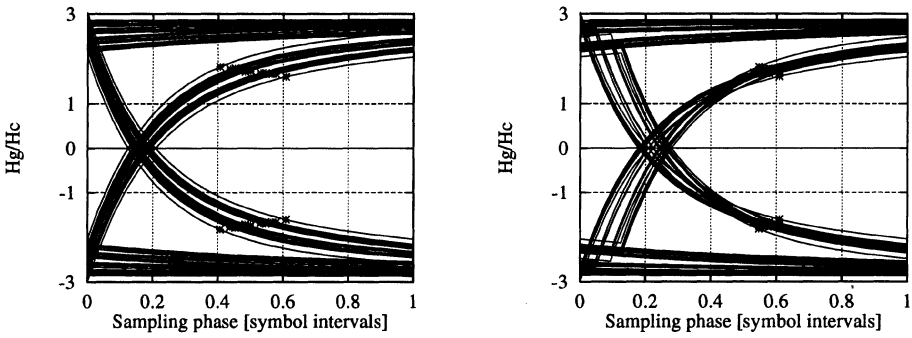


Figure 2.15: Illustrative response of the head field to a unit step in write current. After [48].

In response to a single data transition, the head field initially changes rapidly but creeps increasingly slowly towards its final value (Fig. 2.15). At high data rates the entire response spans many symbol intervals, and successive responses will overlap. The ensuing intersymbol interference is clearly visible in the eye pattern of the gap field H_g, which is illustrated in Fig. 2.16.a.

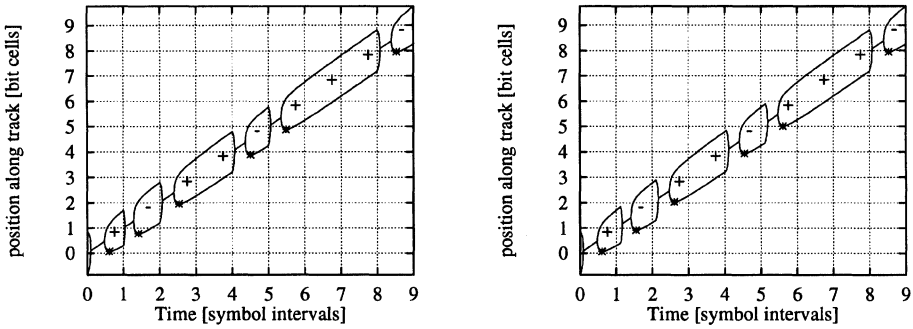


a. No write precompensation.

b. Second-order write precompensation ($\varepsilon_1 = 0.12, \varepsilon_2 = 0.04$).

Figure 2.16: Eye patterns of the gap-field H_g for a system with uncorrelated binary data a_k and a head response according to Fig. 2.15. The data rate $1/T$ amounts to 100 Mb/s, i.e. the symbol interval T lasts 10 nanoseconds (compare Fig. 2.15). The dashed lines indicate the coercivity levels $\pm H_c$. The asterisks indicate the moments at which the transitions are recorded on the medium.

The head moves relative to the medium and so does the center of the write bubble. In Fig. 2.17 it is assumed that the medium is infinitely thin, and that the head moves linearly upward in time. The upper and lower lines demarcate the leading and trailing edges of the write bubble, respectively. The bubble vanishes whenever both lines coincide. The medium is at some distance d from the gap and for a write bubble to emerge the field strength $|H_g|$ must exceed the coercivity H_c by some margin. Beyond this threshold the bubble grows rapidly. Successive bubbles have alternating polarity as indicated by the plus and minus signs, which also indicate the underlying data symbols a_k. Any given point of the medium is typically traversed by several write bubbles in succession, and the magnetization only solidifies once the trailing edge of the last bubble has passed by. Transitions are written whenever the trailing edge of a bubble reverses its motion under the medium. These points are marked by asterisks in Figs. 2.17 and Fig. 2.16. To a first-order approximation, transitions are recorded on the medium wherever $|H_g|$ exceeds a threshold that is somewhat larger than the coercivity H_c (compare Fig. 2.16.a). As a result of the bandwidth limitations, transitions are shifted relative to each other by up to 20 percents of a symbol interval T. These shifts are a nonlinear function of the preceding bits (see Appendix 2A for an analysis). As suggested by the eye pattern, the shifts can be reduced by deferring a data transition somewhat whenever a transition has occurred in one of the immediately preceding bit intervals. This approach is called *write precompensation* and is widely used in hard disk drives. By way of illustration Appendix 2B elaborates a second-order write-precompensation scheme. This scheme clearly decreases peak-to-peak bit-shifts (Fig. 2.16.b), and causes write bubbles to evolve more regularly (Fig. 2.17.b).



a. No write precompensation.

b. Second-order write precompensation ($\varepsilon_1 = 0.12, \varepsilon_2 = 0.04$).

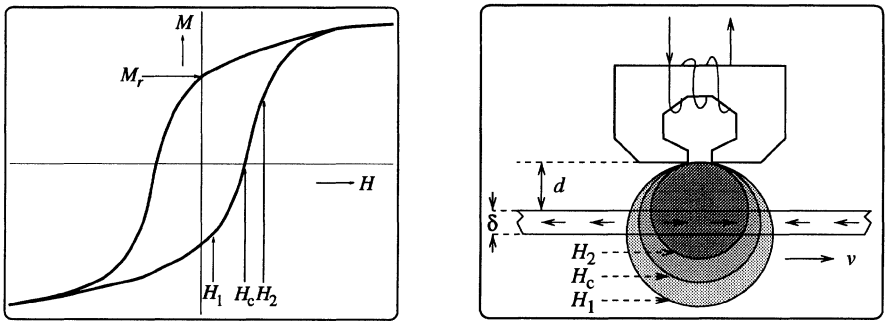
Figure 2.17: Temporal evolution of the write bubble for the system of Fig. 2.16 at a head-to-media velocity $v = 33.3$ m/s, gap width $g = 0.5$ λm and flying height $d = 100$ nm. Plus- and minus-signs indicate the direction of the magnetization. Asterisks indicate the points at which transitions are recorded on the medium.

Non-ideal remanence loop Unlike what we have suggested so far, the media magnetization M does not change abruptly in response to an applied field H. In reality this change is determined by a remanence loop as in Fig. 2.18.a. This hysteresis loop is not square so that the magnetization changes gradually for head fields in the vicinity of the coercivity. (The coercivity H_c is defined as the field for which M changes sign.) To a first-order approximation, transition widths are now determined by the fields H_1 and H_2 that are required to switch 25 and 75 percents of the magnetization, respectively (see Fig. 2.18.a). These fields each yield a distinct write bubble as shown in Fig. 2.18.b. The distance between the trailing edges of the inner and outer write bubbles defines the transition length.

For modeling purposes one often assumes that recorded transitions have an arctangent shape according to $M(x) \propto \arctan(x/a)$ where a is a measure of the transition length (Fig. 2.19). As suggested by Fig. 2.18.b, this length increases with the thickness δ of the medium and with the head-to-media spacing d. From Fig. 2.18.a, it decreases as the hysteresis loop becomes more 'square'. Subject to a number of simplifying assumptions (such as near-optimum write currents, infinitely small write-current rise times and very thin media) it has been estimated [53, Vol. I, Ch. 2] that

$$a \simeq \sqrt{\frac{M_r \delta d}{\pi H_c}}. \tag{2.1}$$

This crude approximation can be used to predict the maximum linear information density that can be achieved for a given head-media system.



a. $M - H$ remanence loop.

b. Write bubbles corresponding to fields H_1, H_c and H_2.

Figure 2.18: Typical $M - H$ remanence loop (a.) and the corresponding write bubbles (b.). Fields H_1 and H_2 are the 25- and 75-percent switching fields; field H_c is the remanent coercivity or 50-percent switching field.

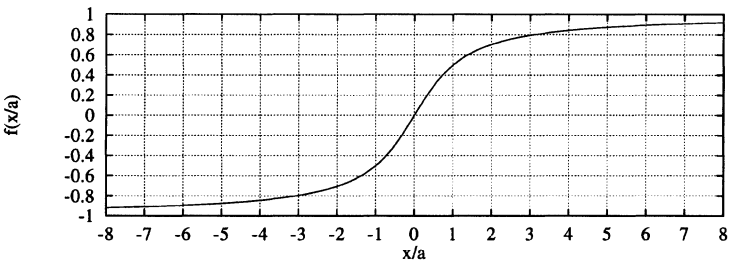


Figure 2.19: Arctangent transition $[f(\frac{x}{a}) = \frac{2}{\pi}\arctan\frac{x}{a}]$.

Example 2.1 For the metal-particle medium of Table 2.1 at a thickness $\delta = 50$ nm and a head-to-media spacing $d = 100$ nm we find that $a \simeq 53$ nm. If we require that a occupies at most 25 percents of a bit cell then bit lengths cannot be smaller than approximately 0.2 μm.

Reductions of the transition width can be obtained in various manners. For example, the magnetic medium can be modified to have a 'squarer' remanence loop. Also, the spatial distance between inner and outer write bubbles in Fig. 2.18.b can be reduced. This is possible by reducing the head-to-media spacing d. A reduction of the write current will make all write bubbles smaller and can help to make the transition sharper provided that the inner circle remains large enough to penetrate the entire medium. This argumentation suggests that there is an optimum to the write current. A rule of thumb is that the maximum write current should achieve a field strength $|H_g|$ in the gap of the head of about $2H_c$ to $3H_c$ [53, Vol. I, Ch. 2], [90].

Demagnetizing fields A large portion of the nonlinear distortion introduced in the recording process can be attributed to data-dependent shifts in the effective positions of

the recorded transitions. As explained above, bandwidth limitations of the write path can be a primary cause of such shifts. Another cause is formed by demagnetizing fields from preceding transitions. These internal fields oppose the direction of magnetization and make it, heuristically speaking, easier to write a new transition as the previous transition is closer by (see [22] for more details). To a first-order approximation, a written transition comes ε_1 symbol intervals earlier in the presence of an immediately preceding transition than for transitions that occur two or more symbol intervals ago (Fig. 2.20).

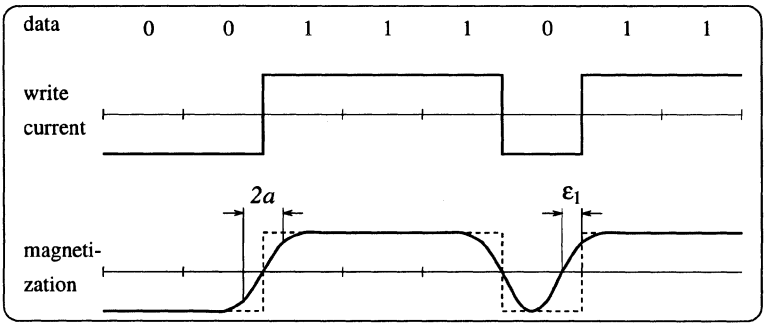


Figure 2.20: First-order nonlinear bit shift ε_1 in the presence of an immediately preceding transition.

The magnitude of ε_1 depends heavily on the precise recording conditions, and decreases rapidly as the minimum spacing between transitions increases. To achieve such an increment it is possible to use a modulation code that constrains the minimum run-length (see Section 4.8). Such a code reduces nonlinearities in exchange for a reduction of signal-to-noise ratio. An alternative technique for counteracting nonlinear bit-shifts involves the use of write precompensation. The idea is to compensate for the nonlinear shifts by writing the corresponding transitions somewhat later.

Exercise 2.1 Show that the presence or absence of a nonlinear bit shift in Fig. 2.20 is fully determined by the bit a_k that is being written in conjunction with the past two bits a_{k-1} and a_{k-2}.

In Fig. 2.20, the nonlinear bit-shift depends purely on the presence or absence of an immediately preceding transition. At high information densities, the impact of older transitions can also become significant [62]. This may necessitate higher-order forms of write precompensation. Appendix 2B is concerned with nonlinearity models and write precompensation for the situation that the shift is affected by the previous two transitions. Further studies on write nonlinearities and their characterization may be found in [62], [28], [69], [64], and [91]. Any non-compensated non-linear bit-shifts will induce nonlinear intersymbol interference in the replay signal. Such ISI can be handled, for example, by special forms of the decision-feedback equalizer and Viterbi detector (see Section 6.3 and Example 7.12).

Overwrite noise In most digital recording systems there is no specific erasure process, and new data are simply written over the old data. This technique generally fails to erase the old data completely. Especially the low-frequency components of the old data, which extend deeply into the medium, tend to be resilient [53, Vol. I, Ch. 5] and this is one reason why modulation codes for digital magnetic recording often have suppressed low-frequency content. It is also important to keep the media thin [34]. A basic scaling rule is that the medium thickness and the bit interval should remain in the same ratio; higher linear densities necessitate thinner coatings. Other important factors include the rise/fall times and bias current of the write preamplifier. A rule of thumb is that rise times should not exceed $0.4T$ where T is the bit spacing [90].

At the moment when a new transition is to be written, the 'past' portion of the medium (i.e. the portion that has left the write bubble) has just been overwritten and the 'future' portion (which has not yet reached the bubble) still contains the old data. We have argued above that the past portion may cause transition shifts depending on the presence or absence of immediately preceding transitions. The future portion has a similar effect. Again it will, crudely speaking, be relatively hard to write bits whose magnetization opposes the preexisting magnetization of the medium and written transitions will shift accordingly. In the particular event of a medium that is DC-erased in one direction (such as a virgin disk platter), transitions that are written in the opposite direction are shifted later by an amount ε_0 [62, 34].

Transition noise This phenomenon occurs mainly in thin-film media [20],[76], [72]. Principal cause is the microstructure of magnetic domains which causes the borderline between adjacent bit cells to have a zig-zag shape (Fig. 2.21). This shape is random and varies from one transition to the next. It defines an effective center and an effective width of the transition. A typical transition is centered properly and has an effective width $2a$. Excursions away from this average include random transition broadening [56] and random transition shifts [20], [72]. A simple additive model for these shifts is developed in Appendix 2C.

As the size of bit cells decreases, adjacent zig-zag domains eventually come into contact as in the right portion of Fig. 2.21. The net result is that written bit cells are partially erased [28]. Partial erasure can occur only for minimum-width bit cells, i.e. when both neighboring bit cells have opposite polarity. In this regard it is a nonlinear phenomenon. Interestingly, partial erasure can be counteracted to some degree by means of write precompensation, especially at data frequencies well below the Nyquist rate [64]. The basic trick is to move the two transitions that enclose a minimum-width bit-cell away from each other, both by the same amount. This is only done, of course, for transitions that border on a single minimum-width bit cell. Transitions that separate two such cells are not shifted.

2.3.5 The replay process

Replay heads come in two main types: inductive and magneto-resistive (Fig. 2.22). Inductive heads use a coil (not shown in Fig. 2.22) to sense the derivative of the flux that

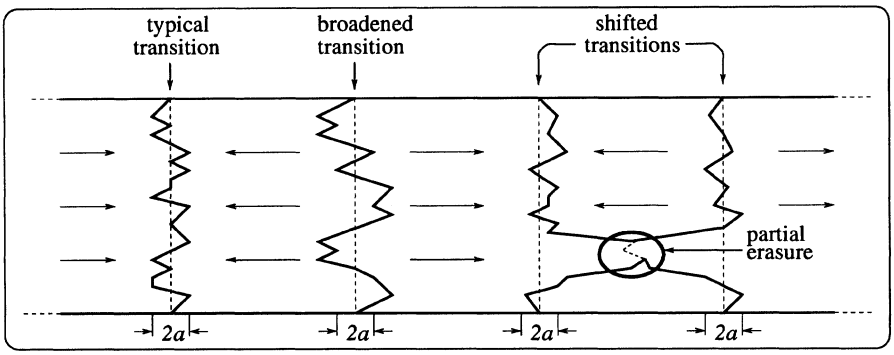


Figure 2.21: Illustration of transition noise and partial erasure.

circles the yoke. This flux enters the head via the gap. If the head is in full contact with the medium then the smallest magnetic details that can be resolved are in the order of the gap width g. Any distance d between head and media causes resolution to decrease.

MR heads use one or two sensors of magneto-resistive material that are typically placed between two shields. The shield-to-shield spacing b defines the spatial resolution of the head in a similar manner as the gap width g for inductive heads. The linear behavior of shielded MR heads, in fact, is generally quite similar to that of inductive heads.

Both inductive and MR heads come in many tastes (inductive: ferrite, metal-in-gap, thin-film, ...; MR: shielded/nonshielded, single-stripe/dual stripe, SAL/recessed sensor, ...). In our discussion of the replay process we shall focus on phenomena that are common to all. Differences between the various tastes are discussed, for example, in [53, Vol. I, Ch. 4] and (for MR heads) in [73].

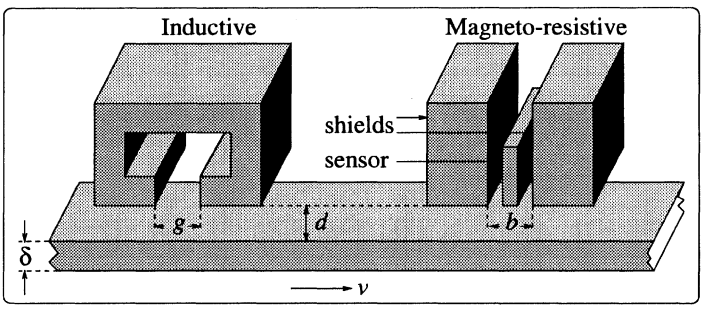


Figure 2.22: Schematic model of inductive and shielded magneto-resistive replay heads.

Linear intersymbol interference Unlike the recording process, the replay process is essentially linear, especially for inductive heads. Ideally the head would only be sensit-

ive to the magnetization of the medium in an infinitely narrow strip in front of the head. In practice, however, head resolution is imperfect and the head output also depends on 'past' and 'future' portions of the magnetization pattern. This dependence causes intersymbol interference (ISI). Typical waveforms are depicted in Fig. 2.23.

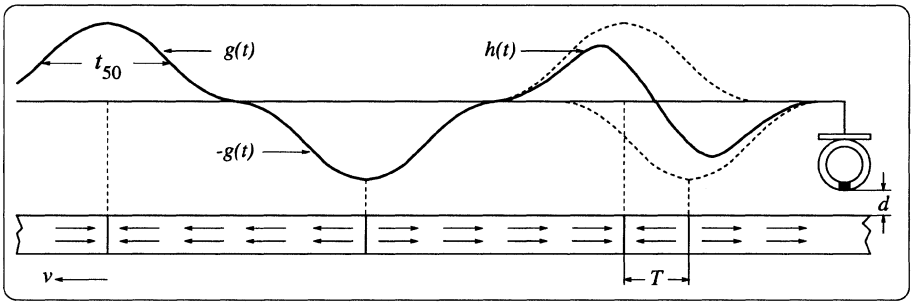


Figure 2.23: Illustration of typical replay waveforms in magnetic recording.

The head senses the derivative of the flux, i.e. it responds primarily to *transitions* of the magnetization pattern. For an isolated transition it produces a pulse $g(t)$ or its inverse $-g(t)$, depending on the direction of the transition. One usually refers to $g(t)$ as the *isolated transition response* or just *transition response*. Because of the symmetric construction of medium and head, $g(t)$ is symmetric in time, i.e. it has linear phase. The width at half amplitude defines the resolution of the recording process. It is denoted pw_{50} when measured spatially and t_{50} when measured temporally ($\mathrm{pw}_{50} = v.t_{50}$). Successive transition responses alternate in polarity and partly cancel each other when spaced closely together. The degree of cancellation is largest for transitions at the minimum spacing, i.e. a spacing of a single bit cell T. The response of the replay head to such a cell, i.e. to two transitions at minimum spacing, is designated *dibit response*, *bit response* or *symbol response* and is denoted $h(t)$. Clearly $h(t) = g(t) - g(t-T)$. This is a DC-free response whose high-frequency roll-off increases with t_{50}. The ratio $D \triangleq t_{50}/T$ is a normalized measure of the information density and defines how many bits are packed within the 'resolution unit' t_{50}. Practical values for D are in the order of 1 to 3. An increment of D causes a rapid decrement of the amplitude of the dibit response $h(t)$ and this, in turn, decreases signal-to-noise ratios and makes reliable bit-detection more difficult. An increment of D also increases the duration of $h(t)$ expressed in units T. As a result, any given bit will interfere with successively more preceding and subsequent bits, i.e. ISI will become more severe. These combined effects necessitate comparatively complicated reception techniques. A mainstream approach is called partial-response maximum-likelihood (PRML) reception (see Section 4.8.7). A simpler yet less powerful technique is discussed in Section 8.9. At low normalized information densities, on the other hand, signal-to-noise ratios tend to be comfortable and individual peaks of the transition responses remain clearly separate. The recorded data pattern as well as the clock can then be recovered in a simple manner by detecting these peaks (see Section 10.3.7 and in particular Fig. 10.21). Peak detection has long

been popular in conjunction with run-length-limited modulation codes, and continues to be applied in low-end hard disk drives.

In mathematical terms we may describe the data component $x(t)$ of the head output as

$$x(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT) \tag{2.2}$$

where a_k is the recorded bit sequence with symbols in the alphabet $\{-1, 1\}$. Upon realizing that $h(t) = g(t) - g(t - T)$ we may equivalently express $x(t)$ as

$$x(t) = \sum_{k=-\infty}^{\infty} b_k g(t - kT) \tag{2.3}$$

where $b_k \triangleq a_k - a_{k-1}$ the *transition sequence* associated to a_k. Clearly $b_k = 2$ when there is an 'upgoing' transition between the previous and current bit cell, $b_k = -2$ when there is a 'downgoing' transition, and $b_k = 0$ when there is no transition. System models according to (2.2) and (2.3) are depicted in Fig. 2.24.

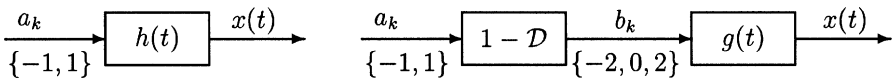


Figure 2.24: The data component $x(t)$ of the replay signal can equivalently be thought of as a filtered version of the data sequence a_k (left) or of the transition sequence $b_k = a_k - a_{k-1}$ (right; the operator $\mathcal{D}$ denotes a delay of one symbol interval).

By definition, $g(t)$ is the response to a step of unit magnitude. The -1-to-$+1$ transition $b_k = 2$ has magnitude 2 and by (2.3) yields a response $2g(t)$. Fig. 2.23 is somewhat deceptive in that the corresponding transition is labeled $g(t)$ for ease of reference. In similar vein, $h(t)$ is the response to a single bit cel of unit magnitude whereas the bit cel in Fig. 2.23 has magnitude 2 (and baseline -1). Thus the pulse that is labeled symbolically $h(t)$ in Fig. 2.23 is actually $2h(t)$.

The classical model for the transition response, due to O. Karlqvist [45], is the Lorentzian pulse

$$g(t) = \frac{T}{\pi t_{50}} \frac{1}{1 + (\frac{2t}{t_{50}})^2} = \frac{1}{\pi D} \frac{1}{1 + (\frac{2t}{DT})^2}. \tag{2.4}$$

This model accounts only for the transition width a (for arctangent transition shape) and the head-to-media spacing d, i.e. it assumes that the gap width g (or shield-to-shield spacing b) and the media thickness δ are negligibly small. Subject to these assumptions, $\mathrm{pw}_{50} = v.t_{50}$ is fully determined by a and d via the simple relation $\mathrm{pw}_{50} = 2(a + d)$ so that $t_{50} = 2(a + d)/v$. In frequency-domain notation we may write $G(\Omega) = (T/2)\exp\{-\pi D|\Omega|\}$ where $\Omega = fT$ is a normalized measure of frequency, equal to 1 for the symbol rate $1/T$. In the same notation, the Fourier transform of $h(t)$ is

$$H(\Omega) = 2jT\sin(\pi\Omega)G(\Omega) = j\sin(\pi\Omega)e^{-\pi D|\Omega|}. \tag{2.5}$$

These time and frequency responses are illustrated in Fig. 2.25 for various values of D. The energy of $h(t)$ decreases rapidly with D. Signal-to-noise ratios decrease accordingly (see Appendix 3A.1 for an analysis).

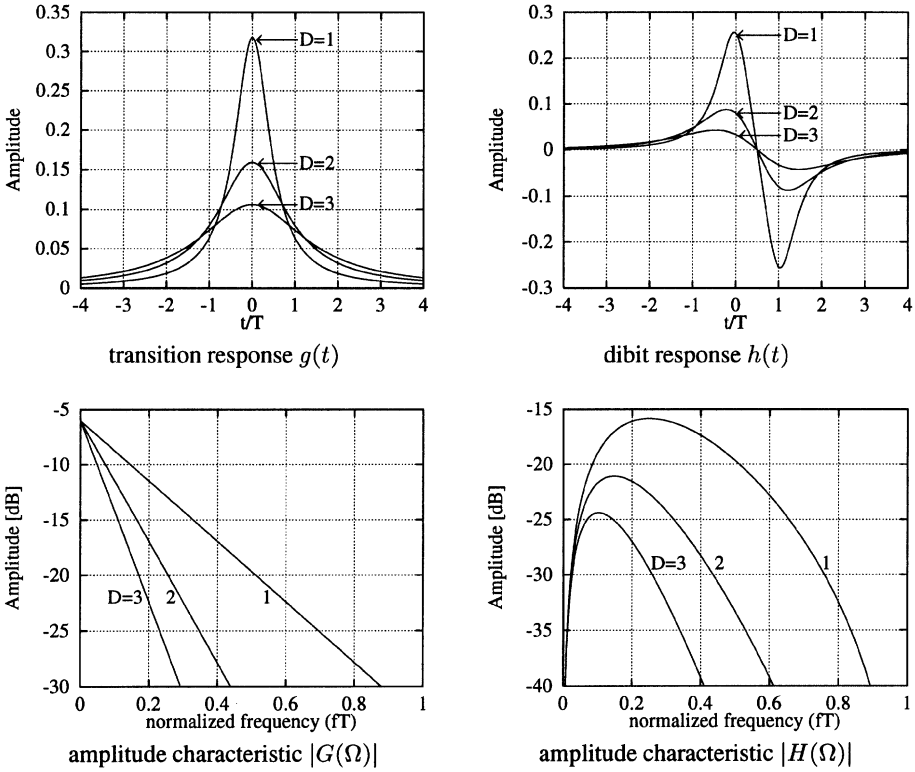


Figure 2.25: Transition response $g(t)$ and dibit response $h(t)$ of a Lorentzian channel for various values of the normalized information density $D = t_{50}/T$. Also depicted are the corresponding amplitude-frequency characteristics.

The amplitude characteristic $|G|$ of the transition response exhibits linear roll-off versus frequency when plotted in dB. Changes of D cause the line to rotate about its DC-value. Bandwidth is inversely proportional to D. In practice D may vary dynamically due, for example, to fluctuations of the head-to-media separation d. This will hardly affect $|G|$ in the vicinity of DC, but instabilities of $|G|$ will increase with frequency. These HF instabilities, which are equally manifest in the dibit response, may hamper reliable bit-detection unless adequate countermeasures are taken, such as a modulation code with suppressed HF content (see Chapter 4), reception techniques that avoid HF amplitude equalization (see Chapter 6), and adaptation techniques that help to track the fluctuations of D (see Chapter 8).

Non-negligible values of g, b and δ all yield additional high-frequency losses. For example, the medium thickness induces a *thickness loss* $[1 - \exp\{-2\pi\Lambda|\Omega|\}]/[2\pi\Lambda]$ that affects F and H multiplicatively [77]. This is a low-pass function whose bandwidth is determined by the parameter $\Lambda \triangleq \delta/(vT)$ which represents the media thickness normalized with respect to the bit length. For $\Lambda \ll 1$, the transition response remains approximately Lorentzian but with an effective density $D' = D + \Lambda$ (where $D = 2(a+d)/(vT)$) [21]. The value of t_{50} increases accordingly.

In similar vein, a nonzero gap size g induces a *gap loss* that may be approximated as $[\sin \pi G\Omega]/[\pi G\Omega]$ where $G \triangleq g/(vT)$ is the gap width normalized to the bit length [77]. This is a sinc function with zero crossings at $|\Omega| = 1/G, 2/G, \ldots$. A more accurate analysis reveals that these zeros are shifted slightly and the first zero is more typically in the vicinity of $0.88/G$ [53, Vol. I, Ch. 2]. Ideally $G \ll 1$. In that case the transition response remains approximately Lorentzian but with an effective density $D' \simeq \sqrt{(G/2)^2 + D^2}$ [53, Vol. I, Ch. 2].

The loss function that is associated with the shield-to-shield spacing b depends, among other factors, on whether the head has one or two stripes. For a dual-stripe head it may be approximated as $([\sin \pi B\Omega]/[\pi B\Omega])^2$ where $B \triangleq b/(vT)$ is the shield-to-shield spacing normalized to the bit length. For small B, the transition response remains approximately Lorentzian but with effective density $D' \simeq \sqrt{(0.65B)^2 + D^2}$ [89].

Other factors that affect the transition response include the exact shape of written transitions, the geometry of the yoke (including pole-tip effects) and the thickness of MR stripes. These factors are discussed, for example, in [67], [53, Vol. I, Ch. 2], and [89].

Noise This disturbance originates in the medium, the head and the head preamplifier. As a rule, none of these contributions is negligible. A detailed discussion may be found in [18].

Media noise is different for particulate and thin-film media. In particulate media, noise may be analyzed subject to assumptions on uniformity and Poisson packing statistics of magnetic particles [75], [53, Vol. I, Section 5.2.3]. Subject to these assumptions it is found that media noise is additive, stationary and Gaussian, and that its power spectrum, to a first-order approximation, has a bandpass shape akin to that of the amplitude characteristics of the dibit response.

Thin-film media are primarily hampered by transition noise as discussed in Section 2.3.4. This noise is, in general, neither additive nor stationary. It may, however, be remodeled as additive if the noise magnitude is not too large (see Appendix 2C). Furthermore, if the transition response $g(t)$ has negligible excess bandwidth, i.e. if $|G|$ has negligible content for normalized frequencies $|\Omega| > 0.5$, then a stationary model accrues.

Head noise is different for inductive and thin-film heads, but is additive, stationary and Gaussian in both cases. In inductive heads, primary noise sources are hysteretic losses and the thermal resistance of the head coil [53, Vol. I, Section 5.2]. The former losses cause the noise spectrum to increase with frequency. Principal noise sources in

MR heads are thermal resistances of MR stripes and stripe contacts. Both lead to a white noise spectrum.

Preamplifier noise is produced by the electronic circuits that make up the preamplifier, in particular those of the first preamplifier stage [46], [47]. This noise is additive, stationary and Gaussian. Noise spectra tend to be flat in the pass-band of the preamp and may exhibit $1/f$ behavior and noise peaking at the lower and upper band edges, respectively.

Media noise, head noise and preamp noise are mutually uncorrelated and thus add on a power basis. Their combined power spectral density is often rather flat throughout the pass-band of the system. Noise outside this band is normally rejected by a receiver filter and its precise spectral characteristics are, therefore, of no concern. For mathematical convenience it is then reasonable to assume that the noise component of the replay signal is white and indeed this assumption is commonly made. We will see in later chapters that the in-band noise spectrum has a considerable impact on performance and optimal dimensioning of data receivers and for this reason it is important to model this portion of the noise spectrum faithfully.

Crosstalk Recorded track widths are usually less than the track pitch, i.e. an unrecorded strip or *guard band* is left between adjacent tracks. The reason for this is that the head inevitably picks up some flux outside its nominal track width. This *side reading* effect is particularly severe at low frequencies and would cause an untolerable level of adjacent track interference in the absence of guard space, even if the head were perfectly on track. To lower this *crosstalk* to acceptable levels in the face of tracking errors, a guard band of 10 to 50 percent of the track width is commonly used. An alternative approach, widely adopted in helical-scan recording, is called *slant azimuth recording* (Fig. 2.26).

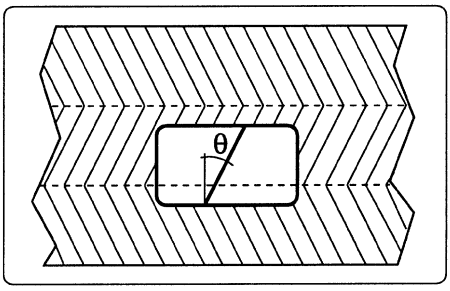


Figure 2.26: Slant azimuth recording. When the head is off track, side reading remains small because the gap of the head is at an angle 2θ with respect to the magnetization direction of adjacent tracks.

Two heads, both with small but opposite slant angles $\pm\theta$, lay down alternate tracks of opposite azimuth angle. The head that reads the home track has a large azimuth error with respect to the two immediately adjacent tracks and this error diminishes crosstalk

substantially, especially at high frequencies. The choice of the slant angle is a compromise between in-track signal-to-noise ratio, which is largest for an angle of 0°, and crosstalk, which decreases as the azimuth error increases. Typical slant angles are in the order of ±10°.

Azimuth losses may be described by a frequency-dependent loss factor [53, Vol. III, Ch. 2] $\alpha \sin[\pi\beta\Omega]/[\pi\beta\Omega]$ where α is the fraction of the adjacent track that is covered by the head and β is a measure of the slant angle θ ($\beta = \alpha W \tan|2\theta|$ were W is the track width normalized to the bit length). Hard disk drives do not use slant azimuth recording and a crosstalk model is discussed in [16].

Crosstalk is basically a filtered version of the data in adjacent tracks, and is as such neither stationary nor Gaussian. Since adjacent tracks are usually written at the same bit spacing as the target track, crosstalk is, in fact, generally cyclostationary with respect to the in-track data, at an *a priori* unknown phase offset. These characteristics are similar to those of the transition jitter model that is developed in Appendix 2C. By analogy, we may regard crosstalk as stationary for the purpose of analysis and optimization of data receivers in the particular but usual event that there is negligible spectral content beyond the Nyquist frequency $1/(2T)$.

Read nonlinearities This distortion occurs primarily in magneto-resistive (MR) heads. Most MR heads in use today sense magnetic flux with a single MR stripe. Heads of this type fundamentally have a nonlinear transfer characteristic. To linearize this characteristic, a bias current that traverses the MR stripe defines an operating point, relative to which flux is sensed. Even so, various anomalies, such as non-optimum bias currents and head imperfections that cause nonlinearities near the bias point, may cause a significant degree of nonlinear distortion [29], [64]. A prevailing effect is *pulse asymmetry*, which causes positive and negative transition responses to have different heights and/or widths. These asymmetries can easily be as large as 30 to 40 percents.

Nonlinear distortion can be reduced significantly by using two MR stripes rather than one [17]. The outputs of both sensors are combined differentially such that the net head output doubles while all even-order distortion, to a first-order approximation, cancels out.

Local media defects All magnetic recording systems exhibit *dropouts* that are due to local media defects. In tape-based systems, dropouts typically induce a floor at around 10^{-5} to 10^{-6} in the raw bit-error-rate characteristic. Signal-to-noise ratios of the head-media system must be compatible with this floor but need not be much better.

In hard disk drives, media defects are detected during drive manufacturing and the corresponding sectors are spared out, i.e. they are not used to store information. New defects may appear during the life time of the drive as a result, for example, of head slaps. For this reason the defect map is not static but is instead updated from time to time.

Other read artefacts The above list of anomalies covers some of the major artefacts but is far from complete. For example, MR heads may exhibit domain instabilities

that cause *Barkhausen noise*, while the output of single-stripe heads may exhibit large thermal transients whenever the head bumps into an asperity on the disk [73]. As another example, *rubbing noise* may be generated when a ferrite head is in sliding contact with the medium. In helical-scan systems, the rotating heads are usually coupled to the outside world via rotating transformers. These may induce crosstalk and saturation effects.

The design of the head/media system involves an intricate tradeoff between the entire disturbance constellation on the one hand and system complexity and cost on the other. The signal processing techniques that are discussed in later chapters are perhaps best regarded as means to facilitate this tradeoff.

2.4 DIGITAL OPTICAL RECORDING

Optical recording dates back to the early seventies [96]. First products to be introduced were video disc systems but the big commercial breakthrough was caused by Compact Disc[2] (CD), which was launched in 1983 [113]. CD uses prerecorded, replicated discs to store digital audio at an information density of about 1 μm^2/bit. The success of CD has spurred the development of derivatives such as CD-ROM (read only memory), CD-I (interactive) and CD-V (video). CD-ROM has overtaken floppy disk as the primary distribution medium for software. The capacity of a CD-ROM (650 MB per disk) is Gargantuous compared to that of a floppy disk, but is nevertheless smaller than desirable for multimedia applications. The successor of the original CD standard, designated DVD (digital versatile disc), enlarges storage capacity by roughly an order of magnitude [93], [100], [101]. Key enablers of DVD are the commercial viability of diode lasers with a shorter wavelength than those in CD, and the use of multiple recording layers on a single disc (see Section 2.4.4).

CD and its derivatives are all meant to replay prerecorded information, and do not permit information to be written and/or erased. The early eighties brought systems that allowed data to be written but not erased (WORM: Write Once, Read Many). Rewritable storage systems emerged more recently.

WORM drives normally use a high-power laser that burns holes in the medium. This is an *ablative* recording technique. A second technique, applied in CD-recordable (CD-R), uses an organic polymer on which marks are recorded by means of a phase-change process.

Rewritable optical storage systems also come in various tastes. The most important one is *magneto-optical* (MO) storage. Here the storage medium is magnetic but writing and reading are accomplished with optical means. MO media have a critical temperature, designated *Curie temperature*, above which their magnetization vanishes. To write information, the medium is heated beyond the Curie temperature by means of a high-power laser. As the material cools down, the magnetization returns and aligns with an

[2]In optical recording, the Anglo-saxon spelling 'disc' is normally used in lieu of the American spelling 'disk'. Similarly, storage densities in optical recording are normally expressed in Mb/cm^2 or μm^2/bit, as opposed to Mb/I^2 or Gb/I^2 in hard disk drives.

external magnetic bias field. This field is weak enough so that it cannot alter the magnetization of the MO layer at room temperature. Reading is based on the Kerr effect, i.e. on the magnetization-dependent rotation of the polarization direction of monochromatic light that is reflected off the medium.

Rewritable storage can also be based on *phase-change* techniques. Here the media is an alloy with a crystalline and an amorphous phase. To form a mark, a spot of crystalline material is momentarily melted by a laser pulse. Then the spot cools quickly into an amorphous phase mark, whose reflectivity differs from the surrounding, crystalline-phase material. Recorded marks can be erased through an annealing process, during which a continuous erase beam heats the material just below its melting point, returning the material to its crystalline state.

The simplicity of phase-change technology gives it an edge over MO technology, which requires more complicated optical paths and electronics. A disadvantage is the restricted rate at which the material can change phase, which constrains the data-writing rate. MO technology is used, for example, in various data recording systems [108], in the MiniDisc audio recording system [126] and in the VDR professional video recording system [100], while phase-change techniques are used in CD-erasable (CD-E).

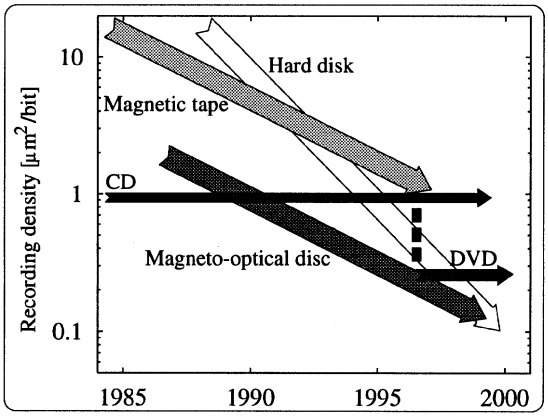


Figure 2.27: Evolution of areal densities for various digital storage systems.

In comparison to hard disk drives, rewritable optical drives feature removability, at the cost of a slower access time which is due in part to a larger moving mass of the optical read/write head. Recording densities in magnetic storage used to lag far behind those in optical storage, but this gap has gradually disappeared (Fig. 2.27). A principal reason for the comparatively slow evolution of optical storage densities in Fig. 2.27 is the slow pace at which the wavelength of laser diodes has improved; this wavelength largely determines the smallest optical details that can be resolved. Another factor is the time-consuming standardization process that precedes any new storage product with removable media. This 'committee overhead', which also plagues tape storage systems, is non-existent for hard disk drives.

2.4.1 Optical recording technology

Fig. 2.28 depicts some of the basic elements of an optical disc player. The digital information on the disc is contained in microscopically small lands and pits arranged in a spiral path. These details are scanned by a beam of light, from a semiconductor laser, which is reflected from the surface of the disc on to a detector. The arm contains all the facilities necessary to position the beam precisely. The motor must rotate the disc at the correct speed. Areal recording densities are often constant across the disc and if the bit rate is to be constant then the angular velocity of the disc must decrease gradually from the innermost to the outermost tracks. The unit containing the optical components is sometimes called the *optical pick-up* or *light pen* of the system (Fig. 2.29).

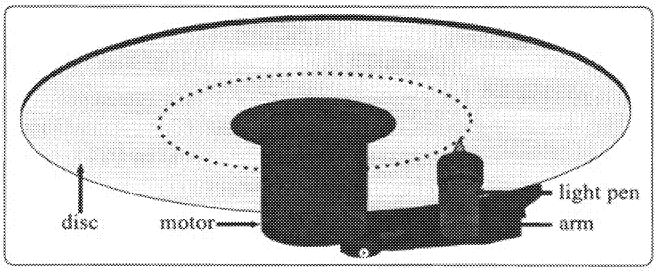


Figure 2.28: Main components of an optical disc player.

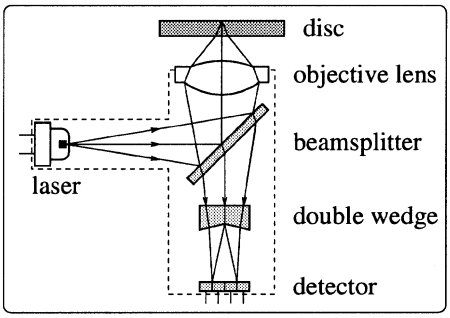


Figure 2.29: Typical construction of the light pen in a CD player.

Light of the semiconductor laser is focused on the disc, reflected back, and detected by means of a detector. This detector is a multiple sensing device, i.e. apart from information about the recorded data pattern it also delivers information about focusing and positioning of the beam with respect to the optical track on the disc. Various physical principles are used to induce variations in the reflected light beam:

- differences in level between pits and lands (as in CD-like systems) give differences in intensity on reflection;

- differences in magnetization of a recording film (as in MO recording) affect the polarization of the reflected light;

- local differences in crystal structure (as in phase-change recording) give differences in intensity on reflection.

Servo The depth of focus of the laser beam is typically in the order of 2μm. It is completely impractical to specify the flatness and alignment of the disc to this degree. Instead, the separation of the objective lens and the disc is continuously adjusted during writing and read-out. This adjustment is made using a servo system that controls an electromechanical actuator for the lens movement. Several methods exist to measure the amount and direction of defocus [95, Section 2.5]. Two commonly used ones are the *Foucault knife-edge method* and the *astigmatic method.* Both use multiple photo detectors whose outputs are combined in such a manner that an S-shaped focus error signal emerges which can be used as a basis for control of the objective lens.

A further dynamic measurement included in the optical system is needed to keep the laser beam on track. The track pitch is typically in the order of 1 to 2 μm and the laser spot should be centered on the track to within 0.1 μm. Several methods are used to detect tracking errors [95, Section 2.5], [108], [126]:

1. The *three-spot method* employs a grating structure in the light path to generate two satellite spots from the main beam. The relative intensity of these satellite spots is a measure of the tracking error.

2. The *push-pull method* relies on the optical interference of the zeroth and two first-order beams, diffracted from the track structure, in the pupil of the optical system. A tracking-error signal is obtained by differencing the outputs of two detectors that are split along the plane of symmetry of the the radial diffraction pattern.

3. The *pre-groove method* uses shallow prerecorded grooves that are located halfway between the data tracks. The spot is on track when both adjacent tracking grooves contribute equally to the detector output. A variation of this approach, used in MO recording [126], uses a single groove in which the magnetization pattern is recorded.

4. The *sampled-servo method* is the optical counterpart of the embedded servo approach of hard disk drives. It uses special patterns of optical effects that are written on the disc at predetermined intervals. The read-out of the patterns is used to adjust tracking and time-base errors. The servo system is then disabled until the next special pattern is encountered.

The first two techniques can only be used during playback, while the latter two can, in principle, also be used during recording.

2.4.2 The recording process

Read-only systems For read-only systems such as CD, discs are prerecorded via a *mastering* process. Information is cast in the form of a height profile in a layer of aluminum. It is through this profile that mass replication of the information is relatively simple (see [95, Chapter 5] for a review). A typical recorded pattern of pits and lands is depicted in Fig 2.30.

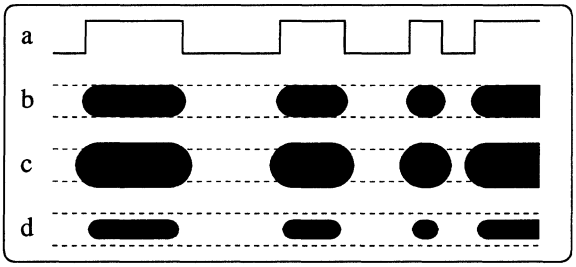


Figure 2.30: Example write signal (a) and corresponding recorded pattern of pits and lands on a read-only disk for nominal mastering conditions (b) and in the presence of over-etching (c) and under-etching (d).

Pits and lands correspond to the ones and zeros of the data sequence. They are nominally equally long for strings of ones and zeros of equal length. In practice it is difficult to accurately control the size of recorded domains. For example, over-etching and under-etching during the mastering process will cause recorded pits to be systematically too large or too small (Figs. 2.30.c and d).

Magneto-optical systems In order to be able to write data, MO systems need added provisions on top of the read-out system discussed above. Specifically, a bias coil is needed to produce the magnetic MO bias field, and the diode laser must have a high-power mode to permit the medium to be heated beyond the Curie temperature. Data may be written by switching the laser on and off in accordance with the data pattern. This is called *laser modulation.* The simplest manner to rewrite data is to first erase a complete track and to write the new data on a subsequent disk revolution. This method is painfully slow and has spurred the development of various *direct overwrite* techniques, which avoid a separate erase cycle [104], [107], [120], [115]. The most popular approach is called *magnetic field modulation* [120]. Here the external magnetic field is modulated by the data stream (Fig. 2.31).

The magnetic head is located opposite to the optical head and is close to the magnetic layer or in contact with it. The laser is either kept on all the time or pulsed continuously so as to heat the medium within the laser spot above the Curie temperature. The recorded magnetization is determined by that of the bias field at the point where

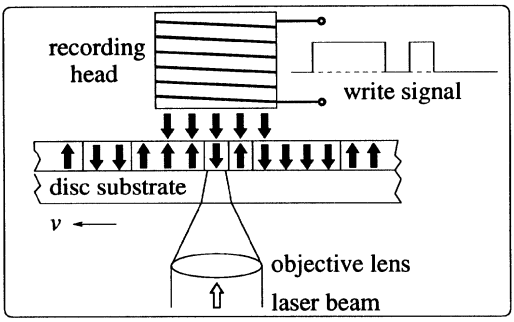


Figure 2.31: Magnetic field modulation overwrite scheme.

the medium cools below the Curie temperature. This mechanism produces recorded magnetization patterns as in Fig. 2.32.

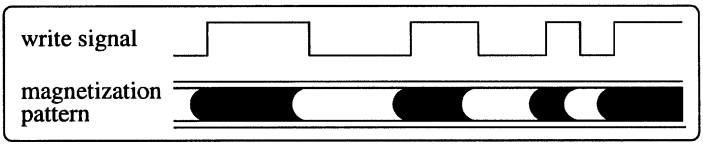


Figure 2.32: Example write signal and corresponding magnetization pattern for magnetic field modulation.

Magnetic field modulation is inherently symmetric for ones and zeros, i.e. it does not suffer from the asymmetry that plagues read-only recordings (compare Figs. 2.30.c and d). Achievable recording densities are significantly higher than for laser modulation because recorded transitions need to occupy only a small fraction of the spot width. Unlike laser modulation, the recorded magnetization pattern is quite insensitive to tilt of the disc and distortion and defocusing of the laser spot as it falls on the disk [126].

Write imperfections Ideally the recorded domains are a perfect spatial copy of the binary write signal. In practice various anomalies cause a detraction from this ideal. We now discuss some of the major imperfections.

Domain bloom *Asymmetry* or *domain bloom* is an imperfection in the disc which is due, among other factors, to under- or over-etching during the mastering process and which causes pits to be systematically too large or too small as in Figs. 2.30.c and d. On poorly manufactured discs, this phenomenon can cause pit-to-land and land-to-pit transitions to shift by a large fraction of the length of a single bit. This is an important reason why CD-like systems use a *run-length-limited* modulation code. In such a code, runs of like symbols are constrained to have a minimum and maximum duration. The CD modulation code, called EFM, has a minimum run-length of 3 symbols and on the

disc this length can shrink to only 1.3 as a result of over-etching or under-etching [118]. If no minimum run-length were used in this situation, then runs of one symbol would hardly leave a trace on the disc (see Problem 2.5).

In MO systems, domain bloom is virtually absent with magnetic field modulation but can be significant with laser modulation. In the replay signal, bloom manifests itself in the form of nonlinear intersymbol interference (see Example 2.2 and Appendix 2D). A major portion of this ISI is a DC-shift.

Transition jitter The edges of the recorded domains are not always sharply defined, and tend to exhibit random displacements with respect to their ideal position. This transition jitter may be modeled as in Appendix 2C. Of particular interest is the model of Fig. 2C.3.

Pinholes and wedges This anomaly occurs mainly in prerecorded discs and is caused by imperfections in the aluminum layer. Pinholes in this layer induce *dropouts* which may cause the replay signal to vanish during tens or even hundreds of bit intervals. A similar effect is caused by scratches on the layer which are called *wedges* in CD terminology. Error protection techniques are indispensable for protecting the recorded user data against these anomalies.

Other write imperfections MO recording is essentially a form of magnetic recording and therefore the write imperfections described in Section 2.3.4 can occur here as well. With laser modulation, thermal effects can cause various additional imperfections [108, Section III], [126].

2.4.3 The replay process

At the heart of the optical replay process is a focused optical beam that scans the recorded data pattern. The smallest details that can be resolved are in the order of the spot diameter d. If we define d as the full width at half maximum of the light intensity, then $d \simeq 0.55\lambda/\mathrm{NA}$ where λ is the wavelength of the laser light and NA is the numerical aperture of the objective lens [95, Chapter 2]. To obtain a high information density in the radial and tangential directions, d must be small. Mass-produced semiconductor lasers all have a wavelength in the order of 800 nm. Significantly shorter wavelengths have been demonstrated experimentally but are not yet commercially viable. The numerical aperture is typically in the order of 0.45 to 0.55. Larger values are, in principle, feasible but cause a rapid degradation of various tolerances. For example, the manufacturing tolerance to the disk thickness is proportional to NA^{-4} and the depth of focus is proportional to NA^{-2} [95, Chapter 2]. In practice, then, spot sizes d are in the order of 0.7 to 1 μm.

A significant feature of the read process, both for read-only and for MO systems, is that it responds to DC. That is, the read signal makes a transition from low to high or *vice versa* at the edges of the domains or pits, and remains high or low in between

(Fig. 2.33). This is unlike magnetic recording, where the read head basically responds to recorded transitions (compare Fig. 2.23). It should be mentioned that certain optical detector arrangements also permit edge detection (see, for example, [95, Chapter 2] and [106]), but these arrangements have not found widespread application.

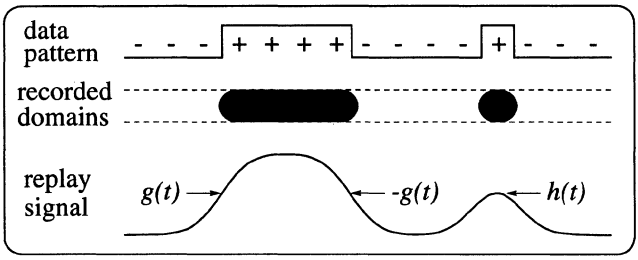


Figure 2.33: Illustration of typical replay waveforms in optical recording.

Just as in magnetic recording, the read process in optical recording is essentially linear. The optical intensity profile of the spot is usually assumed to be Gaussian. Furthermore, the replay signal is normally modeled as a point-to-point multiplication of the spot profile and the recorded data pattern. These first-order models, however crude, tend to correlate well with scalar diffraction calculations and, more importantly, with measured read signals from actual storage systems.

As the spot scans the leading edge of a recorded domain, the replay signal exhibits an upgoing transition $g(t)$ that can be thought of as the *transition response* of the recording channel. The derivative $f(t)$ of $g(t)$ is the *impulse response* of the channel. This is basically a temporal version of the Gaussian spot profile according to

$$f(t) = \frac{2}{t_0\sqrt{\pi}}\exp\{-(\frac{2t}{t_0})^2\} \tag{2.6}$$

where t_0 characterizes the $1/e$-width of the spot. Individual bits of the write signal $s(t)$ are carried by rectangular pulses $c(t)$ of width T. The response $h(t)$ of the channel to a single pulse $c(t)$ is called the *symbol response*. By linearity, $h(t)$ is just the convolution of $c(t)$ with $f(t)$, i.e. $h(t) = (c * f)(t)$. We arrive at the model of Fig. 2.34. Here the recorded data sequence a_k is first converted into a binary write signal $s(t)$ via a linear pulse modulator with symbol response $c(t)$. The data component $x(t)$ of the replay signal is obtained by applying $s(t)$ to a linear filter with Gaussian impulse response $f(t)$.

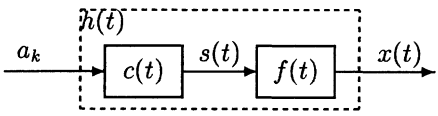


Figure 2.34: Linear model of the data component $x(t)$ of the optical replay signal.

For the sake of completeness we mention that $h(t)$ can also be thought of as the response of the channel to two successive transitions according to $h(t) = g(t) - g(t - T)$. This representation is, however, less natural than for magnetic recording because of the fact that $g(t)$ now has an infinite extent.

By definition, $g(t)$ is the response to a 0-to-1 transition. A -1-to-$+1$ transition has magnitude 2 and yields a response $2g(t)$ plus a DC shift. Fig. 2.33 is somewhat deceptive in that the corresponding response is labeled $g(t)$ for notational convenience. In similar vein, the pulses that are labeled symbolically $-g(t)$ and $h(t)$ are actually twice as large.

Linear ISI Successive bits a_k will interfere with each other when the effective duration of $f(t)$ exceeds the symbol spacing T. The severeness of this ISI increases with the ratio $S \triangleq t_0/T$. By analogy to magnetic recording systems, we refer to S as the *normalized information density* of the recording channel. Practical values for S are in the order of 0.5 to 2. In terms of S we may express $f(t)$ as

$$f(t) = \frac{2}{ST\sqrt{\pi}} \exp\{-(\frac{2t}{ST})^2\}. \tag{2.7}$$

The Fourier transform of $f(t)$ characterizes the transfer function of the optical read-out path and is sometimes called *modulation transfer function* (MTF). It is given by

$$F(\Omega) = \exp\{-(\frac{\pi S\Omega}{2})^2\} \tag{2.8}$$

where Ω is a normalized measure of frequency, equal to 1 for the symbol rate $1/T$. The MTF exhibits a much more rapid roll-off than the Lorentzian transfer characteristic of a magnetic recording channel (compare (2.5)). One reason for this is that the light which carries the high-frequency components gets diffracted so that it does not reach the detector. A sharp cut-off occurs at a frequency v/d where v is the medium velocity and d is the spot size. This sharp cut-off is not present in (2.8) but even so a Gaussian function is a reasonable approximation to the actual MTF. If the effects of the rounded ends of written marks are also accounted for then the MTF gets even closer to Gaussian.

For the adopted frequency normalization, the linear pulse modulator of Fig. 2.34 has transfer function $C(\Omega) = T\sin(\pi\Omega)/(\pi\Omega)$, and the transfer function of the entire channel is

$$H(\Omega) = C(\Omega)F(\Omega) = T\frac{\sin \pi\Omega}{\pi\Omega} \exp\{-(\frac{\pi S\Omega}{2})^2\}. \tag{2.9}$$

Fig. 2.35 illustrates this function along with the corresponding symbol response $h(t)$ for various values of S.

The width of the symbol response increases rapidly with S, and so does ISI. At the same time the amplitude of the response decreases. This causes signal-to-noise ratios to decrease. The transfer characteristics of Fig. 2.35 all have a low-pass nature. The

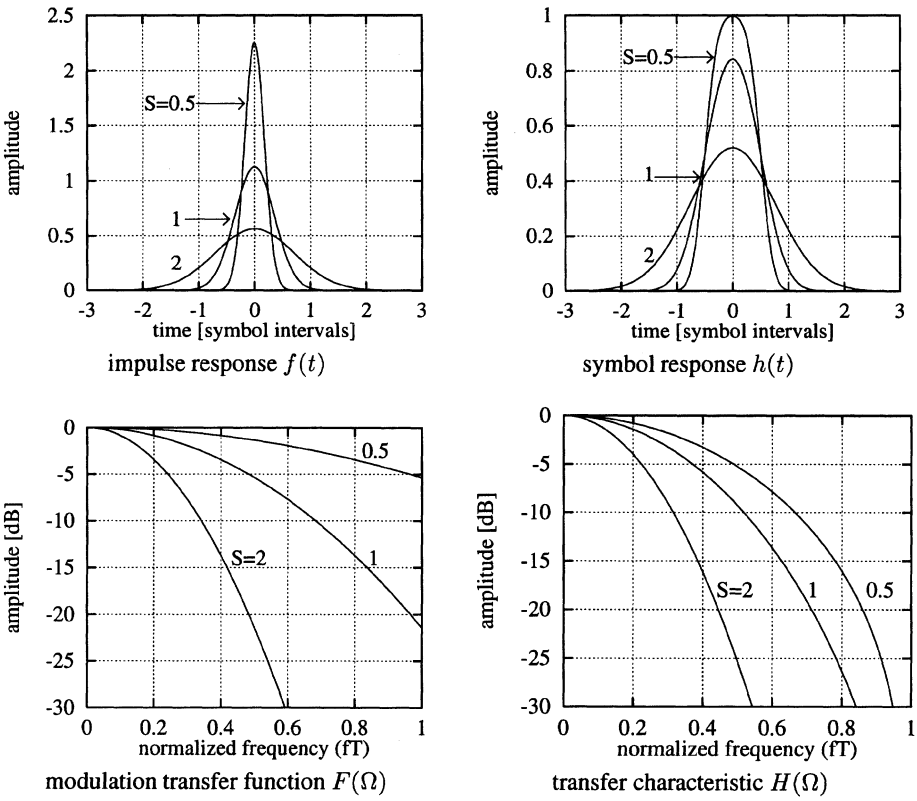


Figure 2.35: Symbol response $h(t)$ and transfer characteristic $H(\Omega)$ of an idealized optical recording channel for various values of the normalized information density $S = t_0/T$. Also depicted are the impulse response $f(t)$ and the modulation transfer function $F(\Omega)$.

transfer loss at high frequencies increases rapidly with S. The channel is, therefore, best suited to convey low data frequencies. Even so, frequencies near DC are often suppressed so as to prevent data from interfering with the servo information. This is done via a DC-free modulation code (see Section 4.9).

The responses of Fig. 2.35 are only approximations (albeit fairly reasonable ones) of those in actual recording systems. They pertain, moreover, to the nominal conditions of the optical path. In reality these conditions may deviate from nominal and as a result the responses will change. For example, disc unevennes and disc wobble may induce dynamic defocusing errors. Such errors cause additional high-frequency losses that may be characterized by a loss factor [99, page 124] $[\sin 2\pi\lambda S\Omega(1 - S|\Omega|)]/[2\pi\lambda S\Omega(1 - S|\Omega|)]$. Here λ is a normalized measure of the degree of defocusing, a value of 1 indicating defocusing by about ± 2 times the spot size. The impact of these added losses on system performance may be lowered via a modulation code with reduced high-frequency

content (see Examples 4.4 and 4.10 for illustrations). Alternatively, the variations can be counteracted adaptively in the receiver (see Chapter 8).

Other deviations that affect the amplitude- and/or phase characteristics of the modulation transfer function include tracking errors, chromatic aberration and astigmatism of the objective lens (see [95, Chapter 2] for more details).

Nonlinear ISI As noted earlier, the optical read process is essentially linear. The primary cause of nonlinear ISI is domain bloom that originates in the write process. This ISI is analyzed in Appendix 2D. Principal component is a DC-shift. All other components are of second order and are determined by adjacent bits. The DC-shift can be accommodated at the receiving end of the system by means of an adaptive detection threshold [118], or it can be compensated via adaptive DC-compensation techniques (see Section 8.8.2). The other ISI components are more cumbersome and require more complicated countermeasures, such as nonlinear sequence feedback as in Example 7.12 (see also [94]). As explained above, the use of a minimum run-length constraint tends to reduce the impact of bloom on the replay signal considerably.

Example 2.2 The CD system uses a modulation code called EFM for which runs of identical bits a_k have a length of at least 3. For a code of this type and a normalized information density $S = 2$, Fig. 2.36 depicts eye patterns of the replay signal for a situation without domain bloom and two situations with severe bloom ($\gamma = \pm 1$ in the nomenclature of Appendix 2D).

In the absence of domain bloom, there is some ISI but the eye pattern has a large eye opening and is completely symmetric with respect to DC. Bit decisions can be taken by a threshold detector whose decision threshold is set at DC.

Domain bloom primarily affects the innermost eye traces of one polarity. It causes these traces to shift towards DC, thereby decreasing the eye height. The decision level must be shifted by half this amount so as to remain in the center of the eye. This can be accomplished adaptively [118] (see also Section 8.8.2). Even for the severe degree of bloom in Figs. 2.36.b and c, sufficient eye opening is left to permit correct decisions. If no minimum run-length constraint were used, by comparison, then the eye patterns would silt up completely and the system would break down (see Problem 2.5).

Noise In optical recording, noise in the replay signal has both additive and multiplicative components. Noise sources include the medium, laser, detector and preamplifier.

Media noise is primarily due to transition jitter and may be modeled as additive (see Appendix 2C). Laser noise is due essentially to fluctuations of the light intensity. It is usually multiplicative, i.e. it takes the form of random fluctuations of the amplitude of the replay signal. It is caused by light that reflects from optical components or the medium back into the laser [123]. The photodetector and preamplifier both produce additive noise. This noise is relevant mainly to MO systems, where detected optical effects are much smaller than for read-only systems. Other types of noise include shot noise, polarization noise and noise that arises due to the roughness of the disc surface. A review of these and other noise sources may be found in [109], [123] and [98].

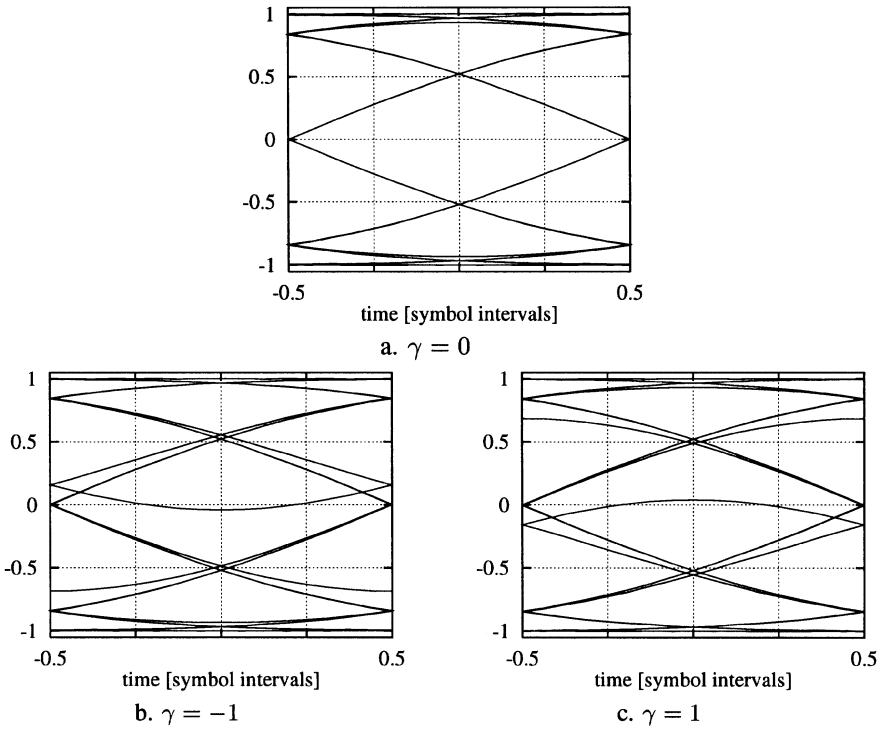


Figure 2.36: Eye patterns of replay signal for data with a minimum run-length of 3 symbol intervals and a channel with normalized information density $S = 2$ in the absence of bloom (a) and for the situations that pits are two symbol intervals longer (b) or shorter (c) than the lands.

In read-only systems, transition jitter is typically the dominant type of noise. In MO systems, on the other hand, detector and preamplifier noise tend to be significant. The compound noise component of the replay signal is sometimes modeled as additive, Gaussian and white [116], [108]. Although this model is convenient, it may not be very accurate, especially for read-only systems.

Other read artefacts Further impairments of the replay signal include cross-talk from neighboring tracks (see [95, Chapter 2] for an analysis), dropouts that are caused by dust, fingerprints and scratches on the disc surface, and dynamic variations of the bit rate that are caused by disc eccentricity [118].

2.4.4 Prospects

In the past, storage densities in optical recording have improved at a much slower pace than those in hard disk drives (see Fig. 2.27). Novel application areas such as

multimedia and digital video demand large increments in storage capacity and this commercial impetus now spurs a whirlwind of activity. Major developments include:

Improved lasers: In the past decades, the wavelength of commercially available semiconductor lasers has improved from about 780 nm to some 670 nm, and further improvements are actively pursued. Attainable storage densities improve quadratically with the wavelength and even small wavelength reductions are, therefore, of interest.

Multilayer recording: Here multiple recording layers are stacked on top of each other. The layers are separated by a thicker photopolymer and read by a single laser beam, which can be focused on any of the layers. An example of this technique [101] is found in DVD, which uses two layers.

Improved format efficiencies: Advances in coding and synchronization as well as in VLSI technology permit the use of error protection, modulation coding and block synchronization schemes that require less overhead. Also, improved production techniques permit various margins to be tightened. As a result of these measures, storage densities may more than double [101].

Improved signal processing techniques: Early optical recording systems such as CD employ run-length-limited modulation codes with a high degree of redundancy. These codes render the replay signal quite insensitive to the various imperfections of the optical recording channel, thereby facilitating reliable data reception. Advances in signal processing, as described in the forthcoming chapters, have led to data receivers that require little if any reception-easing overhead in the modulation code. As a result, code redundancy can decrease so as to improve storage densities.

Optical superresolution: This technique modifies the optical power distribution of the beam focused on the optical disk. Power is shifted away from the center of the beam toward the edges, producing -through diffraction- a smaller focused spot. This smaller spot can resolve smaller features on the disk, allowing capacity increments by 20 to 30 percents [125], [111].

Magnetic superresolution: This technique is specific to MO recording and is based on the use of double-layered exchange-coupled media [92]. Bits are stored at high density in a buried memory layer, which is covered by a readout layer. During readout, the focused laser beam heats a small area of the readout layer to a temperature that is high enough for data to be copied from the memory layer to the readout layer. In this way the data becomes visible to the readout mechanism. Since only a fraction of the laser spot produces temperatures that are high enough for exchange coupling to take place, only data in a submicron area is seen and resolved.

Multi-track recording and cross-talk cancellation: With the help of laser arrays and multiple detectors the optical system can, in principle, be designed for multiple tracks to be written and/or read in parallel. This technique increases the throughput of the system. It also permits tracks to be spaced closely together, provided that the ensuing cross-talk between tracks is handled by means of a cross-talk canceler. In such a

canceler, the unwanted signal from the neigbouring tracks is mimicked electronically and subtracted from the desired signal [103]. This technique is very similar to echo cancellation as discussed in Section 2.2.1.

Optical tape recording: In existing optical storage products, information is stored on a removable disc. The last frontier in optical recording is storage on tape. Optical tape recording is currently an area of active research, and various systems have been proposed (write-once: [119], [105]; rewritable: [112], [122]). Among the most elegant ones is the system of [114], which is based on a very fast rotating polygon and permits storage densities in the order of 1 μm^2/bit.

2.5 FURTHER READING

Digital subscriber lines: The March 1986 issue of IEEE Communications Magazine focuses on narrowband ISDN. A survey of transmission techniques for narrowband ISDN may be found in [10]. An excellent introduction to echo cancellation for baseband data transmission is [12]. The August 1991 issue of the IEEE Journal on Selected Areas in Communications is entirely devoted to HDSL and ADSL. The article of Werner in that issue [14] provides a detailed survey of all of the disturbances that were mentioned in Section 2.2.1. The historical perspective of Lechleider [8] contains an encyclopedic list of references. A recommended introduction to multitone modulation is [2]. A tutorial introduction to ADSL may be found in [9]. The December 1995 issue of the IEEE Journal on Selected Areas in Communications contains a range of articles on copper wire access technologies for high-performance networks.

Digital magnetic recording: An in-depth treatment of many of the aspects and applications of digital magnetic recording may be found in the series of 3 books edited by Mee and Daniel [53]. Other books that cover aspects of magnetic recording are [19], [26], [81], [43], [71], [31], [38], [78], [42], [55], [18], and [52]. Tutorial articles on the subject include [51], [35], [85], [22], [87], and [76]. Historical perspectives may be found in [82], [38], [35], [36], and [26]. Projections on the evolution of digital magnetic recording appear in [50], [70], [83], and [49]. The January 1992 issue of the IEEE Journal on Selected Areas in Communications and the March 1995 issue of IEEE Transactions on Magnetics are both entirely devoted to signal processing techniques for digital recording. Other articles on these techniques include [57], [86], and [30].

Digital optical recording: The November 1993 issue of the Japanese Journal of Applied Physics is entirely devoted to optical recording and discusses a broad spectrum of recent developments, ranging from physical to system-level. The collected papers of [117] give a good overview of major developments in optical recording. Books on the subject include [95], [102], [110], and [124]. Articles with a tutorial nature include [121], [108], [93], and [100]. Signal processing techniques for optical recording are discussed in [95, Chapter 7], [113], [116], [118], and [108].

APPENDIX 2A

Nonlinear bit-shifts due to bandwidth limitations of the write path

Fig. 2A.1 depicts a highly simplified model of the write process. A binary data sequence $a_k \in \{-1, 1\}$ of data rate $1/T$ is to be recorded. It is first converted into a ternary sequence $b_k = a_k - a_{k-1} \in \{-2, 0, 2\}$ that characterizes the data transitions. Specifically $b_k = +2$ when there is an upgoing data transition between instants $k-1$ and k, $b_k = -2$ when there is a downgoing transition, and $b_k = 0$ when there is no transition. A linear pulse modulator operates on b_k to produce the head field $H_g(t)$. The symbol response $p(t)$ of the modulator is the response of the head to a unit step in write current as in Fig. 2.15. The time-dependent media magnetization $z(t)$ is determined by $H_g(t)$ via a square hysteresis loop with threshold levels $\pm H_t$ where H_t is somewhat larger than the media coercivity H_c (compare Fig. 2.16). Transitions cannot be written unless the peak value of $|H_g|$ exceeds H_t.

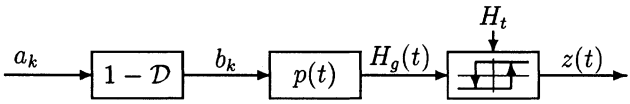


Figure 2A.1: Simplified model of the write process.

It should be stressed that the model of Fig. 2A.1 is rather crude in that it neglects such effects as head saturation, the nonlinear dependence of the size of the write bubble on the head field H_g, and non-squareness of the remanence loop of the medium. Even so, it captures some of the essential features of the write process while being simple enough to permit a mathematical analysis.

Average position of transitions The head field $H_g(t)$ may be expressed as

$$H_g(t) = \sum_{k=-\infty}^{\infty} b_k p(t - kT) = \sum_{k=-\infty}^{\infty} a_k q(t - kT) \tag{2A.1}$$

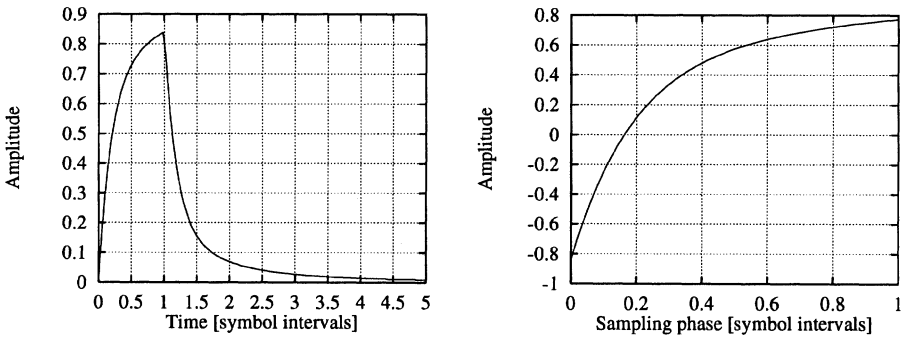
where $q(t) \triangleq p(t) - p(t - T)$ is the symbol response of the entire write path up until the head. We first consider an upgoing data transition between instants $k - 1$ and k ($a_{k-1} = -1$, $a_k = 1$, $b_k = +2$). In response to this transition, H_g will increase and after some delay will exceed the threshold level H_t. This delay must be strictly smaller than one symbol interval or else minimum-width bit-cells cannot be written. We are thus led to consider an instant $t_k \triangleq (k + \phi)T$ where $\phi \in [0, 1]$ is a prescribed sampling phase (normalized in symbol intervals T). Clearly $H_g(t_k) = (a * q^\phi)_k$ where $q_k^\phi \triangleq q((k + \phi)T)$. This is a causal impulse repsonse and since $a_k = 1$ and $a_{k-1} = -1$ we

have

$$H_g(t_k) = q_0^\phi - q_1^\phi + \sum_{j\geq 2} a_{k-j} q_j^\phi. \tag{2A.2}$$

We see that $H_g(t_k)$ does not depend on any of the future bits $a_{k+1}, a_{k+2}, \ldots$, but that it does depend on the past bits $a_{k-2}, a_{k-3}, \ldots$. It is this dependence that causes nonlinear bit-shifts. For zero-mean data a_k, the average contribution of these past bits amounts to zero so that $E[H_g(t_k)] = q_0^\phi - q_1^\phi$. In order for ϕ to demarcate the average position of threshold crossings we must require that $E[H_g(t_k)] = H_t$, i.e. that $q_0^\phi - q_1^\phi = H_t$. To solve this expression for ϕ we may plot the function $F(\phi) = q_0^\phi - q_1^\phi$ for $0 \leq \phi \leq 1$ and determine its intersection with the threshold H_t. If no such intersection exists then the write current is too small for all bits to be properly recorded. For reasons of symmetry the average position of downgoing transitions will be the same as that of upgoing transitions.

Example 2.3 Fig. 2A.2 depicts the symbol response $q(t)$ and the function $F(\phi) = q(\phi T) - q((1+\phi)T)$ for the step response of Fig. 2.15 at a data rate $1/T$ of 100 Mb/s. In the eye pattern of Fig. 2.16.a the hysteresis threshold H_t is in the order of 0.5 (relative to the peak value of H_g). Clearly $F(\phi) = H_t$ for $\phi \simeq 0.5$. This is indeed the average sampling phase at which transitions are recorded (compare the squares in Fig. 2.16.a).



a. Symbol response $q(t)$. b. $F(\phi) = q(\phi T) - q((1+\phi)T)$.

Figure 2A.2: Two functions that pertain to the step response of Fig. 2.15.

Transition shifts Magnetization transitions all occur in the vicinity of the average sampling phase $\overline{\phi}$. We first consider an upgoing transition ($a_{k-1} = -1, a_k = 1, b_k = 2$). The upgoing traces in the eye pattern of Fig. 2.16.a all have essentially the same slope λ in the vicinity of the threshold H_t. This allows us to approximate the transition shifts Δ_k as $\Delta_k \simeq -\lambda^{-1}[H_g(t_k) - H_t]$ where $H_g(t_k) - H_t$ is the 'intersymbol interference' of $H_g(t_k)$ at the average sampling phase $\overline{\phi}$. At this phase we have $q_0^{\overline{\phi}} - q_1^{\overline{\phi}} = H_t$ so that

$$H_g(t_k) - H_t = \sum_{j\geq 2} a_{k-j} q_j^{\overline{\phi}}. \tag{2A.3}$$

A similar derivation is possible for a downgoing transition ($b_k = -2$) and it is found that Δ_k is inverted relative to $b_k = +2$ (for the same bits $a_{k-2}, a_{k-3}, \ldots$). In the absence of a transition Δ_k becomes undefined and we can arbitrarily set Δ_k to zero. To cover all three situations we can, therefore, use a premultiplication factor $b_k/2$ so that

$$\Delta_k \simeq -\frac{b_k}{2} \sum_{j \geq 2} \gamma_j a_{k-j} \tag{2A.4}$$

where

$$\gamma_j \triangleq \lambda^{-1} q_j^{\overline{\phi}}. \tag{2A.5}$$

Past bits a_{k-j} contribute to Δ_k in proportion to $q_j^{\overline{\phi}} = q((j + \overline{\phi})T)$. As suggested by Fig. 2A.2.a, this sequence of coefficients tends to vanish rapidly with j, i.e. only the first few coefficients γ_j ($j \geq 2$) tend to be significant.

It is common practice to express nonlinear bit-shifts exclusively in terms of past transitions $b_{k-1}, \ldots, b_{k-2}, \ldots$ (as opposed to past bits $a_{k-2}, a_{k-3}, \ldots$). Problem 2.6 serves to recast (2A.4) in this form. It is found that

$$\Delta_k \simeq \frac{b_k}{4} \sum_{j \geq 0} \varepsilon_j b_{k-j} \tag{2A.6}$$

where

$$\varepsilon_j \triangleq 2\lambda^{-1}[p_\infty^{\overline{\phi}} - p_j^{\overline{\phi}}] \text{ for all } j \geq 1 \text{ and } \varepsilon_0 \triangleq \frac{\varepsilon_1}{2}. \tag{2A.7}$$

Here $p_j^{\phi} \triangleq p((j + \phi)T)$ denotes the sampled step response of the write path up until the head. Clearly $q_j^{\phi} = p_j^{\phi} - p_{j-1}^{\phi}$ for all j. The term $\varepsilon_0 b_k^2/4$ basically represents a fixed delay because it amounts to ε_0 for $b_k = \pm 2$, while Δ_k can be defined arbitrarily for $b_k = 0$. The only real data dependence in Δ_k is brought about by the components $\varepsilon_j b_k b_{k-j}/4$ with $j \geq 1$, which are said to be j^{th} order bit-shifts. These shifts are ternary ($\varepsilon_j b_k b_{k-j}/4 \in \{-\varepsilon_j, 0, \varepsilon_j\}$) and their magnitude ε_j is fully determined by the extent to which $p(t)$ has approached its limiting value $p(\infty)$ at the instant $t = (j + \overline{\phi})T$. As suggested by Fig. 2.15, $p(t)$ tends to creep slowly towards $p(\infty)$ so that many coefficients ε_j tend to be significant. In this regard the representation of (2A.4) is more 'canonical' than the one of (2A.6).

Exercise 2.2 Prove that $\gamma_j = \frac{1}{2}[\varepsilon_{j-1} - \varepsilon_j]$ for all $j \geq 1$. Assume that ε_j tends slowly towards zero as j increases. Argue that γ_j vanishes rapidly in this case.

APPENDIX 2B

Nonlinear bit-shifts and write precompensation

In this appendix we are concerned with models and write precompensation for nonlinear bit-shifts that depend on the preceding two transitions. In the absence of bit-shifts

and noise, the data component $x(t)$ of the replay signal assumes the form (2.3) where $b_k = a_k - a_{k-1} \in \{-2, 0, 2\}$ is the transition sequence, $g(t)$ is the transition response, and $a_k \in \{-1, 1\}$ is the recorded data sequence. Nonlinear bit-shifts cause written transitions to be displaced with respect to their nominal positions. The displacement Δ_k (expressed in symbol intervals) is a deterministic function of one or more data symbols a (or, equivalently, of one or more transitions b). We denote the corresponding replay signal by $y(t)$ so as to discriminate it from $x(t)$. Accordingly

$$y(t) = \sum_{k=-\infty}^{\infty} b_k g(t - (k + \Delta_k)T). \tag{2B.1}$$

A positive value of Δ_k means that the transition b_k is deferred and a negative value means that it comes prematurely.

As described in Section 2.3.4, first-order bit-shifts cause a transition b_k to come ε_1 symbol intervals early in the presence of an immediately preceding transition (see also Fig. 2.20). Mathematically we may write [62]

$$\Delta_k = \varepsilon_1 \frac{b_k b_{k-1}}{4}. \tag{2B.2}$$

Accordingly, Δ_k can only differ from zero if a transition is currently being written ($b_k \neq 0$) and if there was also an immediately preceding transition ($b_{k-1} \neq 0$). Then $b_k b_{k-1}$ will be negative because successive transitions cannot have the same polarity. More specifically, $b_k b_{k-1} = -4$ so that $\Delta_k = -\varepsilon_1$.

From (2B.1), $y(t)$ consists of a sequence of shifted pulses $g(t - (k + \Delta_k)T)$ with amplitude b_k. The pulse amplitude is zero when there is no transition ($b_k = 0$), and the corresponding values of Δ_k are completely immaterial to $y(t)$. It is, therefore, unnecessary to normalize Δ_k to zero when $b_k = 0$, as in (2B.2). This freedom allows us to redefine Δ_k in a more convenient form. We need only be concerned with instants k at which $b_k \neq 0$, i.e. at which $a_k = -a_{k-1}$. Then $b_k = -2a_{k-1}$ so that

$$\Delta_k = -\varepsilon_1 a_{k-1} \frac{a_{k-1} - a_{k-2}}{2} = -\varepsilon_1 \frac{1 - a_{k-1}a_{k-2}}{2} = -\varepsilon_1 [a_{k-1} \oplus a_{k-2}] \tag{2B.3}$$

where '$\oplus$' denotes exclusive-or ($[a \oplus b] = 1$ if $a \neq b$ and 0 if $a = b$). This sequence Δ_k differs from the one of (2B.2) at some of the moments when $b_k = 0$, but it is identical at all other moments so that it yields the same signal $y(t)$.

Thus far we have only considered the effect of an immediately preceding transition. If the nonlinear bit-shift depends on the two most recent transitions then a second parameter ε_2 comes into play [62]:

$$\Delta_k = \varepsilon_1 \frac{b_k b_{k-1}}{4} + \varepsilon_2 \frac{b_k b_{k-2}}{4}. \tag{2B.4}$$

This may also be written $\Delta_k = \frac{b_k}{4}[(\varepsilon_1 - \varepsilon_2)b_{k-1} + \varepsilon_2(b_{k-1} + b_{k-2})]$. Along the same lines as for the first-order nonlinearity we can redefine Δ_k by considering only instants k for which $b_k \neq 0$. After some elementary manipulations it follows that

$$\Delta_k = -(\varepsilon_1 - \varepsilon_2)[a_{k-1} \oplus a_{k-2}] - \varepsilon_2 [a_{k-1} \oplus a_{k-3}]. \tag{2B.5}$$

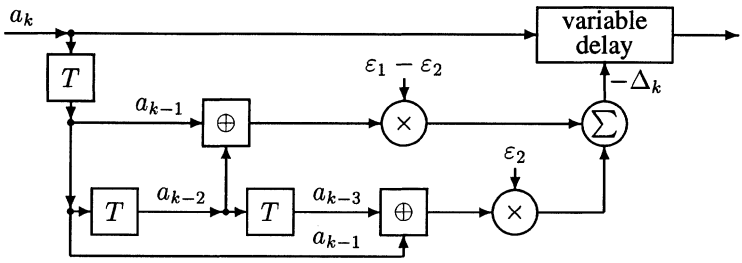


Figure 2B.1: Schematic model of second-order write-precompensation circuit.

This is a linear combination of two binary variables ($[a_{k-1} \oplus a_{k-2}]$ and $[a_{k-1} \oplus a_{k-3}]$). As such it has a simpler structure than the shift according to (2B.4), which is a linear combination of a binary and a ternary variable ($b_k b_{k-1}$ and $b_k b_{k-2}$, respectively). The representation of (2B.5) is essentially equivalent to the one of (2A.4), and can be extended in a similar fashion to higher-order nonlinear bit-shifts.

Write precompensation requires any shift Δ_k to be counteracted by an equally large shift in the opposite direction. The write-precompensation circuit implied by (2B.5) is depicted in Fig. 2B.1.

APPENDIX 2C

Additive models for transition jitter

Transition jitter is similar to nonlinear bit-shifts in that it also causes written transitions to shift with respect to their nominal positions. The distinguishing feature of transition jitter is that these shifts occur randomly and independently of the data sequence. Thus the replay signal $y(t)$ again has the form

$$y(t) = \sum_{k=-\infty}^{\infty} b_k g(t - (k + \Delta_k)T) \tag{2C.1}$$

where $b_k = a_k - a_{k-1} \in \{-2, 0, 2\}$ is the transition sequence, $g(t)$ is the transition response, and $a_k \in \{-1, 1\}$ is the recorded data sequence. The displacement Δ_k (normalized in symbol intervals T) is now a random variable that is statistically independent of a_k and b_k. If $|\Delta|$ is sufficiently small and $g(t)$ is sufficiently bandlimited then

$$g(t + \Delta T) \simeq g(t) + \Delta T g'(t) \text{ for all } t. \tag{2C.2}$$

This first-order approximation is reasonable in most conditions of practical interest, and permits us to write

$$y(t) \simeq \sum_{k=-\infty}^{\infty} b_k g(t - kT) + \sum_{k=-\infty}^{\infty} d_k \dot{g}(t - kT) \tag{2C.3}$$

where $d_k \triangleq -b_k\Delta_k$ is a random noise sequence and $\dot{g}(t) \triangleq Tg'(t)$ is the normalized derivative of $g(t)$. The first term of the right hand side may be recognized as $x(t)$ (compare (2.3)) and therefore $y(t) \simeq x(t) + z(t)$ where $z(t) \triangleq \sum_{k=-\infty}^{\infty} d_k\dot{g}(t - kT)$ is an additive disturbance component that accounts fully for transition jitter (Fig. 2C.1).

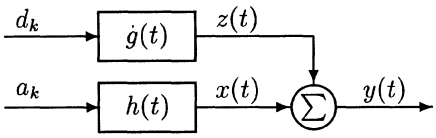


Figure 2C.1: Additive cyclostationary model for small transition jitter.

It can be observed that $z(t)$ is cyclostationary with the same period (T) as the data sequence. In this regard $z(t)$ is very similar to cross-talk. Statistical properties of $z(t)$ are largely determined by the underlying noise sequence d_k and as such by b_k and Δ_k. Since Δ_k represents a transition *shift*, the average or expected value of Δ is, by definition, zero. Furthermore, since Δ_k represents random transition jitter rather than nonlinear bit-shifts, Δ_k does not depend on the transition sequence b_k and we will, in fact, assume that Δ_k and b_k are statistically independent. It is then easy to see that the noise sequence d_k has the following properties.

1. It is uncorrelated with the transition sequence b_k since $E[b_k d_k] = -E[b_k^2]E[\Delta_k] = 0$.

2. It has zero mean because $E[d_k] = -E[b_k\Delta_k] = -E[b_k]E[\Delta_k] = 0$.

3. It has autocorrelation function $R_k^d \triangleq E[d_l d_{k+l}] = E[b_l b_{k+l}]E[\Delta_l\Delta_{k+l}] = R_k^b R_k^\Delta$ where R_k^b and R_k^Δ denote the autocorrelation functions of b_k and Δ_k, respectively.

4. If successive transition shifts are uncorrelated, i.e. if $R_k^\Delta = \sigma_\Delta^2\delta_k$ where σ_Δ denotes the RMS transition shift (normalized in symbol intervals) and δ_k denotes the Kronecker delta function, then d_k is also uncorrelated and has variance $\sigma_d^2 = \sigma_b^2\sigma_\Delta^2$ where $\sigma_b^2 = E[b_k^2]$ denotes the variance of b_k.

Exercise 2.3 Assume that the data sequence a_k is uncorrelated. Show that $\sigma_b^2 = 2$. What properties must the autocorrelation function $R_k^a \triangleq E[a_l a_{k+l}]$ of a_k meet in order for σ_b^2 to be smaller than 2?

It is not uncommon to assume that Δ_k is Gaussian. In that case d_k and $z(t)$ are also Gaussian.

Because of the cyclostationarity of $z(t)$, the mean and autocorrelation function of $z(t)$ are, in principle, periodic in time with period T. There is, however, one important exception. At normalized information densities of practical interest, the transition response $g(t)$ has little if any excess bandwidth, i.e. $|G|$ is negligible for normalized

frequencies $|\Omega| > 0.5$. In this event the mean and autocorrelation function of $z(t)$ become independent of time (see Problem 2.2), and $z(t)$ has power spectral density

$$P_z(\Omega) = \frac{1}{T} P_d(e^{j2\pi\Omega})(2\pi\Omega)^2 |G(\Omega)|^2 \tag{2C.4}$$

where $P_d(e^{j2\pi\Omega})$ denotes the power spectral density of d_k. If successive transition shifts are uncorrelated then (2C.4) simplifies to

$$P_z(\Omega) = \frac{\sigma_\Delta^2 \sigma_b^2}{T} (2\pi\Omega)^2 |G(\Omega)|^2. \tag{2C.5}$$

The fact that mean and autocorrelation function of $z(t)$ are independent of time does not imply that the distribution function of $z(t)$ is necessarily time-independent. Even so, in many instances $z(t)$ is approximately Gaussian for all t, and is then effectively a stationary noise signal as in Fig. 2C.2.

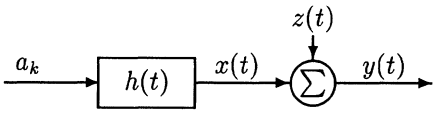


Figure 2C.2: Additive stationary model for small transition jitter, applicable when excess bandwidth is negligible.

The model of Fig. 2C.2 can be brought into an alternative and sometimes more desirable form by writing $h(t)$ as $(c * f)(t)$ where $c(t)$ is a hold function of width T and $f(t)$ is the impulse response of the channel, i.e. the derivative of the transition response $g(t)$ ($f(t) = g'(t)$). According to this decomposition, the data sequence a_k is first converted into a binary continuous-time write signal $s(t)$ by means of a linear pulse modulator with symbol response $c(t)$, and subsequently applied to a filter with impulse response $f(t)$ that mimics the action of the continuous-time channel (Fig. 2C.3). To account for transition jitter, we might inject additive noise $z(t)$ at the receiving end of the system as in Fig. 2C.2. In terms of the Fourier transform $F(\Omega)$ of $f(t)$, this noise has power spectral density

$$P_z(\Omega) = \sigma_\Delta^2 \sigma_b^2 T |F(\Omega)|^2. \tag{2C.6}$$

This expression suggests that we may equivalently obtain the noise component of $y(t)$ as in Fig. 2C.3, i.e. by adding white noise $n(t)$ of power spectral density $N_0 \triangleq \sigma_\Delta^2 \sigma_b^2 T$ to $s(t)$. An important feature of the model of Fig. 2C.3 is that data and noise are both filtered by the same filter. As a result, the spectral signal-to-noise ratio (see Chapter 3 for a definition) is independent of $f(t)$. (This is evidently also true for the model of Fig. 2C.2, but it is not as directly visible.) A direct consequence of this independence is that optimum receiver performances must be independent of $f(t)$ (provided, of course, that there are no other disturbances than those modeled in Fig. 2C.3).

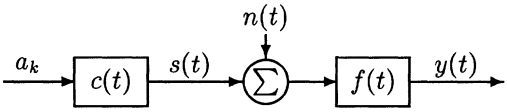


Figure 2C.3: Alternative additive stationary model for small transition jitter, applicable when excess bandwidth is negligible.

APPENDIX 2D

Nonlinear ISI induced by domain bloom

In this appendix we analyze the impact of domain bloom on the replay signal. The analysis follows the lines of Appendices 2B and 2C and uses the same notation.

In the absence of bloom, the data component $x(t)$ of the replay signal may be written

$$x(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT) \tag{2D.1}$$

where a_k is the recorded data sequence and $h(t)$ is the symbol response of the system. This may be recast in terms of the transition sequence $b_k = a_k - a_{k-1} \in \{-2, 0, 2\}$ according to

$$x(t) = \sum_{k=-\infty}^{\infty} b_k g(t - kT) \tag{2D.2}$$

where $g(t)$ is the transition response of the system. As illustrated in Fig. 2.33, this response is, in general, infinitely long because of the fact that the optical recording channel passes DC.

Domain bloom causes all transitions to be displaced with respect to their nominal positions. We denote the resulting replay signal by $y(t)$ so as to discriminate it from $x(t)$. Thus

$$y(t) = \sum_{k=-\infty}^{\infty} b_k g(t - (k + \Delta_k)T) \tag{2D.3}$$

where Δ_k denotes the displacement at moment k (normalized in symbol intervals T). A positive value of Δ_k means that the transition b_k is deferred and a negative value means that it comes prematurely. Displacements due to bloom are equally large for upgoing and downgoing transitions yet opposite in direction. Accordingly

$$\Delta_k = \gamma b_k \tag{2D.4}$$

where γ characterizes the degree of bloom. A positive value of γ means that the recorded pits (which correspond to $a_k = +1$) become systematically shorter than the lands ($a_k = -1$), and vice versa when γ is negative.

To analyze the nonlinear ISI that results from bloom we make the assumption that $g(t)$ is sufficiently bandlimited and that $|\Delta_k|$ is small enough to justify the first-order approximation

$$g(t + \Delta T) \simeq g(t) + \Delta T g'(t) \text{ for all } t. \tag{2D.5}$$

Based on this approximation we may write

$$y(t) \simeq \sum_{k=-\infty}^{\infty} b_k g(t - kT) - T \sum_{k=-\infty}^{\infty} b_k \Delta_k g'(t - kT). \tag{2D.6}$$

The first term of the right hand side may be recognized as $x(t)$. The second term is an additive disturbance component and may brought into a more desirable form by noting that $b_k \Delta_k = \gamma b_k^2 = 2\gamma(1 - a_k a_{k-1})$ and that $g'(t) = f(t)$ where $f(t)$ denotes the Gaussian impulse response of the optical read path (see Section 2.4.2 and in particular Figs. 2.33 and 2.34). Correspondingly $y(t) \simeq x(t) + z_0(t) + z_2(t)$ where

$$z_0(t) \triangleq -2\gamma T \sum_{k=-\infty}^{\infty} f(t - kT) \tag{2D.7}$$

and

$$z_2(t) \triangleq 2\gamma T \sum_{k=-\infty}^{\infty} a_k a_{k-1} f(t - kT). \tag{2D.8}$$

It follows that bloom manifests itself in the replay signal in the form of a periodically time-varying DC-offset $z_0(t)$ along with second-order nonlinear ISI that is induced by neighboring data symbols. The DC-offset becomes time-invariant in the usual event that the modulation transfer function $F(\Omega)$ is negligible at the Nyquist frequency $1/(2T)$ and beyond (see Problem 2.7). In that case

$$z_0(t) = z_0 = -2\gamma F(0). \tag{2D.9}$$

Thus z_0 is primarily determined the MTF at a single frequency $\Omega = 0$. Conversely, z_2 depends on basically all frequency components of F when data a_k is random. Because of the low-pass nature of the MTF, z_0 will be larger in magnitude than z_2, i.e. the principal effect of bloom on the replay signal is a DC offset.

PROBLEMS

2.1 Write currents of magnetic recording heads are normally produced by a write preamplifier with a single-ended supply voltage in the order of 5 Volts. As a rule, the maximum voltage across the head coil cannot exceed the supply voltage. Assume that the head has an inductance

of 100 nH and that a steady-state write current of ±40 mA is needed for the write bubble to be sufficiently large. What is the largest change of the write current per unit of time that is compatible with the assumed supply voltage? What rise time (10 percents to 90 percents of the distance between the two steady-state write currents) does this correspond to? How does this compare to a bit interval T at bit rates of 100 Mb/s? Hint: Make use of the fact that $V = LdI/dt$ where V is the current across the head windings, L is the head inductance and I is the write current.

2.2 Let $r(t)$ be a cyclostationary signal of the form $r(t) = \sum_{k=-\infty}^{\infty} d_k q(t - kT)$ where $q(t)$ is a given symbol response and d_k is a stationary noise sequence with zero mean and power spectral density $P_d(e^{j2\pi\Omega})$. Assume that $r(t)$ is sampled at instants $(k + \phi)T$ where ϕ denotes the sampling phase, and denote the sampler output by z_k.

1. Show that z_k has zero mean irrespective of ϕ.
2. Assume that $q(t)$ has negligible excess bandwidth, i.e. that the Fourier integral $Q(\Omega)$ of $q(t)$ vanishes for $|\Omega| > 0.5$. Express the power spectral density $P_d(e^{j2\pi\Omega})$ of z_k in terms of $Q(\Omega)$ and $P_d(e^{j2\pi\Omega})$. Show that $P_z(e^{j2\pi\Omega})$ does not depend on ϕ.

2.3 Show that the symbol response $h(t)$ of Fig. 2.35 may be expressed as

$$h(t) = Q(2\sqrt{2}\frac{t - \frac{T}{2}}{t_0}) - Q(2\sqrt{2}\frac{t + \frac{T}{2}}{t_0}) \tag{2P.1}$$

where $Q(x)$ is the Q-function as defined in (1.81).

2.4 Prove (2C.4) and (2C.5).

2.5 In this problem we investigate the impact of a minimum-run-length constraint on the severeness of domain bloom. The introduction of such a constraint inevitably commands a premium in terms of redundancy that must be added to recorded data stream. For a minimum run-length of 3 symbol intervals (as in Example 2.2) this redundancy is in the order of 100 percents, i.e. for every user bit a_k there are two encoded bits b_n and b_{n+1} (see Section 4.8). If the net recording density is to be the same as for uncoded storage (i.e. for the situation that a_k is recorded directly), then these two bits must together have the same physical length L on disc as a single bit a_k would have.

1. Express the physical minimum run-lengths for uncoded and coded storage in terms of L.
2. Assume that bloom is as severe as in Figs. 2.36.b and c. How much longer or shorter (expressed in units L) are pits and lands with respect to their nominal length? What is the length (expressed in units L) of the shortest pits and lands for both uncoded and coded storage? Could uncoded storage work?
3. Repeat the previous question for bloom that is half as severe as that in Figs. 2.36.b and c.

2.6 Consider the finite sum

$$S \triangleq \sum_{j=2}^{N} a_{k-j} d_j \tag{2P.2}$$

where a_k is a binary data sequence and the coefficients d_j are differences of prescribed coefficients c_j according to $d_j = c_j - c_{j-1}$. Show that

$$S = a_{k-1}(c_N - c_1) + \sum_{j=1}^{N-1} b_{k-i}(c_i - c_N) \tag{2P.3}$$

where $b_k = a_k - a_{k-1}$ is the transition sequence corresponding to a_k. Use this result along with the fact that $b_k a_{k-1} = -b_k^2/2$ to prove (2A.6) and (2A.7).

2.7 Use Poisson's summation formula to prove (2D.9). What happens to $z_0(t)$ if $F(\Omega)$ is not yet negligble at the bit frequency ($|\Omega| = 1$)?

REFERENCES

Digital subscriber lines:

[1] J.W.M. Bergmans, 'On the Design of Smearing Filters for ISDN Transmission Systems', *Philips J. Res.*, Vol. 42, No. 2, pp. 172-208, 1987.

[2] J.A.C. Bingham, 'Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come', *IEEE Communications Magazine*, pp. 5-14, May 1990.

[3] 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 J. Selected Areas Commun.*, Vol. SAC-13, No. 9, pp. 1656-1669, Dec. 1995.

[4] 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. Commun.*, Vol. 44, No. 6, pp. 675-685, June 1996.

[5] A.N. Coles, D.G. Cunningham, and S.G. Methley, '100 Mb/s Data Transmission on UTP and STP Cabling for Demand Priority Networks', *IEEE J. Selected Areas Commun.*, Vol. SAC-13, No. 9, pp. 1684-1691, Dec. 1995.

[6] P.J. van Gerwen, N.A.M. Verhoeckx and T.A.C.M. Claasen, 'Design Considerations for a 144 kbit/s Digital Transmission Unit for the Local Telephone Network', *IEEE J. Selected Areas Commun.*, Vol. SAC-2, No. 2, pp. 314-323, March 1984.

[7] K.J. Kerpez and K. Sistanizadeh, 'High Bit Rate Asymmetric Digital Communications Over Telephone Loop', *IEEE Trans. Commun.*, Vol. COM-43, No. 6, pp. 2038-2049, June 1995.

[8] J.W. Lechleider, 'High Bit Rate Digital Subscriber Lines: A Review of HDSL Progress', *IEEE J. Selected Areas in Commun.*, Vol. 9, No. 6, pp. 769-784, Aug. 1991.

[9] P.J. Kyees, R.C. McConnell, and K. Sistanizadeh, 'ADSL: A New Twisted-Pair Access to the Information Highway', *IEEE Communications Magazine*, pp. 52-59, Apr. 1995.

[10] D.G. Messerschmitt, 'Design Issues in the ISDN U-Interface Transceiver', *IEEE J. Selected Areas Commun.*, Vol. SAC-4, No. 8, Nov. 1986.

[11] J.W. Modestino, D. Sargrad and R.E. Bollen, 'Use of Coding to Combat Impulse Noise on Digital Subscriber Loops', *IEEE Trans. Commun.*, Vol. COM-36, No. 5, pp. 529-537, May 1988.

[12] N.A.M. Verhoeckx, H.C. van den Elzen, F.A.M. Snijders and P.J. Van Gerwen, 'Digital Echo Cancellation for Baseband Data Transmission', *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-27, No. 6, pp. 768-781, Dec. 1979.

[13] W. Walkoe and T.J.J. Starr, 'High Bit Rate Digital Subscriber Line: A Copper Bridge to the Network of the Future', *IEEE J. Selected Areas in Commun.*, Vol. SAC-9, No. 6, pp. 765-769, Aug. 1991.

[14] J.-J. Werner, 'The HDSL Environment', *IEEE J. Selected Areas in Commun.*, Vol. SAC-9, No. 6, pp. 785-800, Aug. 1991.

[15] K.J. Wouda, S.J.M. Tol, P. Zuidweg and O. Reingruber, 'Towards a Single Chip ISDN Transmission Unit', *Proc. ISSLS'86*, pp. 250-255, Toky, Japan, 1986.

Digital magnetic recording:

[16] W.L. Abbott, J.M. Cioffi, and H.M. Thapar, 'Performance of Digital Magnetic Recording with Equalization and Offtrack Interference'. *IEEE Trans. Magn.*, Vol. MAG-27, No. 1, pp. 705-716, Jan. 1991.

[17] T.C. Anthony, S.L. Naberhuis, J.A. Brug, M.J.K. Bhattacharyya, L.T. Tran, V.W. Hesterman, and G.G. Lopatin, 'Dual-Stripe Magnetoresistive Heads for High Density Recording', *IEEE Trans. Magn.*, Vol. MAG-30, No. 2, pp. 303-308, March 1994.

[18] T.C. Arnoldussen and L.L. Nunnelley, *Noise in Digital Magnetic Recording*. Singapore: World Scientific Publishing Co. Pte. Ltd., 1992.

[19] S.J. Begun, *Magnetic Recording*. New York: Rinehart books, 1949.

[20] N.R. Belk, P.K. George and G.S. Mowry, 'Noise in High Performance Thin-Film Longitudinal Magnetic Recording Media', *IEEE Trans. Magn.*, Vol. MAG-21, No. 5, pp. 1350-1355, Sept. 1985.

[21] J.W.M. Bergmans, 'Discrete-Time Models for Digital Magnetic Recording', *Philips J. Res.*, Vol. 41, No. 6, pp. 531-558, 1986.

[22] H.N. Bertram, 'Fundamentals of the Magnetic Recording Process', *Proc. IEEE*, Vol. 74, No. 11, pp. 1494-1512, Nov. 1986.

[23] P.G. Blanken and J.J.L.M. van Vlerken, 'Modeling of Electromagnetic Systems', *IEEE Trans. Magn.*, Vol. MAG-27, No. 6, pp. 4509-4515, Nov. 1991.

[24] A.D. Booth, 'Magnetic Digital Storage System', *Electronic Engineering*, Vol. 21, pp. 234-38, July 1949.

[25] S.M.C. Borgers, W.A.L. Heynemans, E. de Niet and P.H.N. de With, 'An Experimental Digital VCR With 40 mm. Drum, Single Actuator and DCT-Based Bit-Rate Reduction', *IEEE Trans. Consumer Electr.*, Vol. CE-34, No. 3, pp. 597-605, Aug. 1988.

[26] M. Camras, *Magnetic Tape Recording*. New York: Van Nostrand Reinhold, 1985.

[27] W.L. Carlson and G.W. Carpenter, 'Radio Telegraph System', U.S. patent 1 640 881 (1927). Reproduced in [26].

[28] X. Che, M.K. Bhattacharyya and H.N. Bertram, 'Studies of Nonlinear Bit Shift and Partial Erasure Using Pseudo-Random Sequence', *IEEE Trans. Magn.*, Vol. MAG-29, No. 6, pp. 3972-3974, Nov. 1993.

[29] K. Chopra, L. Nix and K.D. Taberski, 'The Effects of MR Induced Non-Linearities on Channel Performance', *IEEE Trans. Magn.*, Vol. MAG-30, No. 6, pp. 4200-4202, Nov. 1994.

[30] J.M. Cioffi, W.L. Abbot, H.K. Thapar, C.M. Melas and K.D. Fisher, 'Adaptive Equalization in Magnetic-Disk Storage Channels', *IEEE Communications Magazine*, pp. 14-29, Feb. 1990.

[31] Y. Eto, S. Mita and N. Doi, *Digital Video Recording Techniques* (in Japanese). Tokyo: Nikkan Industrial Newspapers Co., 1990 (ISBN 4-526-02768-5 C3055).

[32] Y. Eto, 'Signal Processing for Future Home-Use Digital VTR's', *IEEE J. Selected Areas in Commun.*, Vol. SAC-10, No. 1, pp. 73-79, Jan. 1992.

[33] L.-S. Fan, H.H. Ottesen, T.C. Reiley, and R.W. Wood, 'Magnetic Recording Head Positioning at Very High Track Densities Using a Microactuator-Based, Two-Stage Servo System', *IEEE Trans. Industrial Electr.*, Vol. 42, No. 3, pp. 222-233, June 1995.

[34] R.E. Fayling, T.J. Szczech and E.F. Wollack, 'A Model for Overwrite Modulation in Longitudinal Recording', *IEEE Trans. Magn.*, Vol. MAG-20, No. 5, pp. 718-720, Sept. 1984.

[35] J.M. Harker, D.W. Bede, R.E. Pattison, G.R. Santana and L.G. Taft, 'A Quarter Century of Disk File Innovation', *IBM J. Res. Develop.*, Vol. 25, No. 5, pp. 677-689, Sep. 1981.

[36] J.P. Harris, W.B. Phillips, J.F. Wells, and W.D. Winger, 'Innovations in the Design of Magnetic Tape Subsystems', *IBM J. Res. Develop.*, Vol. 25, No. 5, pp. 691-699, Sept. 1981.

[37] Y. Hashimoto, 'Experimental HDTV Digital VTR with a Bit Rate of 1 Gbps', *IEEE Trans. Magn.*, Vol. MAG-23, pp. 3167-3172, Sep. 1987.

[38] A.S. Hoagland and J.E.Monson, *Digital Magnetic Recording* (2nd ed.). New York: Wiley, 1991.

[39] A. Hoogendoorn, 'Digital Compact Cassette', *Proc. IEEE*, Vol. 82, No. 10, pp. 1479-1489, Oct. 1994.

[40] S.-I. Iwasaki, 'Perpendicular Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-16, No. 1, pp. 71-76, Jan. 1980.

[41] S.-I. Iwasaki, 'Perpendicular Magnetic Recording - Evolution and Future', *IEEE Trans. Magn.*, Vol. MAG-20, No. 5, pp. 657-662, Sept. 1984.

[42] S.-I. Iwasaki and J. Hokkyo (Eds.), *Perpendicular Magnetic Recording*. Tokyo: Ohmsha, Ltd., 1991.

[43] F. Jorgensen, *The Complete Handbook of Magnetic Recording* (3rd ed.). Blue Ridge Summit: TAB Books, 1988.

[44] J.A.H. Kahlman and K.A. Schouhamer Immink, 'Channel Code with Embedded Pilot Tracking Tones for DVCR', *IEEE Trans. Consumer Electr.*, Vol. CE-41, No. 1, pp. 180-185, Feb. 1995.

[45] O. Karlqvist, 'Calculation of the Magnetic Field in the Ferromagnetic Layer of a Magnetic Drum', *Trans. Roy. Inst. Technol. Stockholm*, No. 86, pp. 1-27, 1954.

[46] K.B. Klaassen, 'Magnetic Recording Channel Front-Ends', *IEEE Trans. Magn.*, Vol. MAG-27, pp. 4503-4508, 1991.

[47] K.B. Klaassen and J.C.L. van Peppen, 'Read/Write Amplifier Design Considerations for MR Heads', *IEEE Trans. Magn.*, Vol. MAG-31, No. 2, pp. 1056-1061, March 1995.

[48] K.B. Klaassen and R.G. Hirko, 'Writing and Reading at High Data Rates', preprint, 1995.

[49] M.H. Kryder, 'Data Storage in 2000 - Trends in Data Storage Technologies', *IEEE Trans. Magn.*, Vol. 23, No. 6, pp. 4358-4363, Nov. 1989.

[50] S.B. Luitjens, 'Magnetic Recording Trends: Media and Future (Video) Recording Systems', *IEEE Trans. Magn.*, Vol. MAG-26, No. 1, pp. 6-11, 1990.

[51] J.C. Mallinson, 'Tutorial Review of Magnetic Recording', *Proc. IEEE*, Vol. 64, No. 2, pp. 196-208, Feb. 1976.

[52] J.C. Mallinson, *The Foundations of Magnetic Recording* (2nd ed.). Orlando, FL: Academic Press, 1993.

[53] C.D. Mee and E.D. Daniel (Eds.), *Magnetic Recording*, Volumes I-III. New York: McGraw-Hill, 1987.

[54] S. Mita, M. Izumita, N. Doi and M. Umemoto, 'Digital Video Recording Techniques Using 1/2 Inch Metal Particle Tape', *IEEE Trans. Consumer Electron.*, Vol. CE-31, No. 3, pp. 386-397, Aug. 1985.

[55] J. Moon and L.R. Carley, *Sequence Detection for High-Density Storage Channels.* Dordrecht: Kluwer Academic Publishers, 1992.

[56] J. Moon and J.-G. Zhu, 'Nonlinear Effects of Transition Broadening', *IEEE Trans. Magn.*, Vol. MAG-27, No. 6, pp. 4831-4833, Nov. 1991.

[57] S. Nakagawa, K. Yokoyama and H. Katayama, 'A Study on Detection Methods of NRZ Recording', *IEEE Trans. Magn.*, Vol. MAG-16, No. 1, pp. 104-110, Jan. 1980.

[58] Y. Nakamura and I. Tagawa, 'Possibilities of Perpendicular Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-24, No. 6, pp. 2329-2334, 1988.

[59] T. Nakahashi, S. Mita and M. Izumita, 'Characteristics Evaluation of MFM Group Channel Codes', *1988 National Convention Record of the Institute of TV Engineers of Japan*, pp. 175-176 (in Japanese).

[60] H. Nakajima and M. Kosaka, 'The DAT Conference: Its Activities and Results', *IEEE Trans. Consumer Electron.*, Vol. CE-32, pp. 404-415, Aug. 1986.

[61] P. Newby and R. Wood, 'The Effects of Nonlinear Distortion on Class IV Partial Response', *IEEE Trans. Magn.*, Vol. MAG-22, No. 5, pp. 1203-1205, Sept. 1986.

[62] D. Palmer, P. Ziperovich, R. Wood and T.D. Howell, 'Identification of Nonlinear Write Effects Using Pseudorandom Sequences', *IEEE Trans. Magn.*, Vol. MAG-23, No. 5, pp. 2377-2379, Sept. 1987.

[63] D. Palmer, J. Coker, M. Meyer and P. Ziperovich, 'Overwrite in Thin Media Measured by the Method of Pseudorandom Sequences', *IEEE Trans. Magn.*, Vol. 24, No. 6, pp. 3096-3098, Nov. 1988.

[64] D. Palmer, J. Hong, D. Stanek and R. Wood, 'Characterization of the Read/Write Process for Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-31, No. 2, pp. 1071-1076, March 1995.

[65] F. Pfleumer, D.R. Patent No. 500 900 (1930).

[66] J.T. Potter and P.C. Michel, 'High Density Digital Recording System', *IRE Trans. Electronic Computers*, pp. 60-72, Dec. 1952.

[67] R.I. Potter, 'Digital Magnetic Recording Theory', *IEEE Trans. Magn.*, Vol. MAG-10, pp. 502-508, 1974.

[68] V. Poulsen, 'The Telegraphone: A Magnetic Speech Recorder', *The Electrician*, Vol. 46, pp. 208-210, 1900. Reproduced in [26]. Also: Ann. Phys., Vol. 3, p. 754, 1900; and U.S. Patent 661 619, July 8, 1899.

[69] N.P. Sands and J.M. Cioffi, 'Nonlinear Channel Models for Digital Magnetic Recording', *IEEE Trans. Magn.*, Vol. 29, No. 6, pp. 3996-3998, Nov. 1993.

[70] V.H. Sugaya, 'The Past Quarter-Century and the Next Decade of Videotape Recording', *SMPTE Journal*, Vol. 101, No. 1, pp. 10-13, Jan. 1992.

[71] K.A. Schouhamer Immink, *Coding Techniques for Digital Recorders*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[72] B. Slutsky and H.N. Bertram, 'Transition Noise Analysis of Thin Film Magnetic Recording Media', *IEEE Trans. Magn.*, Vol. 30, No. 5, pp. 2808-2817, Sept. 1994.

[73] F.B. Shelledy and J.L. Nix, 'Magnetoresistive Heads for Magnetic Tape and Disk Recording', *IEEE Trans. Magn.*, Vol. 28, No. 5, pp. 2283-2288, Sept. 1992.

[74] O. Smith, 'Some Possible Forms of Phonograph', *Electrical World*, pp. 116-117, 1888. Reproduced in [26].

[75] L. Thurlings, 'Statistical Analysis of Signal and Noise in Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-16, No. 3, pp. 507-513, May 1980.

[76] C. Tsang, M. Chen and T. Yogi, 'Gigabit-Density Magnetic Recording', *Proc. IEEE*, Vol. 81, No. 9, pp. 1344-1359, Sept. 1993.

[77] R.L. Wallace, Jr., 'The Reproduction of Magnetically Recorded Signals', *Bell Syst. Tech. J.*, Vol. 30, pp. 1145-1173, Oct. 1951.

[78] J. Watkinson, *R-DAT*. Oxford, U.K.: Butterworth-Heinemann Ltd., 1991.

[79] C.F. West and J.E. De Turk,' A Digital Computer for Scientific Applications', *Proc. IRE*, Vol. 36, pp. 1452-1460, Dec. 1948.

[80] W.K. Westmijze, 'The Principle of the Magnetic Recording and Reproduction of Sound', *Philips Technical Review*, Vol. 15, No. 3, pp. 84-96, Sept. 1953.

[81] R.M. White, *Introduction to Magnetic Recording*. New York: IEEE Press, 1985.

[82] C.F. Wilson, 'Magnetic Recording - 1888-1952', *IRE Trans. Audio*, pp. 53-81, May-June 1956.

[83] C. Wood and P. Hodges, 'DASD Trends: Cost, Performance, and Form Factor'. *Proc. IEEE*, Vol. 81, No. 4, pp. 573-584, April 1993.

[84] R.W. Wood and R. W. Donaldson, 'The Helical-Scan Magnetic Tape Recorder as a Digital Communication Channel', *IEEE Trans. Magn.*, Vol. MAG-15, No. 2, pp. 935-943, March 1979.

[85] R.W. Wood, 'Magnetic Recording Systems', *Proc. IEEE*, Vol. 74, No. 11, pp. 1557-1569, Nov. 1986.

[86] R.W. Wood, 'Magnetic and Optical Storage Systems: Opportunities for Communications Technology', *Proc. ICC'89*, pp. 1605-1612, Boston, USA, June 11-14, 1989.

[87] R.W. Wood, 'Magnetic Megabits', *IEEE Spectrum*, pp. 32-38, May 1990.

[88] R.W. Wood, M. Williams, and J. Hong, 'Considerations for High Data Rate Recording With Thin-Film Heads', *IEEE Trans. Magn.*, Vol. MAG-26, No. 6, pp. 2954-2959, Nov. 1990.

[89] H.W. Wong-Lam, 'A Simple Transition Response Model for Magnetoresistive Heads', submitted to *IEEE Trans. Magn.*, 1995.

[90] H.W. Wong-Lam, 'Remarks on Write-Field Waveform and Rise-Time in Digital Magnetic Recording', submitted to *IEEE Trans. Magn.*, 1995.

[91] W. Zeng and J. Moon, 'Systems Modeling of Nonlinear Effects in High Density Digital Magnetic Recording', *Proc. GLOBECOM'94*, pp. 1139-1143, San Francisco, CA, Nov. 28-Dec. 2, 1994.

Digital optical recording:

[92] K. Aratani, A. Fukumoto, M. Ohta, M. Kaneko and K. Watanabe, 'Magnetically-Induced Superresolution in Novel Magneto-Optical Disk', *SPIE Proceedings on Optical Data Storage*, Vol. 1499, pp. 209-215, 1991.

[93] P. Asthana, 'A Long Road to Overnight Success', *IEEE Spectrum*, pp. 60-66, Oct. 1994.

[94] J.W.M. Bergmans, S. Mita and M. Izumita, 'Receiver for Data Transmission System with Nonlinearities', U.S. patent no. 5,131,011, Jul. 14, 1992.

[95] G. Bouwhuis, J. Braat, A. Huijser, J. Pasman, G. van Rosmalen and K. Schouhamer Immink, *Principles of Optical Disc Systems*. Bristol, U.K: Adam Hilger Ltd., 1985.

[96] K. Compaan and P. Kramer, 'The Philips 'VLP' System', *Philips Tech. Rev.*, Vol. 33, pp. 178-80, 1973.

[97] A.G. Dewey, 'Optimizing the Noise Performance of a Magneto-Optic Read Channel', *SPIE Top. Meet. Opt. Data Storage*, Vol. 1078, pp. 279-286, 1989.

[98] B. Finkelstein and W. Williams, 'Noise Sources in Magneto-Optical Recording', *Applied Optics*, Vol. 27, No. 4, pp. 703-709, 1988.

[99] J.W. Goodman, *Introduction to Fourier Optics*. New York: McGraw-Hill, 1968.

[100] 'Digital Videodisc Development Spins Into New Territory', *Journal of the Electronics Industry*, pp. 18-23, June 1995.

[101] J.J.L. Horikx, K. Asato, N. Eguchi and J.L. Bakx, 'System Parameters and Detection Margins of the MultiMedia Compact Disc System', Proc. ODS-95.

[102] J. Isailovic, *Videodisc and Optical Memory Systems*. New York: Prentice-Hall, 1985.

[103] K. Kayanuma et al., 'High Track Density Magneto-Optical Recording Using a Cross Talk Canceler', *SPIE Proceedings on Optical Data Storage*, Vol. 1316, pp. 35-39, 1990.

[104] M.H. Kryder and M.D. Schultz, 'Direct Overwrite in Magneto-Optical Recording Technology', *Japan J. Appl. Phys.*, Vol. 28, suppl. 3, pp. 3-9, 1989.

[105] T. Larsen, F. Woodard, and S. Pace, 'A New Tape Product for Optical Data Storage', *Jap. J. Appl. Phys.*, Vol. 32, Part 1, No. 11B, pp. 5461-5462, Nov. 1993.

[106] M.D. Levenson, R.T. Lynch, Jr., and S.M. Tan, 'Edge Detection for Magneto-Optical Storage', *Appl. Opt.*, Vol. 30, No. 2, pp. 232-252, Jan. 1991.

[107] C.-J. Lin, 'Critical Assessment of the Physics Underlying Direct Overwrite in Magneto-Optic Recording', *J. Appl. Phys.*, Vol. 67, pp. 4409-4414, 1990.

[108] R.T. Lynch, Jr., 'Channels and Codes for Magnetooptical Recording', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 57-72, Jan. 1992.

[109] M. Mansuripur, G.A.N. Connell, and J.W. Goodman, 'Signal-to-Noise in Magneto-Optic Storage', *Proc. Soc. Photo-Opt. Instrum. Eng.*, Vol. 329, pp. 215-222, 1982.

[110] A.B. Marchant, *Optical Recording: A Technical Overview*. Reading, MA: Addison-Wesley, 1990.

[111] T. Milster and C. Curtis, 'Transfer Function Characteristics of Super Resolving Systems', *SPIE Proceedings on Optical Data Storage*, Vol. 1663, pp. 123-126, 1992.

[112] T. Narahara, Y. Kamatani, T. Nakao, S. Kumai and T. Ozue, 'A New Tracking Method for Helical Scanning Optical Tape Recorder', *Jpn. J. Appl. Phys.*, Vol. 32, Part 1, No. 11B, pp. 5421-5427, Nov. 1993.

[113] J.B.H. Peek, 'Communications Aspects of the Compact Disc Digital Audio System', *IEEE Communications Magazine*, Vol. 23, No. 2, pp. 7-15, Feb. 1985.

[114] G.E. van Rosmalen, J.A.H. Kahlman, P.L.M. Put and C.M.J. van Uijen, 'A Compact Optical Tape Recording System', *SPIE Vol. 2338 Optical Data Storage*, pp. 9-14, 1994.

[115] D. Rugar, 'Magnetooptic Direct Overwrite Using a Resonant Bias Coil', *IEEE Trans. Magnet.*, Vol. MAG-24, No. 1, pp. 666-669, Jan. 1988.

[116] K.A. Schouhamer Immink, 'Coding Methods for High-Density Optical Recording', *Philips J. Res.*, Vol. 41, No. 4, pp. 410-430, 1986.

[117] G.T. Sincerbox and J.M. Zavislan (Eds), *Selected Papers on Optical Storage*. Bellingham, Washington USA: SPIE Optical Engineering Press, 1992.

[118] E.F. Stikvoort and J.A.C. v. Rens, 'An All-Digital Bit Detector for Compact Disc Players', *IEEE J. Selected Areas in Commun.*, Vol. SAC-10, No. 1, pp. 191-200, Jan. 1992.

[119] A. Strandjord, S. Webb, D. Perettie, and R. Cipriano, 'Flexible Storage Medium for Write-Once Optical Tape', *Proc. SPIE*, Vol. 1323, p. 362, 1992.

[120] F. Tanaka, S. Tanaka, and N. Imamura, 'Magneto-Optic Recording Characteristics of TbFeCo Media by Magnetic Field Modulation Method', *Japan J. Appl. Phys.*, Vol. 28, pp. 231-235, 1987.

[121] G.E. Thomas, 'Future Developments in Optical Recording', *Philips Tech. Rev.*, Vol. 44, No. 2, pp. 51-57, April 1988.

[122] H. Tokumaru, K. Arai, S. Yoshimura, and H. Oshima, 'Recording Experiment with Rotating Optical Head for Magnetooptical Tape Recording System', *Jap. J. Appl. Phys.*, Vol. 32, Part 1, No. 11B, pp. 5428-5432, Nov. 1993.

[123] D. Treves and D.S. Bloomberg, 'Signal, Noise, and Codes in Optical Memories', *Opt. Eng.*, Vol. 25, No. 7, pp. 881-891, July 1986.

[124] E.W. Williams, *The CD-ROM and Optical Disc Recording Systems*. Oxford, U.K.: Oxford University Press, 1994.

[125] Y. Yamanaka, Y. Hirose, H. Fujii and K. Kubota, 'High Density Recording by Superresolution in an Optical Disk Memory System', *Applied Optics*, Vol. 29, No. 20, pp. 3046-3051, July 10, 1990.

[126] T. Yoshida, 'The Rewritable MiniDisc System', *Proc. IEEE*, Vol. 82, No. 10, pp. 1492-1500, Oct. 1994.

3

CHARACTERISTICS OF DIGITAL BASEBAND TRANSMISSION

3.1 INTRODUCTION

Digital transmission systems usually transport information in a band of restricted width that is centered around some carrier frequency. The carrier frequency is typically large with respect to the bandwidth of the information-bearing signal, but in some cases (notably in certain voiceband data modems) it is of the same order of magnitude. The distinguishing feature of *baseband* transmission is that the carrier frequency is zero, i.e. that data is transmitted in a band around DC. As exemplified in Chapter 2, this approach is well-suited for channels that only pass frequencies near DC, such as wire pairs and digital recording channels.

Forthcoming chapters concentrate on distinct building blocks of digital baseband transmission systems, such as the modulation encoder and decoder (Chapter 4), the equalizer and detector (Chapters 5 to 7), the adaptation subsystem (Chapter 8) and the timing-recovery subsystem (Chapters 9 to 11). To set the stage for these chapters, the present chapter explores the topology of baseband transmission systems, introduces basic notions and terminology, and discusses fundamental performance limits.

Section 3.2 describes a general system model, introduces the various building blocks, and defines various measures of signal-to-noise ratio. Sections 3.3 to 3.5 are concerned with the *capacity* of the channel, which is, in essence, the largest data rate that the channel can support. The modulation and demodulation steps which surround the channel need not be optimum and therefore the capacity of the *discrete-time channel* that is formed by the modulator, the actual continuous-time channel and demodulator in cascade cannot exceed the capacity of the continuous-time channel alone (Section 3.6). Capacity losses are typically inevitable in the modulator, but can be avoided in the demodulator via a so-called *matched filter*, whose sampled output is a *sufficient statistic* for optimum detection of the modulator input (Sections 3.7 and 3.8). Closely related to this filter is the *matched-filter bound*, which sets an upper limit to attainable performance for any data receiver (Section 3.9). The matched filter emerges naturally as a part of a hypothetical receiver for maximum-likelihood detection of the transmitted data sequence. In Chapter 7 this receiver will serve as a starting point for our development of

the Viterbi detector. Section 3.10 derives discrete-time channel models that are equivalent to the original continuous-time channel in terms of attainable performance, and in passing derives the *whitened matched filter*, which is the optimum prefilter for the Viterbi detector. Based on one of these channel models, Section 3.11 analyzes the performance of the maximum-likelihood sequence detector.

3.2 SYSTEM MODEL AND NOMENCLATURE

Fig. 3.1 shows the topology of a baseband transmission system. Main building blocks are a *data source*, a *transmitter*, a *channel*, a *receiver* and a *data sink*. We now make a brief guided tour.

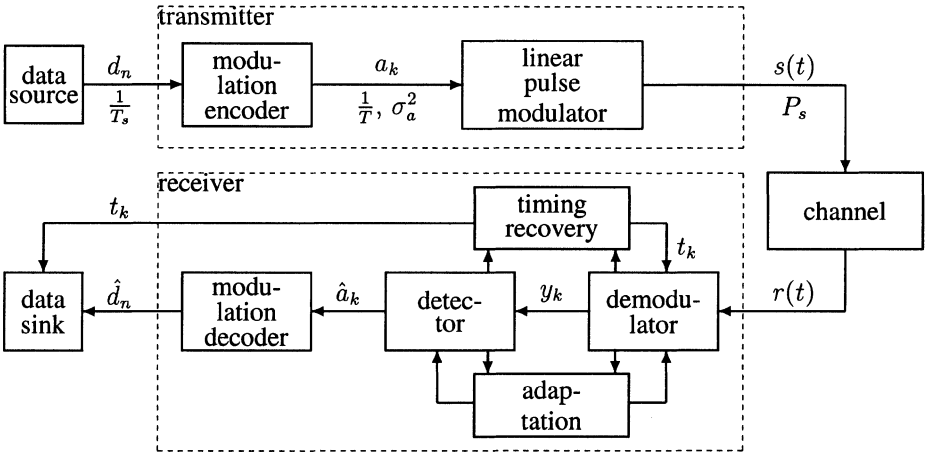


Figure 3.1: Baseline system topology.

3.2.1 Data source

The data source emits a bit sequence d_n of data rate $1/T_s$. This sequence represents the information that is to be conveyed. It is the combined result of all operations that precede the transmitter, including possible source coding, secrecy coding and error-protection coding steps. These preceding steps are immaterial to what follows and are beyond the scope of this text. What matters is that d_n is usually random but that it may also contain long strings of ones, zeros, or other deterministic data patterns. Furthermore, d_n is sometimes subdivided into bursts or slots of restricted length that are to be transmitted separately (*time-division multiplex*). In all cases the information should get across reliably.

3.2.2 Transmitter

The transmitter is meant to convert d_n into a continuous-time signal $s(t)$ that is matched to the channel. This obviously requires prior knowledge about the channel characteristics, and we shall see in Chapter 4 how this knowledge materializes in a number of practical encoding schemes. The power P_s of $s(t)$ is usually constrained to some value that is prescribed by government agencies, restricted by physics, or dictated otherwise. At least conceptually, two principal blocks can be distinguished.

1. A *modulation encoder* which transforms d_n into a data sequence a_k that is still time-discrete but already matched to a certain extent the channel. The transformation rule is referred to as *modulation code* and also as *recording code* and *line code* in recording and cable transmission jargon, respectively. The data rate $1/T$ of a_k is called the *signaling rate* or *baud rate*. It can be equal to the source rate $1/T_s$ but can also be higher or lower. In the latter case a_k must have more than two amplitude levels or else information of d_n will get lost. The ratio $R \triangleq T/T_s$ is the *information rate* of the code. For binary codes R cannot exceed unity.

2. A *linear pulse modulator* that serves to convert a_k into continuous-time format.[1] This block may include a transmit filter as in Sections 5.7, 6.2.3 and 6.3, and is characterized by a *symbol response* $c(t)$. In many instances $c(t)$ is just a hold function of width T.

For brevity we will label these two building blocks *encoder* and *modulator* whenever no confusion can arise. The subdivision of transmitter into encoder and modulator, while always possible, is not necessarily unique. One reason is that one or more linear discrete-time operations may take place just before the conversion into continuous-time format. These operations may be attributed to either encoder or modulator. We will adopt the latter convention by default. An illuminating exposition on the borderline between coding and modulation may be found in [19]. Since $s(t)$ has the same information content as a_k we shall say that $s(t)$ *carries* a_k or, equivalently, that it carries the source sequence d_n.

Example 3.1 Bi-phase (also called Diphase or WAL-1) is a binary DC-free code that can be described in two equivalent ways. According to the model of Fig. 3.2.a, the sequence d_n is applied to an interpolating filter with an oversampling factor of 2 and impulse response $w_k = 1$ for $k = 0$, -1 for $k = 1$ and 0 else. The filter output a_k is a bit sequence of rate $2/T$ that equals d_n at even instants ($k = 2n$) and $-d_n$ at odd instants ($k = 2n + 1$). A binary continuous-time signal $s(t)$ is obtained via a hold function of width $T = T_s/2$.

In the alternative model of Fig. 3.2.b, the encoder degenerates (i.e. $a_n = d_n$ and $1/T = 1/T_s$) and the modulator has the bipolar symbol response $c(t) = 1$ for $t \in (-\frac{T}{2}, 0)$, -1 for $t \in (0, \frac{T}{2})$ and 0 else. In the sequel we will adopt this second viewpoint of Bi-phase. The implicit assumption is that the receiver will normally detect d_n directly, rather than first detecting the oversampled sequence a_k of Fig. 3.2.a prior to reconstructing d_n.

[1] It should be noted that this is *not* a modulator in the radio-frequency sense of the word.

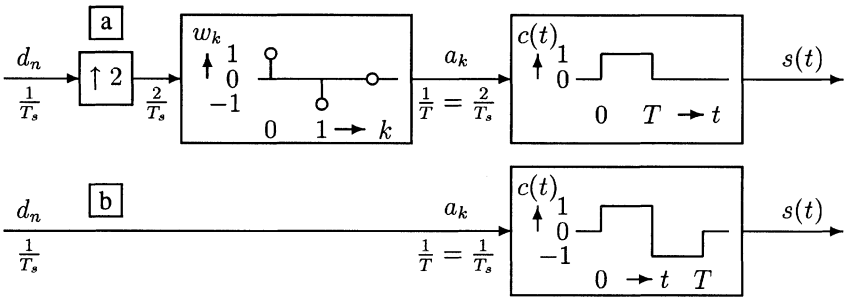


Figure 3.2: Two equivalent manners to produce a Bi-phase signal.

Data symbols a_k are normally constrained to a discrete set that is referred to as the *signal constellation*, *data alphabet* or just *alphabet*. Of particular interest are alphabets with evenly spaced data levels whose center of gravity is the origin. These are sometimes called L-AM alphabets where L denotes the number of data levels and the addition 'AM' refers to the fact that information is represented in the form of distinct amplitude levels (Amplitude Modulation), as opposed to distinct phases and/or frequencies in certain radio-frequency modulation techniques. Table 3.1 presents some of these alphabets. In the table, the distance between adjacent data levels has been normalized to 2. We will adopt this normalization as a default.

Alphabet	Data levels	σ_a^2
2-AM	± 1	1
4-AM	$\pm 1, \pm 3$	5
8-AM	$\pm 1, \pm 3, \pm 5, \pm 7$	21
16-AM	$\pm 1, \pm 3, \ldots, \pm 15$	85

Table 3.1: Overview of some AM data alphabets. Amplitudes are normalized for a distance 2 between adjacent data levels. Also listed is the data variance σ_a^2 for the situation that all data levels are equally likely.

The AM alphabets are normally used in Pulse Amplitude Modulation (PAM). Here groups of N bits d_n are transmitted as symbols from the 2^N-AM alphabet. The resulting data sequence a_k is generally uncorrelated, with all data levels being equally likely. The corresponding variance $\sigma_a^2 \triangleq E[a_k^2]$ is also listed in Table 3.1. When used in conjunction with a binary hold function $c(t)$ of width T, the 2-PAM and 4-PAM schemes are commonly called NRZ (Non-Return-to-Zero) and 2B1Q (2-binary-to-1-quaternary), respectively.

The power P_s of the transmitted signal $s(t)$ depends both on the power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of a_k and on the symbol response $c(t)$ of the modulator. More in par-

ticular

$$P_s = \frac{1}{T}\int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega})|C(\Omega)|^2 d\Omega \tag{3.1}$$

where $C(\Omega)$ denotes the transfer function of the modulator, i.e. the Fourier integral of $c(t)$. We recall that Ω is a normalized measure of frequency, with $\Omega = 1$ corresponding to the signaling rate $1/T$. With the spacing of adjacent data levels fixed at 2, we cannot constrain P_s via the data a_k, i.e. $c(t)$ will have to be scaled properly in order to obtain a prescribed transmit power.

3.2.3 Channel

In Fig. 3.1, the channel delivers a received signal $r(t)$ that is a noisy and distorted version of $s(t)$. In the remainder of this chapter and throughout most of the following chapters we shall use the channel model of Fig. 3.3. This model accounts only for linear intersymbol interference (ISI) and additive stationary noise. The emphasis on these two disturbances is warranted by their predominance in baseband transmission, as exemplified in Chapter 2. Supplemental disturbances such as nonlinear distortion, DC offsets, cross-talk and transition jitter will receive more restricted attention. Specifically, nonlinearities can be analyzed as in Appendices 2A and 2D, and counteracted as in Appendix 2B at the transmitting end of the system and as in Section 6.3 and Example 7.12 at the receiving end. DC-offsets can be handled via adaptive DC-compensation (Section 8.8.2) or via DC-insensitive detection techniques (Appendix 7C). As argued in Section 2.3.5 and Appendix 2C, cross-talk and transition jitter can often be modeled as additive and stationary and thus lumped into the model of Fig. 3.3.

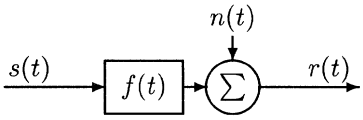


Figure 3.3: Baseline channel model.

The channel of Fig. 3.3 has impulse response $f(t)$ and additive noise $n(t)$ that is assumed to be stationary and Gaussian with zero mean. The power spectral density of $n(t)$ is denoted $\mathcal{N}(\Omega)$. The *transfer function* $F(\Omega)$ of the channel is the Fourier integral of $f(t)$:

$$F(\Omega) \triangleq \int_{-\infty}^{\infty} f(t)e^{-j2\pi\Omega t/T}dt. \tag{3.2}$$

A quantity of particular interest is the *excess bandwidth* of the channel. This is the fraction of the bandwidth that extends beyond the *Nyquist frequency* $1/(2T)$ ($\Omega = 0.5$). Fig. 3.4 provides an illustration. The absolute and normalized frequencies f_c and Ω_c demarcate the upper edge of the passband of the channel. The *Nyquist rate* of the channel is the smallest rate at which the data component of $r(t)$ can be sampled without aliasing. A channel is sometimes said to be *wideband* if it has much excess bandwidth. Conversely, a *narrowband* channel has little if any excess bandwidth.

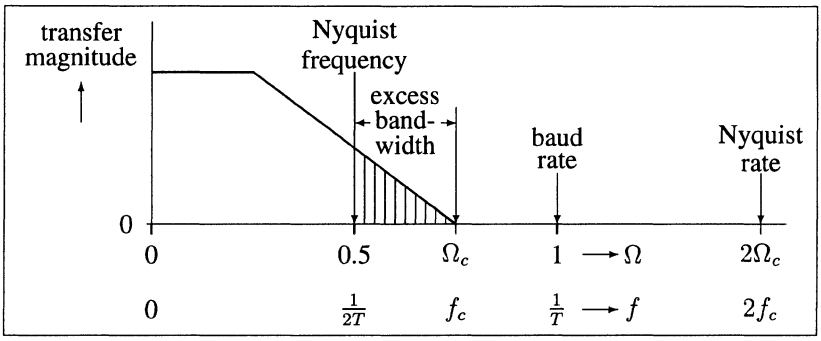


Figure 3.4: Illustration of Nyquist frequency, excess bandwidth, baud rate and Nyquist rate.

Exercise 3.1 Assume that the channel has 50% excess bandwidth. Express the Nyquist frequency in terms of the Nyquist rate. How large is the Nyquist rate for a channel without excess bandwidth?

There are several reasons why excess bandwidth is an important notion:

1. In practical systems, spectrum is costly and bandwidth tends to be a scarce commodity. As a result, excess bandwidth is always under pressure.

2. Intersymbol interference increases rapidly as excess bandwidth decreases (see Chapter 2 for illustrations). At the same time the received signal comes to contain progressively less timing information (see Chapters 9 and 10). Furthermore, receiver performance tends to become more sensitive to channel parameter variations. We will see in later chapters that these factors necessitate more effort in equalization, detection, timing recovery and adaptation, respectively. This translates into more complicated and thus expensive receivers. Notwithstanding these complications, many state-of-the-art systems tend to have little if any excess bandwidth.

Example 3.2 In the early days of digital magnetic recording, information densities were modest and excess bandwidth was plentiful. Nowadays, hard disk drives tend to operate at densities for which excess bandwidth is negligible. A similar remark applies to other modern digital recording systems, such as the compact disc.

In practice excess bandwidth is less sharply defined than Fig. 3.4 might suggest. The reason is that practical channels have a gradual roll-off and no sharply defined upper band edge. To resolve this ambiguity, we can take the power spectral density (PSD) of data and noise components at the channel output as a basis of reference. At some frequency Ω_0 the data PSD will drop below the noise PSD. Beyond this frequency the data is overwhelmed by noise and conveys little useful information. One might, therefore, regard Ω_0 as the upper band edge.

3.2.4 Hybrid channel

In the sequel we will find it convenient to regard the cascade of modulator and channel as a *hybrid channel* with discrete-time input a_k and continuous-time output $r(t)$ (Fig. 3.5). The received signal $r(t)$ has a data component that we can think of as the output of a linear pulse modulator with input a_k and symbol response $h(t) \triangleq (c * f)(t)$. We shall refer to the Fourier integral $H(\Omega)$ of $h(t)$ as the *transfer function* of the hybrid channel. The noise component $n(t)$ is Gaussian and has power spectral density $\mathcal{N}(\Omega)$. We may write

$$r(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT) + n(t). \tag{3.3}$$

Each digit a_k is carried by a pulse $h(t-kT)$ that typically extends across multiple symbol intervals. Successive pulses then overlap. The ensuing intersymbol interference (ISI) tends to affect the performance of the data receiver adversely.

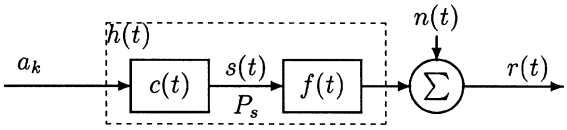


Figure 3.5: Hybrid channel model for the cascade of linear pulse modulator and the actual continuous-time channel.

Per-symbol signal-to-noise ratio measures Detection of a digit a_k in the face of noise is obviously easy if noise is small in the frequency region where most data energy is concentrated, i.e. if $|H(\Omega)|^2$ is large in relation to $\mathcal{N}(\Omega)$ in this region. In the sequel we will, in fact, see that receiver performances are governed by the spectral signal-to-noise ratio

$$\mathrm{SN}(\Omega) \triangleq \frac{1}{T}\frac{|H(\Omega)|^2}{\mathcal{N}(\Omega)} = \frac{1}{T}\frac{|C(\Omega)|^2|F(\Omega)|^2}{\mathcal{N}(\Omega)}. \tag{3.4}$$

The division by T is needed for SN to be dimensionless; since $|H|^2$ is an *energy* spectral density, $|H|^2/T$ is basically a power spectral density, just as $\mathcal{N}$. The definition of (3.4) does not make sense for frequencies Ω where both H and $\mathcal{N}$ are zero. Here we define $\mathrm{SN} = 0$. It can be observed that $\mathrm{SN}(\Omega)$ is a non-negative and symmetric function of frequency that is fully determined by the modulator and the actual continuous-time channel. The statistics of the data sequence do not come into play. The reason is that SN is basically associated to a single digit a_k, i.e. SN is a *per-symbol* signal-to-noise ratio measure.

The integral of SN across all frequencies defines the total per-symbol signal-to-noise ratio that we dub SNR:

$$\mathrm{SNR} \triangleq \int_{-\infty}^{\infty} \mathrm{SN}(\Omega) d\Omega. \tag{3.5}$$

In Section 3.9 we will see that this signal-to-noise ratio measure is closely related to the matched-filter bound, which sets an upper limit to the performance of any data receiver.

For future reference we also define an aliased version of SN according to

$$\mathrm{FSN}(e^{j2\pi\Omega}) \triangleq \sum_{n=-\infty}^{\infty} \mathrm{SN}(\Omega + n). \tag{3.6}$$

This *folded signal-to-noise ratio* is dimensionless, non-negative and periodic in Ω with period 1. It is important because it fully determines discrete-time channel models that are equivalent to the one of Fig. 3.5 in terms of attainable performance (see Sections 3.10 and 3.10.2). These models, in turn, fully determine the optimum performance of such data receivers as the maximum-likelihood sequence detector (Section 3.11), the linear equalizer (Chapter 5), and the partial-response and decision-feedback equalizers (Chapter 6). It is worth observing that FSN also determines SNR. In the terminology of Section 1.5, we may regard SNR as the *arithmetic* average of FSN, i.e.

$$\mathrm{SNR} = \overline{\mathrm{FSN}}^{A} = \int_{-0.5}^{0.5} \mathrm{FSN}(e^{j2\pi\Omega}) d\Omega. \tag{3.7}$$

In Chapters 5 and 6 we will find that the optimum performance of various equalizers can also be expressed as spectral averages of FSN. Appendix 3A illustrates the above signal-to-noise ratio measures for a Lorentzian digital magnetic recording system.

Example 3.3 Assume that $H(\Omega)$ is a low-pass characteristic and that noise is white. Then $\mathrm{SN}(\Omega)$ will also be a low-pass function. In Fig. 3.6 it is assumed that $\mathrm{SN}(\Omega)$ extends beyond the Nyquist frequency $1/(2T)$ ($\Omega = 0.5$). For this reason the aliases of $\mathrm{SN}(\Omega)$ which make up the folded signal-to-noise ratio $\mathrm{FSN}(e^{j2\pi\Omega})$ exhibit overlap. The total signal-to-noise ratio SNR is the area under the curve $\mathrm{SN}(\Omega)$, but is obviously also equal to the area under the curve $\mathrm{FSN}(e^{j2\pi\Omega})$ across a single fundamental interval.

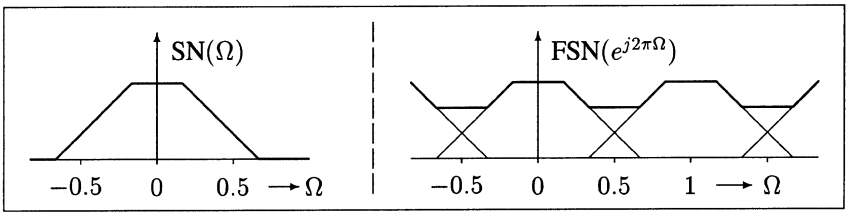


Figure 3.6: Illustration of spectral signal-to-noise measures for a low-pass channel.

Specialization to white noise Many practical systems have a noise PSD that is approximately flat throughout the passband of the channel. Frequencies outside the passband are normally rejected by a filter in the receiver and the precise noise PSD in this stopband is, consequently, irrelevant to receiver performance. For mathematical convenience it is then reasonable to assume that noise is white and indeed this assumption

is often made. The irrelevance of the out-of-band noise spectrum is also manifest in SN in that $\mathrm{SN}(\Omega) = 0$ irrespective of $\mathcal{N}(\Omega)$ whenever $H(\Omega) = 0$.

For white noise we may write $\mathcal{N}(\Omega) = N_0$ where N_0 defines the noise level. Then from (3.5) and (3.4)

$$\mathrm{SNR} = \frac{1}{N_0}\frac{1}{T}\int_{-\infty}^{\infty} |H(\Omega)|^2 d\Omega. \tag{3.8}$$

Upon invoking Parseval's equality this becomes

$$\mathrm{SNR} = \frac{1}{N_0}\int_{-\infty}^{\infty} h^2(t)dt = \frac{E_b}{N_0} \tag{3.9}$$

where $E_b \triangleq \int_{-\infty}^{\infty} h^2(t)dt$ is the energy of the symbol response $h(t)$. This shows that SNR is directly related to the received data energy per transmitted bit. A similarly direct relation does not exist when noise is colored.

Signal-to-noise measures of the received signal The data component of $r(t)$ has power spectral density $\mathcal{A}(e^{j2\pi\Omega})|H(\Omega)|^2/T$, i.e. $r(t)$ has spectral signal-to-noise ratio

$$\mathrm{SN}_r(\Omega) \triangleq \mathcal{A}(e^{j2\pi\Omega})\frac{|H(\Omega)|^2}{T\mathcal{N}(\Omega)} = \mathcal{A}(e^{j2\pi\Omega})\mathrm{SN}(\Omega) \tag{3.10}$$

where the last equality is due to (3.4). Hence $\mathrm{SN}_r(\Omega)$ is fully determined by $\mathcal{A}(e^{j2\pi\Omega})$ and $\mathrm{SN}(\Omega)$. Conversely, $\mathrm{SN}(\Omega)$ may be measured from $r(t)$ with the aid of an uncorrelated data sequence a_k of unit power ($\mathcal{A}(e^{j2\pi\Omega}) = 1$). The average of SN_r across frequency defines the total signal-to-noise ratio SNR_r of the received signal:

$$\mathrm{SNR}_r \triangleq \int_{-\infty}^{\infty} \mathrm{SN}_r(\Omega)d\Omega = \int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega})\mathrm{SN}(\Omega)d\Omega. \tag{3.11}$$

This shows that SNR_r may be viewed as a weighted average of $\mathrm{SN}(\Omega)$. Problem 3.4 explores the relation between SNR_r and $\mathrm{FSN}(e^{j2\pi\Omega})$.

Example 3.4 The NRZ and Bi-Phase modulation codes both use binary data $a_k \in \{-1, 1\}$. Data a_k is normally approximately uncorrelated so that $\mathcal{A} \simeq 1$ for all Ω. The implication is that $\mathrm{SN}_r \simeq \mathrm{SN}$ for all Ω and $\mathrm{SNR}_r \simeq \mathrm{SNR}$.

3.2.5 Receiver

The receiver operates on the channel output $r(t)$ to produce decisions $\hat{d}_n$ with respect to the source data symbols d_n (Fig. 3.7). To this end it first converts $r(t)$ into a discrete-time sequence y_k at the data rate $1/T$. This action is, in a sense, the inverse of the linear pulse modulation in Fig. 3.1 and is therefore designated demodulation. We will see that it involves such steps as suppression of out-of-band noise, equalization of the data spectrum, and a sampling operation at instants t_k that is synchronous with

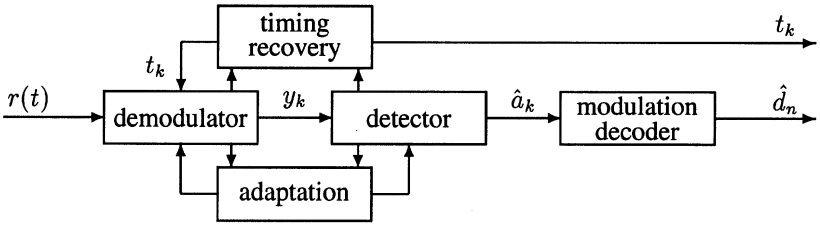


Figure 3.7: Baseline receiver topology.

the transmitted data a_k. This synchronization does not come for free but must be established by means of a separate timing-recovery subsystem. Timing recovery is critical to the performance of the receiver and is covered in detail in Chapters 9-11. The detector operates on y_k to produce decisions $\hat{a}_k$ with respect to a_k, which are converted into decisions $\hat{d}_n$ by means of a modulation decoder. Depending on the modulation code, this decoder may suffer from *error propagation* (*error extension*) effects that cause a single erroneous symbol $\hat{a}_k$ to produce several erroneous bits $\hat{d}_n$. Detection techniques come in many tastes and are discussed in Chapters 5-7.

Proper operation of the receiver requires knowledge of the actual characteristics of the code and channel. To the extent that this knowledge is not available *a priori* it may be necessary to acquire it *a posteriori* from the received signal $r(t)$ and possible side-information. For example, a circuit for automatic gain control (AGC) is often included in the demodulator to undo uncertainties of the received signal level. Adaptive equalizers and/or detectors may be needed when uncertainties extend beyond this parameter. These topics are elaborated in Chapter 8.

The topology of Fig. 3.7 is only meant to give an idea of the operations that are needed to convert a received signal into synchronous bit decisions. Many variations are encountered. For example, it is sometimes possible to detect $\hat{d}_n$ directly, with no need to form 'intermediate' decisions $\hat{a}_k$. A case in point is the partial-response system with precoding that will be discussed in Chapter 6. As another example, the sequence y_k is sometimes only present implicitly in the form of a continuous-time detector input $y(t)$ in conjunction with decision instants t_k that are indicated to the detector by the timing recovery subsystem.

This exposition correctly suggests that the receiver is, in general, more complicated than the transmitter, both conceptually and in terms of hardware. This relative complexity is reflected in the emphasis that is given to the receiver in the remainder of this text. Only a single chapter is devoted to modulation codes, while all other chapters are concerned directly or indirectly with the receiver.

3.2.6 Data sink

The data sink in Fig. 3.1 represents the destination of the source data sequence d_n. It accepts digits $\hat{d}_n$ that are ideally equal to the corresponding symbols d_n. Design of

the transmitter and receiver should be aimed at minimizing the likelihood of erroneous decisions $\hat{d}_n$ or at optimizing a related performance criterion.

The decisions $\hat{d}_n$ are paced by a clock that is derived from the timing recovery subsystem. This clock may act as a master clock for the operations that are 'hidden' in the data sink. Alternatively, an elastic buffer may be used to convert between the channel clock and a master clock within the sink [17, Chapter 17].

3.3 OPTIMUM MODULATION AND DEMODULATION FOR AN IDEAL LOW-PASS CHANNEL

The cascade of modulator, channel and demodulator in Fig. 3.1 can be viewed as a discrete-time channel that transforms a discrete-time input a_k into a linearly filtered and noisy version y_k. This is the same type of action as performed by the original continuous-time channel, and it is of interest to know to what extent these two channels are (or can be) equivalent. We first address this issue for a particularly simple channel, viz. one with white noise whose transfer function is an ideal low-pass function. In Section 3.5 we generalize the insights gained to a wider variety of channels.

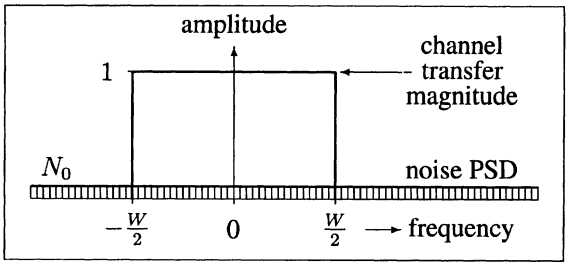


Figure 3.8: Transfer magnitude and noise spectrum of ideal low-pass channel of one-sided bandwidth $W/2$ with white noise.

Let us assume that the channel has a one-sided bandwidth of $W/2$ Hz and unit transfer ($F = 1$) throughout the passband. Noise $n(t)$ is Gaussian and white with power spectral density N_0 (Fig. 3.8). The modulator delivers a signal $s(t)$ that carries the data information (cf. Fig. 3.1). The power P_s of $s(t)$ is constrained. Design of the modulator may be based on the following considerations:

1. Frequencies outside the passband of the channel cannot get across the channel but do contribute to P_s. To avoid a possible waste of valuable transmit power, we should eliminate these 'useless' frequencies. This requires a symbol response $c(t)$ with no spectral content beyond $[-W/2, W/2]$.

2. The modulator should be able to produce any signal $s(t)$ within the passband of the channel. Would it not, then we could not make optimum use of the channel and this would restrict the amount of information that we can transport. Based on the sampling theorem we conclude that the signaling rate $1/T$ must be at least $2.W/2 = W$, and that the passband of $c(t)$ should cover the entire passband of the channel. Together with the foregoing consideration this means that the passband of $c(t)$ must be exactly as wide as that of the channel.

For convenience we choose the smallest allowable signaling rate $1/T = W$ and we take the transfer function C of $c(t)$ to be T across the passband. With the signaling rate $1/T$ now defined, we can express C in terms of normalized frequencies Ω according to

$$C(\Omega) = \begin{cases} T, & |\Omega| < 0.5, \\ 0, & \text{else.} \end{cases} \tag{3.12}$$

The corresponding function $c(t)$ is a sinc pulse

$$c(t) = \frac{\sin \pi t/T}{\pi t/T}. \tag{3.13}$$

With $c(t)$ given, the power P_s of $s(t)$ can only be constrained via the data sequence a_k, and more in particular via its variance $\sigma_a^2 \triangleq E[a_k^2]$.

Demodulator design is governed by similar considerations as those for the modulator. In particular, the demodulator must pass all data components of the received signal to avoid loss of information. Frequencies outside the passband of the channel contain only noise and are best rejected. This suggests that we use an ideal anti-aliasing filter whose bandwidth coincides with that of the channel. By virtue of the sampling theorem, the output of this filter can be sampled with no loss of information if the sampling rate is twice as high as the highest frequency at the output of the filter. Thus it suffices to sample at the symbol rate $1/T$. We arrive at Fig. 3.9.

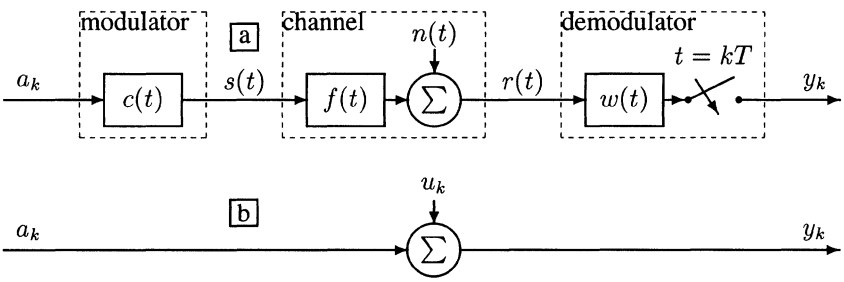


Figure 3.9: a. Ideal low-pass channel with optimum modulator and demodulator; b. equivalent discrete-time channel.

The system of Fig. 3.9.a contains three ideal low-pass operations with the same bandwidth (the block with impulse response $w(t)$ denotes the anti-aliasing filter). The

overall symbol response of the system up until the anti-aliasing filter is equal to $c(t)$ and yields a Kronecker delta function when sampled at the instants $t = kT$. Thus for every input symbol a_k there is a single symbol $y_k = a_k + u_k$ at the output of the sampler, where u_k denotes filtered and sampled noise. This noise sequence is white and Gaussian with zero mean and variance $\sigma_u^2 = N_0/T$. We may, therefore, describe y_k as the output of the *equivalent discrete-time channel* of Fig. 3.9.b. The term 'equivalent' is warranted because we have not lost any information in wrapping the continuous-time channel into a discrete-time 'shell'. The only impairment in the channel of Fig. 3.9.b is additive Gaussian white noise (AWGN). For this reason it is sometimes referred to as the AWGN channel.

3.4 CAPACITY OF AN IDEAL LOW-PASS CHANNEL WITH WHITE GAUSSIAN NOISE

The *capacity* C of a channel is, crudely speaking, the maximum amount of information that the channel can transport per unit of time. Processing complexity and delay are not restricted, but the information should get across with complete fidelity, i.e. with zero probability of error. At a first glance it might seem strange that any information at all can be transported with complete fidelity across a channel with Gaussian noise. After all, the noise samples may, on occasion, become very large. Even so, judicious exploitation of the law of large numbers permits a long message to be conveyed with a very low probability of symbol error. In the limit of an infinitely long message this probability can indeed go to zero [33].

It should be stressed that knowledge of C does not provide us with a constructive procedure to design close-to-optimum transmitters and receivers. For this reason the engineering value of capacity is restricted to that of a benchmark against which a practical transmission system can be judged. This benchmark can, moreover, at best be determined approximately in the usual case that the exact characteristics of the channel are *a priori* uncertain.

A derivation of capacity would require an elementary introduction to information theory and this would carry us too far afield. The interested reader is referred to the classical paper by Shannon in which the foundations of information theory were laid down [25], and to standard textbooks like those of Gallager [12] and Cover [7]. In Shannon's paper it is shown that the AGWN channel has capacity

$$C = \frac{1}{2}\log_2[1 + \text{SNR}] \text{ bits per symbol interval}, \tag{3.14}$$

where $\text{SNR} \triangleq \sigma_a^2/\sigma_u^2$ denotes the signal-power-to-noise-power ratio of the channel.[2] The two-logarithm in (3.14) accounts for the fact that capacity is measured in bits. In

[2]The symbol SNR is commonly used for this ratio, and we adhere to this custom throughout this section. Elsewhere we invariably we use the symbol as in (3.5).

terms of the original continuous-time quantities we have

$$C = \frac{W}{2}\log_2[1 + \frac{P_s}{N_0 W}] \text{ bits per second.} \tag{3.15}$$

This is the classical Hartley-Shannon formula for the capacity of a continuous-time channel of bandwidth W with white additive Gaussian noise [25]. Capacity C increases monotonically with W and for $W \rightarrow \infty$ approaches the asymptotic value $C_\infty = \log_2(e).P_s/N_0$ bits/s (Fig. 3.10).

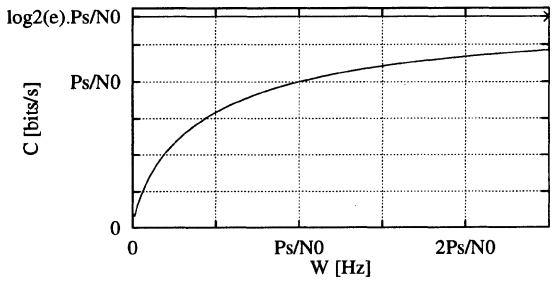


Figure 3.10: Channel capacity C as a function of bandwidth W.

According to (3.14), C increases at high SNRs (SNR $\gg 1$) by one bit per transmission for every 6 dB that SNR is increased. This is intuitively reasonable. A decrement of noise power by 6 dB means that noise amplitude has halved, i.e. that data levels a_k can be spaced twice as closely for the same relative effect of noise. Thus twice as many data levels can be reliably discriminated from each other, i.e. one additional bit of information can be conveyed. This heuristic reasoning may suggest that admissible data levels a_k are necessarily discrete. This is, however, by no means true. In fact, in order to achieve capacity, a_k must have a Gaussian probability density function [25]. Even so, practical transmission schemes almost always use discrete data alphabets. It is obvious that any restriction that is imposed on the size of the alphabet can only reduce capacity. For example, a binary alphabet $\{-1, 1\}$ permits at most one bit of information to be represented, and C cannot be larger than 1 bit per symbol interval. More generally, we can convey at most n bits per symbol interval with a data alphabet of cardinality 2^n. This is reflected in the curves of Fig. 3.11, which were first compiled by Ungerboeck [28]. The curves pertain to some of the AM alphabets that were introduced in Section 3.2.2. Also indicated are the Hartley-Shannon limit and the SNRs that are required with uncoded PAM transmission to achieve a symbol-error rate of 10^{-5}. At poor SNRs even small alphabets permit the Hartley-Shannon limit to be approached closely. This 'robustness', along with their simplicity, explains in part why small data alphabets are often used.

In order to convey one bit per symbol interval with uncoded transmission (i.e. with a 2-AM constellation) we need an SNR of at least 12 dB, and even then we would only approach $C = 1$, i.e. we would still have to tolerate occasional bit-errors. By comparison, the 4-AM curve of Fig. 3.11 promises a capacity in excess of 1 bit per symbol

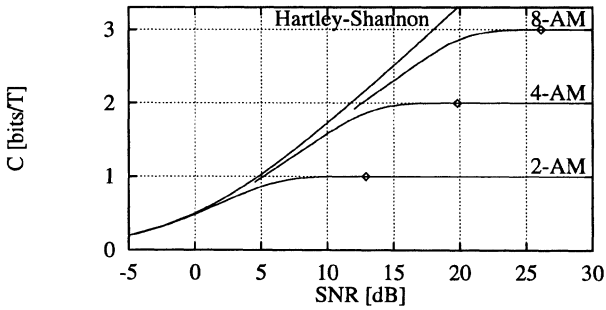


Figure 3.11: Channel capacity C of AWGN channels with discrete-valued and continuous-valued inputs. After Ungerboeck [28].

interval for SNRs of only 5 to 6 dB. This amounts to a saving in SNR of well over 6 dB. To achieve this gain we need to use a *redundant signal set*, i.e. an alphabet that contains more data levels than strictly needed to convey the intended number of bits. Furthermore, we must have an encoder that allocates redundancy in such a manner as for the receiver to be able to exploit it towards an SNR advantage. The more powerful -and hence complicated- the encoder/receiver combination, the closer one can get to the ultimate gain projected in Fig. 3.11. Similar SNR advantages accrue when more than 1 bit is to be conveyed per symbol interval. In all cases it suffices to use a data alphabet that is twice as large as the 'minimal' one. Beyond that cardinality the law of diminishing return sets in.

Channel coding with redundant signal sets, often referred to as *trellis coded modulation* (TCM), was first described by Ungerboeck [28] and has been a hotbed of research activity. It is now a mature field with spin-offs in a variety of products, notably in voiceband modems. A good introduction to the subject area is [29] (see also [30], [9] and [10]).

3.5 CAPACITY OF A CATEGORY OF NOISY DISPERSIVE CHANNELS

Practical channels do not have the ideal low-pass characteristic assumed thus far and do not normally have white noise. It is, therefore, of interest to extend the foregoing considerations to more realistic channel models. Typically a more or less smooth passband B is preceded and/or followed by a gradual or rapid roll-off. Outside B the transfer magnitude becomes too small to be useful. The exact 'useful' band B depends in part on the available transmit power P_s, because added power will help to lift the high-frequency data components above the noise floor. Let us assume for a moment that B is known, and that the power spectral density $S(f)$ of $s(t)$ is given. All transmit power is, of course, concentrated within B, i.e. $S(f) = 0$ for $f \notin B$. Assume that B is sub-

divided into many narrow subbands of width Δf, where Δf is chosen small enough so that the spectral signal-to-noise ratio $\mathrm{SN}(f)$ is approximately constant within any one subband. Then, given $\mathcal{S}(f)$, the capacity within a sub-band of width Δf around frequency f_i is

$$\mathcal{C}(f_i) = \frac{\Delta f}{2} \log_2[1 + \mathcal{S}(f_i)\mathrm{SN}(f_i)] \text{ bits per second.} \tag{3.16}$$

For $\Delta f \to 0$, the aggregate capacity across the whole band is given by

$$\mathcal{C} = \frac{1}{2} \int_B \log_2[1 + \mathcal{S}(f)\mathrm{SN}(f)]df \text{ bits per second.} \tag{3.17}$$

The optimum transmit power spectrum $\mathcal{S}(f)$, along with the capacity-achieving band B, may be determined via the so-called *water-pouring* procedure that is described in Appendix 3B. For channels of practical interest, calculations show that it is important to optimize the transmission band B, but that typically a flat transmit power spectrum is almost as good as the optimum water-pouring spectrum, if B is nearly optimal [6].

A crude idea of the dependence of $\mathcal{C}$ on channel signal-to-noise ratio can be obtained by referring back to expression (3.15) for an ideal low-pass channel of bandwidth W. As $W \to \infty$ all bandwidth restrictions disappear, and C is a linear function of P_s/N_0 (see Fig. 3.10). By comparison, for any fixed value of W the channel is strictly band-limited and according to (3.15), $\mathcal{C} \propto \log_2(P_s/N_0)$ for sufficiently large P_s/N_0, i.e. $\mathcal{C}$ is a linear function of P_s/N_0 expressed in dB. Practical channels are in between infinite bandwidth and 'brickwall', and the dependence of $\mathcal{C}$ on signal-to-noise ratio is somewhere in between linear and linear in dB [23].

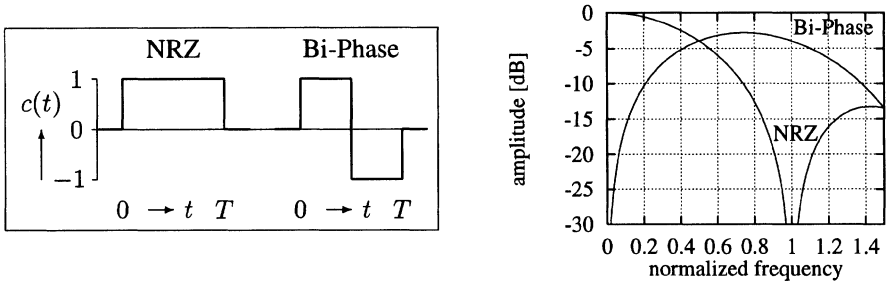
3.6 EFFECT OF EXCESS BANDWIDTH

So far we have avoided excess bandwidth by choosing a suitably high signaling rate $1/T$ and by using strictly bandlimited transfer functions in modulator and demodulator. Reality, however, tends to be different.

Modulator: The symbol response $c(t)$ is rarely if ever a sinc pulse as in (3.13). One reason is ease of implementation. The hold function $c(t)$ of width T, for example, emerges naturally in digital hardware and is therefore often used. A second reason is that some channels, notably in digital recording, accept only binary waveforms $s(t)$. This restricts $c(t)$ severely.

Example 3.5 NRZ and Bi-phase both produce a binary signal $s(t)$. For NRZ, $c(t)$ is a hold function of width T, versus a bipolar function of width T for Bi-phase (Fig. 3.12.a).

The transfer function $C(\Omega)$ for NRZ is a sinc function with spectral zeros at all integer multiples of the bit frequency (Fig. 3.12.b). Some 22 percents of the energy of $c(t)$ is located above the Nyquist frequency ($\Omega = 0.5$), i.e. excess bandwidth is significant.



a. Symbol responses $c(t)$ of NRZ and Bi-Phase.

b. Modulator transfer magnitude $|C(\Omega)|$ versus normalized frequency Ω.

Figure 3.12: Symbol responses $c(t)$ and amplitude characteristics $|C(\Omega)|$ of NRZ and Bi-phase.

Because of its band-pass nature, Bi-phase is ill-suited for low-pass channels, but shows up well for certain recording channels that block DC (see Appendix 3A). For such channels the modulation code ideally concentrates all transmit power within the passband of the channel. To this end, $|C(\Omega)|$ should be centered about this passband and the one-sided bandwidth should not exceed $0.5/T$ so as to permit all passband components of $s(t)$ to be manipulated independently. Bi-phase clearly falls short of the latter ideal and thus has significant 'excess bandwidth' even in the passband sense of the notion.

Further examples of functions $c(t)$ for various modulation codes may be found in Chapter 4. The use of a transmit filter also affects $c(t)$ (see Sections 5.7, 6.2.3 and 6.3). Excess bandwidth of $c(t)$ is almost always significant. There are two possibilities:

1. The channel has little or no excess bandwidth. Then a fraction of the transmit power will be rejected by the channel and thus wasted. This restricts capacity (see [14] for considerations in this context).

2. The channel does have significant excess bandwidth. Then the modulator has insufficient degrees of freedom to *independently* manipulate all spectral components of $s(t)$ that are within the passband of the channel. Equivalently, the modulator is unable to produce *all* signals $s(t)$ that can pass the channel. This also restricts capacity.

Demodulator: If the channel has significant excess bandwidth, then a demodulator filter that rejects all excess bandwidth and thereby a portion of the useful data energy can no longer be optimum. It is not *a priori* obvious how the filter should be dimensioned instead: bandwidth expansion would, after all, cause aliasing and consequential information loss. In Section 3.7 we will provide the solution to this problem. There we will see that the output of a so-called *matched filter* can be sampled at the symbol rate without losing any information about the data a_k. Any information loss that does occur pertains exclusively to noise and is irrelevant to data detection.

3.7 A HYPOTHETICAL MAXIMUM-LIKELIHOOD RECEIVER

In later chapters we will study a variety of reception schemes. As a basis of reference, we now develop and analyze a hypothetical maximum-likelihood receiver, shown in Fig. 3.13, that exploits a maximum amount of prior knowledge about the transmitter and channel. The performance of this receiver cannot possibly be improved upon, but its staggering computational complexity totally disqualifies it from practical application. Its operation, on the other hand, is conceptually straightforward. Also, its performance can be easily analyzed and this yields fundamental limits to the performance of any practical receiver. In Chapter 7 we will discuss the Viterbi detector, which performs maximum-likelihood detection in a recursive and much more practical fashion.

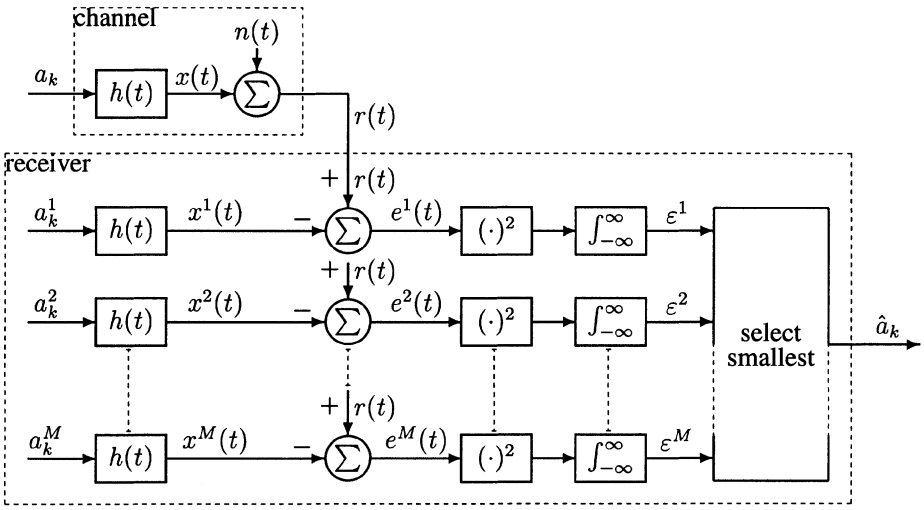


Figure 3.13: Hypothetical receiver for maximum-likelihood detection of a finite-length data signal corrupted by linear ISI and white Gaussian noise.

The channel model in Fig. 3.13 is the hybrid one of Fig. 3.5. For the time being we assume that noise $n(t)$ is Gaussian and white. The receiver of Fig. 3.13 is only applicable to data sequences $\underline{a} \triangleq [a_0, \ldots, a_{N-1}]$ that have a *finite* length N. The total number M of possible sequences depends on the modulation encoder and on the data alphabet. For example, for 2B1Q any succession of quaternary digits a_k is possible, so that $M = 4^N$. The receiver of Fig. 3.13 has full knowledge of the encoding rule in that it has a list of all M possible data sequences $\underline{a}^1, \ldots, \underline{a}^M$. It also has full knowledge of the symbol response $h(t)$. This implies, among other things, that the total delay on the channel is known, i.e. that an ideal timing reference is available in the receiver. We shall see in Chapters 9, 10, 11 and 8 how a combination of timing-recovery and adaptation techniques allows $h(t)$ to be identified. Using this knowledge the receiver can

precompute the M possible channel output signals

$$x^i(t) \triangleq \sum_{k=0}^{N-1} a_k^i h(t - kT) \tag{3.18}$$

that would be received if noise were absent and $\underline{a}^i$ were transmitted. Since the actually transmitted data sequence $\underline{a}$ is one of the M possible sequences $\underline{a}^i$ $(i = 1, \ldots, M)$, $r(t)$ is certain to be a noisy version of one of the M signals $x^i(t)$. The receiver simply observes which of these signals matches $r(t)$ most closely, and outputs the corresponding data sequence as its decision. Its measure of closeness is the *unweighted Euclidian distance*

$$\varepsilon^i \triangleq \int_{-\infty}^{\infty} [r(t) - x^i(t)]^2 dt \tag{3.19}$$

between $r(t)$ and $x^i(t)$. This is an intuitively reasonable measure: $r(t)$ differs from $x(t)$ through the influence of noise $n(t)$. For exactly one of the candidate signals $x^i(t)$ does the error signal $e^i(t) = r(t) - x^i(t)$ consist only of noise $n(t)$, while for all other candidate signals a data component will also be present. This data component is likely to increase the energy of the error signal. Since selection is based on this energy, the actually transmitted sequence is likely to be selected. In Appendix 3C we show more formally that the *distance metric* of (3.19) results in detection of the most likely transmitted data sequence, provided that noise $n(t)$ is Gaussian and white. We will see later that other metrics are more appropriate if $n(t)$ is colored or non-Gaussian. It is worth stressing that the receiver of Fig. 3.13 detects a complete data sequence at once. This is normally called *sequence detection* (or, less accurately, *sequence estimation*), as opposed to *symbol-by-symbol detection* in which symbols are detected one by one.

The receiver of Fig. 3.13 relies primarily on the fact that all possible signals $x^i(t)$ can be precomputed. This means that linearity of the channel is not at all a prerequisite, i.e. that nonlinear ISI can in principle be handled as well. Chapter 7 discusses variants of the Viterbi detector with this ability.

Maximum-likelihood receiver based on matched filter The reception process of Fig. 3.13 can be brought into a largely discrete-time form. To see how, we expand the metric of (3.19) as

$$\varepsilon^i = \int_{-\infty}^{\infty} r^2(t)dt - 2\int_{-\infty}^{\infty} r(t)x^i(t)dt + \int_{-\infty}^{\infty} (x^i)^2(t)dt. \tag{3.20}$$

In optimizing ε^i over all $i \in \{1, \ldots, M\}$, the term $\int_{-\infty}^{\infty} r^2(t)dt$ represents a bias that does not depend on i and can therefore be neglected without affecting the solution. The energy $E^i \triangleq \int_{-\infty}^{\infty} (x^i)^2(t)dt$ does not depend on $r(t)$ and can be precomputed for all i. The only term that remains to be computed upon reception of $r(t)$ can be rewritten with the help of (3.18) as

$$-2\int_{-\infty}^{\infty} r(t)x^i(t)dt = -2\int_{-\infty}^{\infty} r(t)\sum_{k=0}^{N-1} a_k^i h(t - kT)dt = -2\sum_{k=0}^{N-1} a_k^i y_k, \tag{3.21}$$

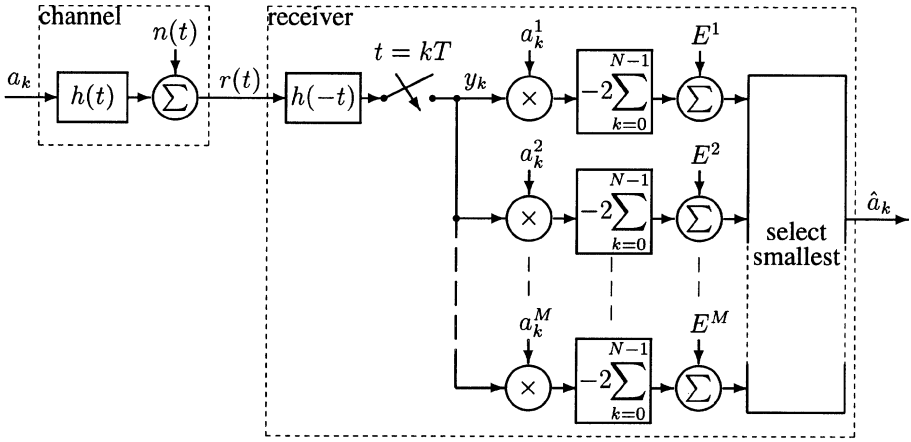


Figure 3.14: With the exception of a matched filtering operation, the maximum-likelihood detection process of Fig. 3.13 can be recast in a fully discrete-time form.

where

$$y_k \triangleq \int_{-\infty}^{\infty} r(t)h(-(kT-t))dt. \tag{3.22}$$

This discrete-time sequence may be recognized as the sampled output of a filter with impulse response $h(-t)$ operating on the received signal $r(t)$. This filter is said to be a *matched filter* because its impulse response is the time-inversed version of the symbol response $h(t)$ of the hybrid channel. The matched filter is, in general, not causal. To arrive at a causal implementation we must add a certain delay. Symbol decisions will be delayed accordingly. This delay is normally acceptable and we will neglect it in the sequel, i.e. we will neglect the causality issues that are associated to the matched filter and its derivatives, such as the whitened matched filter of Section 3.10.2.

Based on these considerations we see that maximum-likelihood sequence detection can alternatively be performed by minimizing the function $E^i - 2\sum_{k=0}^{N-1} a_k^i y_k$ over all i. The implied receiver structure is shown in Fig. 3.14. Instead of a continuous-time integration, one addition and a discrete-time cross-correlation are performed for each candidate data sequence $\underline{a}^i$. The matched filtering operation occurs only once and hardly contributes to the overall complexity of the reception process. A direct and important consequence of the receiver structure of Fig. 3.14 is the following theorem:

Theorem 3.1 *Maximum-likelihood detection of a transmitted data sequence of length N is possible on the basis of N consecutive samples $y_0, \ldots, y_{N-1}$ of the output of a matched filter.*

Thus, instead of observing a continuous-time signal $r(t)$ during an infinite time-interval, it suffices to observe a discrete-time sequence y_k during a finite interval. In statistical terminology one calls the set $\{y_0, \ldots, y_{N-1}\}$ a *sufficient statistic* for optimum

detection of a_k. It is worth noting that the theorem does not impose any conditions on the modulation code or data alphabet. Furthermore, there are no restrictions on the symbol response $h(t)$ and therefore excess bandwidth can be arbitrarily large. Thus, although substantial aliasing may occur in the sampling process, no information about a_k gets lost whatsoever. At a first glance, this fact may seem at odds with the loss of information that is always associated with sampling below the Nyquist rate. This paradox is resolved by noting that a lot of information can still get lost along the way, such as information about the noise component $n(t)$ of $r(t)$ and about the timing of the data sequence. None of this information is, however, relevant to the detection process. To prevent any loss of timing information it is, in general, necessary to resort to oversampling [20] (see also Chapter 10).

Notwithstanding their theoretical significance, matched filters do have some drawbacks that restrict their practicality:

1. Sampling must occur exactly at the instants $t = kT$. Especially for wideband systems, virtually all information about a_k is concentrated at this timing phase, and even small sampling-phase errors result in serious performance degradations.

2. The matched filter has the same amplitude-frequency characteristics as the channel and thereby enhances any channel amplitude distortion. The total amplitude distortion of y_k may be so severe that it can only be dealt with at the cost of extreme dynamic ranges in the discrete-time part of the receiver.

Although the receivers of Figs. 3.13 and 3.14 have a performance that cannot be improved upon, their practical realization is extremely difficult to say the least. Even for moderate message lengths N, the total number M of possible data sequences already becomes staggering, and for each of these a fairly complicated measure of resemblance has to be computed. In Chapter 7 we will see that the Viterbi detector (VD) performs a recursive detection process equivalent to the one of Figs. 3.13 and 3.14 at a per-symbol complexity that does not grow with the message length N. In spite of the fact that it meets this basic condition of practicality, the VD still tends to have a high computational complexity. For this reason Chapter 7 devotes ample space to simplifications of the VD.

3.8 NOISE WHITENING

Before analyzing the performance of the receivers of Figs. 3.13 and 3.14 we generalize the above results to colored Gaussian noise $n(t)$. In Fig. 3.15.a, a *noise-whitening filter* with transfer function $W(\Omega) = 1/\sqrt{\mathcal{N}(\Omega)}$ transforms the received signal $r(t)$ with colored Gaussian noise into a signal $r'(t)$ with a white noise component of power spectral density 1. Such a filter exists when $\mathcal{N}(\Omega)$ is nonzero for all frequencies, and does not reject any information about a_k because its operation can in principle be fully undone by an inverse filter with transfer function $\sqrt{\mathcal{N}(\Omega)}$.[3] It is convenient to

[3]By a more involved argument, the present reasoning can be generalized to virtually all noise spectra that may be encountered in practice, including those that vanish outside the passband of $H(\Omega)$ or with pointwise singularities within the passband of $H(\Omega)$.

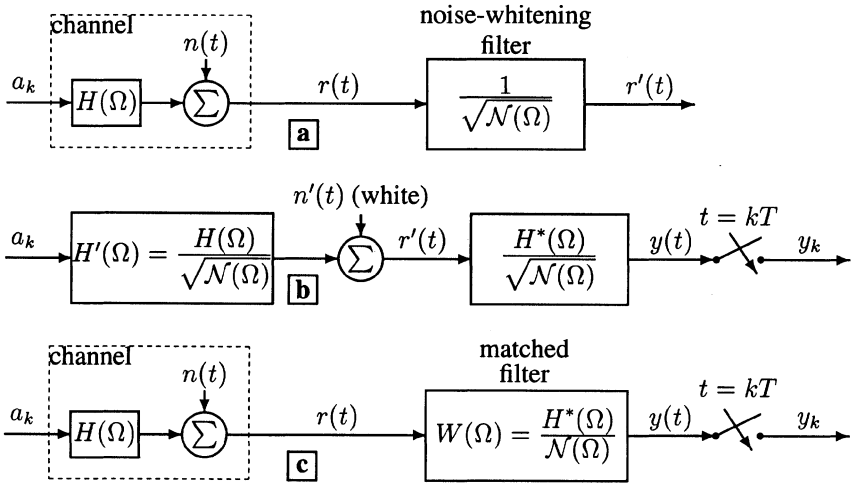


Figure 3.15: a. System with colored noise $n(t)$ and a noise whitening filter; b. Equivalent system with white noise $n'(t)$. The sufficient statistic y_k is obtained by applying $r'(t)$ to a filter matched to $r'(t)$ and sampling at the instants $t = kT$; c. Alternative method for constructing y_k directly from $r(t)$ with the help of a filter matched to $r(t)$.

think of $r'(t)$ as in Fig. 3.15.b. The whitening operation is considered to form part of the operation of the channel. This modified channel has transfer function $H'(\Omega) = H(\Omega)/\sqrt{\mathcal{N}(\Omega)}$ and white Gaussian noise. With Theorem 3.1 we now see that a sufficient statistic y_k about a_k is obtained with the help of a filter with transfer function $H^*(\Omega)/\sqrt{\mathcal{N}(\Omega)}$. Here '*' represents complex conjugation, the frequency-domain analog of time-reversal. By comparing Figs. 3.15.a and 3.15.b we see that we can in fact construct y_k directly from $r(t)$ by means of a filter with transfer function

$$W(\Omega) = \frac{H^*(\Omega)}{\mathcal{N}(\Omega)} \tag{3.23}$$

(see Fig. 3.15.c). This is the transfer function of the matched filter for colored noise.

3.9 MATCHED-FILTER BOUND

The receiver of Fig. 3.14 is applicable for any message length N. The simplest possible situation arises if we transmit a single binary symbol a_0 only. This prevents intersymbol interference (ISI) from arising, enabling the receiver to concentrate all of its attention on dealing with noise. The resulting performance puts an upper limit to what can be achieved in the presence of ISI, i.e. for message lengths N greater than 1. According to Theorem 3.1, y_0 is a sufficient statistic for maximum-likelihood detection of a_0. Since the two possible values $a_0 = -1$ and $a_0 = 1$ are symmetric about the origin,

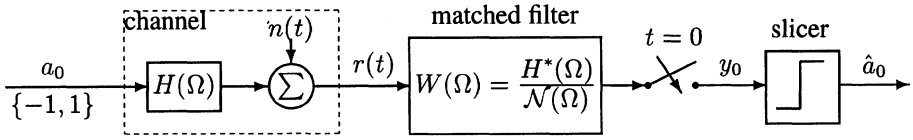


Figure 3.16: Maximum-likelihood detection of single bit a_0 requires a matched filter, a sampler and a slicer.

discrimination between them should be based on the polarity of y_0. This leads to the optimum *one-shot* receiver of Fig. 3.16.

To analyze the performance of this receiver, we use (3.3) to write the decision variable y_0 as

$$y_0 = \int_{-\infty}^{\infty} r(\tau)w(0-\tau)d\tau = a_0 \int_{-\infty}^{\infty} h(t)w(-t)dt + \int_{-\infty}^{\infty} n(t)w(-t)dt, \tag{3.24}$$

where $w(t)$ is the impulse response of the matched filter. We see that y_0 consists of two components:

1. A data component with amplitude

$$\begin{aligned} A &\triangleq \int_{-\infty}^{\infty} h(t)w(-t)dt = \frac{1}{T}\int_{-\infty}^{\infty} H(\Omega)W(\Omega)d\Omega \\ &= \frac{1}{T}\int_{-\infty}^{\infty} \frac{|H(\Omega)|^2}{\mathcal{N}(\Omega)}d\Omega = \int_{-\infty}^{\infty} \mathrm{SN}(\Omega)d\Omega = \mathrm{SNR}. \end{aligned} \tag{3.25}$$

   The equalities are due to Parseval's relation, (3.23), and definitions (3.4) and (3.5), respectively.

2. A zero-mean noise component with variance

$$\sigma^2 = \frac{1}{T}\int_{-\infty}^{\infty} \mathcal{N}(\Omega)|W(\Omega)|^2 d\Omega = \frac{1}{T}\int_{-\infty}^{\infty} \frac{|H(\Omega)|^2}{\mathcal{N}(\Omega)}d\Omega = \mathrm{SNR}. \tag{3.26}$$

   Here the first equality follows from recognizing σ^2 as the power of the noise component behind the matched filter, which is independent of the precise sampling instant by stationarity of $n(t)$. Since $n(t)$ is Gaussian, its filtered and sampled version must also be Gaussian.

The *predetection signal-to-noise ratio* $\rho_{\mathrm{MFB}} \triangleq A^2/\sigma^2$, therefore, amounts to

$$\rho_{\mathrm{MFB}} = \mathrm{SNR}. \tag{3.27}$$

This result is referred to as the *one-shot* or *matched-filter bound.* It is important for various reasons.

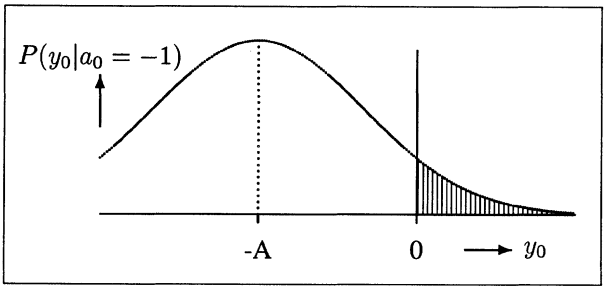


Figure 3.17: Conditional probability density function of y_0 for $a_0 = -1$.

1. It puts an upper limit to the performance of any receiver for channels hampered by linear ISI and noise. This limit is often but not always attainable by means of maximum-likelihood sequence detection, as we will see later.

2. It shows that the entire SNR of the channel is made available by the matched filter at a single discrete moment ($t = 0$). This reconfirms the information-lossless nature of the matched-filter/sampler combination.

3. It underscores the decisive influence of the channel SNR on ultimately attainable performance levels. In this respect, the fine-structure of amplitude distortion, i.e. its precise distribution across frequencies, is according to (3.27) irrelevant: what matters is only a weighted average. The phase distortion of the channel plays no role whatsoever. Of course, to achieve close-to-optimum performance levels, reception techniques do need to take precise account of amplitude and phase distortion as well as noise spectra, as we will see later.

With the aid of (3.27) we can easily determine the probability $\Pr[E]$ that the receiver makes an erroneous decision. To this end we first consider the event that $a_0 = -1$. Then y_0 is Gaussian with variance σ^2 about a conditional mean $E[y_0|a_0 = -1] = -A$ (Fig. 3.17). A decision error is made if the noise component of y_0 is so large that y_0 becomes positive. The probability $\mathrm{P}[E|a_0 = -1]$ that this occurs is determined by the hatched area in Fig. 3.17, i.e.

$$\Pr[E|a_0 = -1] = \frac{1}{\sqrt{2\pi\sigma^2}} \int_0^\infty e^{-\frac{(y-A)^2}{2\sigma^2}} dy. \tag{3.28}$$

This integral cannot be evaluated analytically but may be expressed in terms of the Q-function that was defined in Section 1.4 (equation (1.81)):

$$\Pr[E|a_0 = -1] = Q(\frac{A}{\sigma}) = Q(\sqrt{\rho_{\mathrm{MFB}}}) = Q(\sqrt{\mathrm{SNR}}). \tag{3.29}$$

By symmetry, $\Pr[E|a_0 = 1] = \Pr[E|a_0 = -1]$. The unconditional probability of error $\Pr[E]_{\mathrm{MFB}}$ is the average of both and therefore also given by (3.29), i.e.

$$\Pr[E]_{\mathrm{MFB}} = Q(\sqrt{\rho_{\mathrm{MFB}}}) = Q(\sqrt{\mathrm{SNR}}). \tag{3.30}$$

This leads to the *waterfall curve* of Fig. 3.18. Curves of error rate versus signal-to-noise ratio almost always have this shape. For error rates of practical interest (in the range, say, of 10^{-5} to 10^{-10}), an extra dB of SNR will improve the error rate by roughly an order of magnitude.

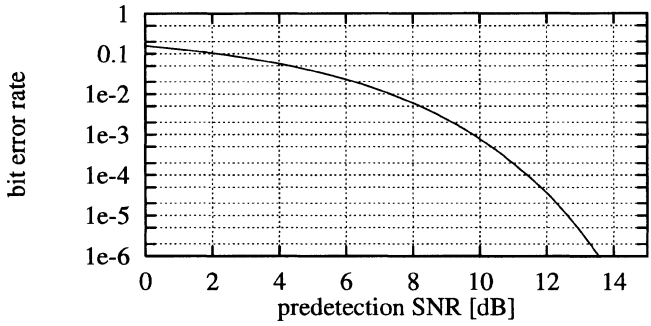


Figure 3.18: Probability of bit-error Pr[E] versus predetection signal-to-noise ratio ρ for binary signaling in the presence of Gaussian noise. For a receiver that achieves the matched-filter bound we may equate ρ with SNR.

3.10 EQUIVALENT DISCRETE-TIME CHANNEL MODELS

The sequence y_k can be thought of as resulting from a_k through a discrete-time filtering operation and addition of discrete-time Gaussian noise. Such a discrete-time characterization is simpler than the one that we have employed so far, and facilitates mathematical analysis. We now derive two discrete-time channel models that are equivalent to the original hybrid channel in terms of attainable performance. Later we use one of these models to analyze this performance.

3.10.1 Equivalent discrete-time model for y_k

The continuous-time output signal $y(t)$ of the matched filter may be described as

$$y(t) = \sum_{k=-\infty}^{\infty} a_k (h * w)(t - kT) + (n * w)(t), \tag{3.31}$$

where $w(t)$ is the impulse response of the matched filter with transfer function $W(\Omega) = H^*(\Omega)/\mathcal{N}(\Omega)$ (see Fig. 3.15.c). The sampled version y_k of $y(t)$ may be written $y_k = (a * f)_k + u_k$, where $f_k \triangleq (h * w)(kT)$ is the sampled response of the system up until the matched filter, while $u_k \triangleq (n * w)(kT)$ is sampled and filtered noise. The

corresponding discrete-time model for y_k is illustrated in Fig. 3.19. Both the Fourier transform $F(e^{j2\pi\Omega})$ of f_k and the power spectral density $\mathcal{U}(e^{j2\pi\Omega})$ of u_k are intimately related to the folded signal-to-noise ratio $\text{FSN}(e^{j2\pi\Omega})$ of (3.6):

1. Since f_k is a sampled version of $(h * w)(t)$, $F(e^{j2\pi\Omega})$ must be an aliased version of $H(\Omega)W(\Omega)$ according to

$$F(e^{j2\pi\Omega}) = \frac{1}{T}\sum_{n=-\infty}^{\infty} H(\Omega+n)W(\Omega+n) = \frac{1}{T}\sum_{n=-\infty}^{\infty} \frac{|H(\Omega+n)|^2}{\mathcal{N}(\Omega+n)} = \text{FSN}(e^{j2\pi\Omega}). \tag{3.32}$$

   The last equality is a direct application of definitions (3.4) and (3.6).

2. Similarly, the noise PSD behind the sampler is a folded version of the noise PSD $\mathcal{N}(\Omega)|W(\Omega)|^2 = |H(\Omega)|^2/\mathcal{N}(\Omega) = T\text{SN}(\Omega)$ just before it, i.e.

$$\mathcal{U}(e^{j2\pi\Omega}) = \frac{1}{T}\sum_{n=-\infty}^{\infty} T\text{SN}(\Omega+n) = \text{FSN}(e^{j2\pi\Omega}). \tag{3.33}$$

This shows that the model for y_k of Fig. 3.19 is completely determined by $\text{FSN}(e^{j2\pi\Omega})$. Amplitude distortion of the channel is only reflected in $\text{FSN}(e^{j2\pi\Omega})$ in relation to the noise spectrum $\mathcal{N}(\Omega)$. The phase characteristics of the channel do not affect $\text{FSN}(e^{j2\pi\Omega})$ at all and are, consequently, immaterial to ultimately attainable performance levels.

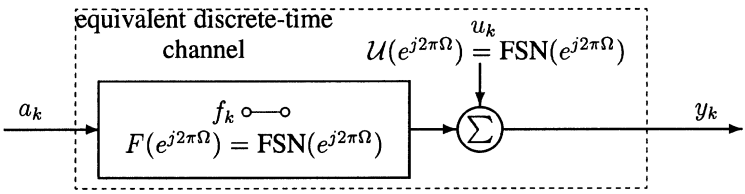


Figure 3.19: Discrete-time model for the sampled output y_k of the matched filter in Fig. 3.15.c.

3.10.2 The whitened matched filter

The detector that forms part of a data receiver normally has a discrete-time input and often works best when the noise component of this input is white. The noise sequence u_k in Fig. 3.19 is, in general, not white. In order to arrive at white noise we can apply a discrete-time noise-whitening filter as in Fig. 3.20. This filter has a periodic transfer function $C_w(e^{j2\pi\Omega})$ and should not be confused with the continuous-time noise-whitening filter of Section 3.8. In order to perform noise whitening the filter must have transfer magnitude

$$|C_w(e^{j2\pi\Omega})| = \frac{1}{\sqrt{\text{FSN}(e^{j2\pi\Omega})}} \tag{3.34}$$

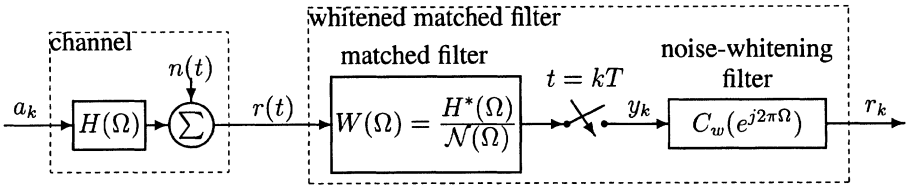


Figure 3.20: A discrete-time whitening filter may be used to transform y_k into a sequence r_k with white noise n_k of unit variance. The cascade of matched filter, sampler and whitening filter is referred to as a *whitened matched filter*.

and this function is well-defined whenever FSN has no spectral zeros.[4] Since $1/C_w(e^{j2\pi\Omega})$ can be approximated arbitrarily closely (apart from a delay), the whitening operation is reversible, and the output r_k of the filter is still a sufficient statistic for optimum detection of a_k. The cascade of a continuous-time matched filter, sampler and noise-whitening filter is referred to as a *whitened matched filter* or WMF [8], and is a basic element of some of the most powerful reception methods. The WMF has a continuous-time input and a discrete-time output and is thereby a time-variant filter. All filtering can be concentrated in the continuous-time domain by reversing the order of the sampler and transversal filter in Fig. 3.20. This leads to the alternative WMF structure of Fig. 3.21. This structure is equivalent to the one of Fig. 3.20 in that it produces the same output r_k (see Problem 1.7). It consists of a continuous-time filter with transfer function $W'(\Omega) \triangleq W(\Omega)C_w(e^{j2\pi\Omega}) = H^*(\Omega)C_w(e^{j2\pi\Omega})/\mathcal{N}(\Omega)$ and a sampler that closes at the instants $t = kT$. The filter degenerates into an ideal anti-aliasing filter in the particular but important event that the channel in Fig. 3.20 has white noise $n(t)$ and no excess bandwidth.

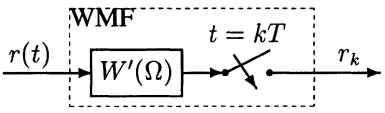


Figure 3.21: Alternative WMF structure.

So far we have not specified the phase characteristics of $C_w(e^{j2\pi\Omega})$. Since these characteristics can be chosen at liberty, infinitely many WMF's exist for any given continuous-time channel. Out of this infinitude, the so-called *canonical whitened matched filter* is of particular interest. This WMF causes the overall impulse response γ_k of channel, matched filter, sampler and whitening filter to have minimum phase. We recall that a function is said to have minimum phase if its z-transform has all poles and zeros inside the unit circle (see Section 1.6). We will find in Chapters 6 and 7 that this choice is especially important for the decision feedback equalizer and certain variants

[4]By a more involved argument, the present reasoning can be extended to functions FSN$(e^{j2\pi\Omega})$ with spectral zeros of finite multiplicity [8].

of the Viterbi detector. An equivalent discrete-time model for the output of the WMF is shown in Fig. 3.22.

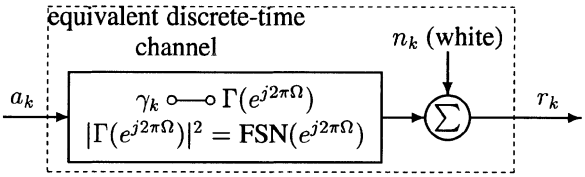


Figure 3.22: Equivalent discrete-time model for the output r_k of the whitened matched filter of Fig. 3.20. The noise sequence n_k is white and has unit variance.

The noise sequence n_k in Fig. 3.22 represents the white Gaussian noise component of r_k and has unit variance. The transfer function $\Gamma(e^{j2\pi\Omega})$ consists of two parts: a transfer function $\mathrm{FSN}(e^{j2\pi\Omega})$ that represents the cascade of continuous-time channel, matched filter and sampler (see Fig. 3.19), and the transfer function $C_w(e^{j2\pi\Omega})$ of the whitening filter. With (3.34) we see that $|\Gamma(e^{j2\pi\Omega})| = \sqrt{\mathrm{FSN}(e^{j2\pi\Omega})}$. Since the phase characteristics of $C_w(e^{j2\pi\Omega})$ can be chosen at liberty, the same applies to those of $\Gamma(e^{j2\pi\Omega})$. The canonical WMF causes γ_k to be a minimum-phase function.

3.11 PERFORMANCE OF MAXIMUM-LIKELIHOOD RECEIVERS

We proceed to assess the performance of a maximum-likelihood receiver operating on r_k. Since r_k is a sufficient statistic for optimum detection of a_k, no receiver that operates directly on the continuous-time signal $r(t)$ can ever do better. The receiver structure, shown in Fig. 3.23, is the discrete-time equivalent of the one of Fig. 3.13, and we only sketch its operation (see also Example 3.13).

The receiver has a list of all M possible data sequences $\underline{a}^1, \ldots, \underline{a}^M$ and their filtered versions $(a^1 * \gamma)_k, \ldots, (a^M * \gamma)_k$. The received sequence r_k is certain to be a noisy version of one of the filtered sequences $(a^i * \gamma)_k$, $i = 1, \ldots, M$. For each of these, the receiver calculates the Euclidian distance

$$\varepsilon^i = \sum_{k=-\infty}^{\infty} (r_k - (a^i * \gamma)_k)^2, \tag{3.35}$$

and settles on the sequence $\underline{a}^i$ for which ε^i is smallest. This decision will be wrong when some erroneous sequence $\underline{a}' \neq \underline{a}$ is closer to r_k in the Euclidian sense than the actually transmitted sequence $\underline{a}$. This is especially likely to happen when $\underline{a}'$ and $\underline{a}$ are hard to distinguish from each other after propagating through the channel, i.e. when $(a' * \gamma)_k$ and $(a * \gamma)_k$ look much alike. In that case even a small amount of noise n_k may cause the receiver to confuse $\underline{a}'$ with $\underline{a}$. Now $(a' * \gamma)_k$ and $(a * \gamma)_k$ will look much alike if and

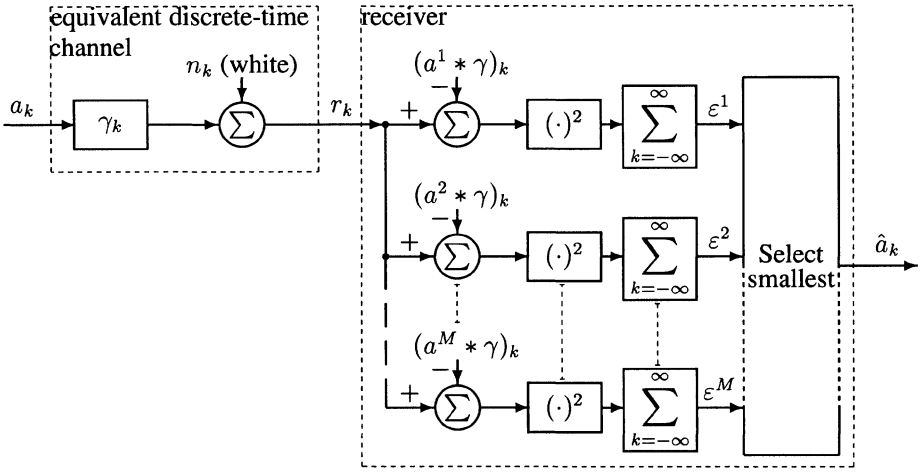


Figure 3.23: Discrete-time maximum-likelihood receiver operating on the output r_k of a whitened matched filter. This output is modeled as in Fig. 3.22.

only if the *error sequence* $\underline{e} \triangleq (\underline{a}' - \underline{a})/2$ propagates poorly across the channel[5], i.e. if $(e*\gamma)_k = [(a'*\gamma)_k - (a*\gamma)_k]/2$ is small. This smallness must, of course, be measured in Euclidian terms. This heuristic reasoning suggests that we consider the set S of all *admissible* error sequences $\underline{e}$, i.e. of all sequences $(\underline{a}' - \underline{a})/2$ for which both $\underline{a}'$ and $\underline{a}$ are admissible. For each $\underline{e} \in S$ we determine a *Euclidian weight*

$$d^2(\underline{e}) \triangleq \sum_{k=-\infty}^{\infty} (e * \gamma)_k^2 \tag{3.36}$$

as in Fig. 3.24. Performance of the MLSD is governed by the worst error sequence, i.e. by the sequence that minimizes $d^2(\underline{e})$. The minimum weight is commonly denoted $d^2_{\min}$. Appendix 3D shows that $d^2_{\min}$ may be interpreted as a predetection signal-to-noise ratio for the adopted normalization of error and noise sequences. At high SNRs the MLSD, therefore, achieves a symbol-error rate $\Pr[E]_{\mathrm{ML}} \simeq Q(\sqrt{\rho_{\mathrm{ML}}})$ where

$$\rho_{\mathrm{ML}} \triangleq d^2_{\min} = \min_{\underline{e} \in S} d^2(\underline{e}) \tag{3.37}$$

is the predetection signal-to-noise ratio of the MLSD.

It should be noted that the worst error sequence is, in general, not unique. This is so because a time shift or a change of polarity of an admissible error sequence does not affect its Euclidian weight and does not normally make it inadmissible. To overcome this non-uniqueness it is common to restrict attention to sequences $\underline{e}$ that start at moment $k = 0$ and have $e_0 > 0$.

[5]The division by 2 serves to simplify forthcoming formulas. In the literature it is sometimes omitted.

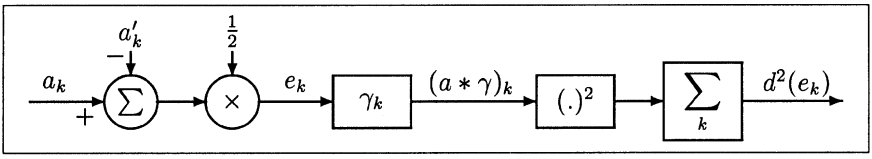


Figure 3.24: Circuit for computation of the Euclidian weight of an error sequence $\underline{e} = \frac{1}{2}(\underline{a}' - \underline{a})$.

Example 3.6 Consider a discrete-time channel whose impulse response γ_k has $\mathcal{D}$-transform $\gamma(\mathcal{D}) = 2 + \mathcal{D}$, and assume that data is uncoded and binary with $a_k \in \{-1, 1\}$.[6] Error symbols $e_k = (a'_k - a_k)/2$ are in the ternary alphabet $\{-1, 0, 1\}$. Since data is uncoded, any succession of binary digits a'_k and a_k is allowable. A direct implication is that any ternary error sequence $\underline{e}$ with symbols $e_k \in \{-1, 0, 1\}$ is admissible. Some of these sequences along with their filtered versions $(e * \gamma)_k$ are listed in $\mathcal{D}$-transform notation in Table 3.2. Also listed are their Euclidian weights.

$e(\mathcal{D})$	$e(\mathcal{D})\gamma(\mathcal{D})$	$d^2(e(\mathcal{D}))$
1	$2 + \mathcal{D}$	5
$1 + \mathcal{D}$	$2 + 3\mathcal{D} + \mathcal{D}^2$	14
$1 - \mathcal{D}$	$2 - \mathcal{D} - \mathcal{D}^2$	6
$1 + \mathcal{D} + \mathcal{D}^2$	$2 + 3\mathcal{D} + 3\mathcal{D}^2 + \mathcal{D}^3$	23
$1 - \mathcal{D} + \mathcal{D}^2$	$2 - \mathcal{D} + \mathcal{D}^2 + \mathcal{D}^3$	7

Table 3.2: Euclidian distance profile of the channel with $\gamma(\mathcal{D}) = 2 + \mathcal{D}$.

The reader may try other error sequences but will find that the sequence $e(\mathcal{D}) = 1$ has minimum weight. This shows that single bit-errors are most likely to occur. Such errors prevail, in fact, on many if not on most channels, and we will soon see that the matched-filter bound is attainable in this case.

Example 3.7 Table 3.3 lists the Euclidian distance profile of the channel with $\gamma(\mathcal{D}) = 1 + 2\mathcal{D} + \mathcal{D}^2$.

$e(\mathcal{D})$	$e(\mathcal{D})\gamma(\mathcal{D})$	$d^2(e(\mathcal{D}))$
1	$1 + 2\mathcal{D} + \mathcal{D}$	6
$1 + \mathcal{D}$	$1 + 3\mathcal{D} + 3\mathcal{D}^2 + \mathcal{D}^3$	20
$1 - \mathcal{D}$	$1 + \mathcal{D} - \mathcal{D}^2 - \mathcal{D}^3$	4
$1 + \mathcal{D} + \mathcal{D}^2$	$1 + 3\mathcal{D} + 4\mathcal{D}^2 + 3\mathcal{D}^3 + \mathcal{D}^4$	36
$1 - \mathcal{D} + \mathcal{D}^2$	$1 + \mathcal{D} - \mathcal{D}^3 - \mathcal{D}^4$	4
$1 - \mathcal{D} + \mathcal{D}^2 - \mathcal{D}^3$	$1 + \mathcal{D} + \mathcal{D}^4 + \mathcal{D}^5$	4

Table 3.3: Euclidian distance profile of the channel with $\gamma(\mathcal{D}) = 1 + 2\mathcal{D} + \mathcal{D}^2$.

[6] We recall that the operator $\mathcal{D}$ corresponds to a delay of one symbol interval.

The single bit-error has weight $d^2(1) = 6$. Unlike the previous example, this is not the smallest possible weight. For example, the sequence $e(\mathcal{D}) = 1 - \mathcal{D}$ corresponds to a double bit-error at the instants $k = 0$ and $k = 1$ and has weight $d^2(e(\mathcal{D})) = 4 < 6$. In fact, all alternating sequences $e(\mathcal{D}) = 1 + \mathcal{D} - \mathcal{D}^2 + \mathcal{D}^3 - \ldots$ have this same weight, irrespective of their length. The reader may try other sequences but will find that the alternating ones have minimum weight. It follows that multiple bit-errors prevail and that the predetection signal-to-noise ratio is a factor 6/4, i.e. 1.77 dB, worse than in the situation that single bit-errors would prevail. We will soon see that this factor may be interpreted as a loss with respect to the matched-filter bound. Problem 3.14 illustrates that 1.77 dB is not the largest possible loss for a channel of memory length $M = 2$. A systematic procedure to find the worst-case channels of any given memory length M was described in [18]. Losses for some of these channels are determined in Problems 3.14 and 3.15, and are found to increase rapidly with M.

Example 3.8 Digital recording systems sometimes use binary modulation codes in which the data sequence a_k consists of runs of identical symbols with a prescribed minimum length (see Chapter 4). For example, a minimum run-length of 2 symbols means that the string '$\ldots - - + + + - - + + + + \ldots$' is allowed while strings like '$\ldots + + - + + \ldots$' and '$\ldots - - - + - - + + \ldots$' are forbidden. Run-length constraints obviously restrict the category of admissible error sequences $\underline{e}$. For example, for minimum run-lengths of 2 and higher, one easily verifies that alternating error sequences of the form $\underline{e} = \ldots + - + - \ldots$ are not admissible. This causes single bit-errors to prevail for the channel of Example 3.7, i.e. there is no longer a loss with respect to the matched-filter bound. Maximum-likelihood reception becomes particularly simple in this case (see Example 7.7 in Chapter 7).

Spectral interpretation of ρ_{ML} It is instructive to interpret ρ_{ML} in spectral terms. To this end we first apply Parseval's equality to (3.36). This yields

$$d^2(\underline{e}) = \int_{-0.5}^{0.5} |E(e^{j2\pi\Omega})|^2 |\Gamma(e^{j2\pi\Omega})|^2 = \int_{-0.5}^{0.5} |E(e^{j2\pi\Omega})|^2 \mathrm{FSN}(e^{j2\pi\Omega}) d\Omega \tag{3.38}$$

where $E(e^{j2\pi\Omega})$ denotes the discrete-time Fourier transform of e_k. The second equality is a principal result of Section 3.10.2 (see also Fig. 3.22). The predetection signal-to-noise ratio ρ_{ML} is just the smallest possible Euclidian weight $d^2(\underline{e})$. We can distinguish two cases.

1. If $\mathrm{FSN}(e^{j2\pi\Omega})$ is not a strong function of frequency then the Euclidian weight of an error sequence is essentially an unweighted measure of its energy. The smallest possible energy emerges for a single symbol-error in which we confuse two adjacent data levels. We recall that we have normalized data alphabets for a distance of 2 between adjacent levels. For this reason the error sequence with minimum energy is single bit-error with $e_k = \delta_k$. This sequence has $|E(e^{j2\pi\Omega})| = 1$ for all Ω, whence

$$\rho_{\mathrm{ML}} = \int_{-0.5}^{0.5} \mathrm{FSN}(e^{j2\pi\Omega}) d\Omega = \mathrm{SNR} = \rho_{\mathrm{MFB}}. \tag{3.39}$$

The MLSD, therefore, achieves the matched-filter bound (MFB) whenever single symbol-errors prevail. It can be shown that is so, among other cases, when

FSN($e^{j2\pi\Omega}$) dips nowhere more than 6 dB below its average value $\overline{\text{FSN}} = \text{SNR}$ [27]. Hence even significant amounts of ISI need not render the MFB unattainable.

2. If FSN($e^{j2\pi\Omega}$) varies greatly with frequency then *multiple symbol-errors* may prevail. The corresponding error sequences have a larger energy than those of single symbol errors, but this energy is concentrated in the frequency region where FSN($e^{j2\pi\Omega}$) is poorest. As a result, less energy is left after the channel than for single bit-errors.

Exercise 3.2 Compute FSN($e^{j2\pi\Omega}$) for the channels of Examples 3.6 and 3.7. Do these channels corroborate the above interpretation?

Example 3.9 For certain binary modulation codes, the set S of admissible error sequences $\underline{e}$ excludes the category that corresponds to single bit-errors. Depending on the channel, such codes may actually yield a performance *gain* with respect to the MFB. This is illustrated in Example 7.5 and Appendix 7A.3.

Example 3.10 Let us consider an equivalent discrete-time channel with the *bipolar* impulse response $[\gamma(\mathcal{D}) = 1 - \mathcal{D}]$ and white Gaussian noise of variance 1. This channel has folded signal-to-noise ratio FSN($e^{j2\pi\Omega}$) $= |\Gamma(e^{j2\pi\Omega})|^2 = 4\sin^2 \pi\Omega$ (Fig. 3.25, left). The error sequences $e(\mathcal{D}) = 1 + \mathcal{D} + \mathcal{D}^2 + \ldots + \mathcal{D}^{N-1}$ of arbitrary length $N \geq 1$ have filtered versions $e(\mathcal{D})\gamma(\mathcal{D}) = 1 - \mathcal{D}^N$ and weight $d^2(e(\mathcal{D})) = 2$. No sequences have smaller weight. The shortest minimum-weight sequence $e(\mathcal{D}) = 1$ corresponds to the single bit-error, i.e. the MFB is achievable. Even so, performance is on the verge of breaking away from this bound because infinitely many error sequences of length $N > 1$ also have minimum weight. The right part of Fig. 3.25 depicts amplitude spectra $|E(e^{j2\pi\Omega})|$ of some minimum-weight sequences. For $N > 1$ a major fraction of the energy is concentrated near DC. This is indeed the spectral region where FSN($e^{j2\pi\Omega}$) is poorest.

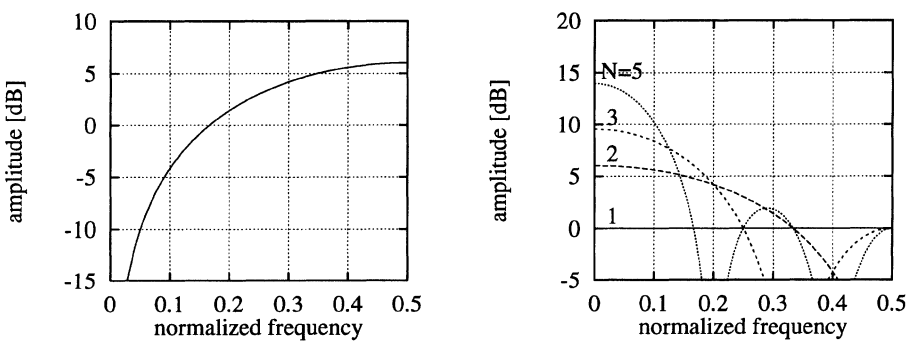


Figure 3.25: Folded signal-to-noise ratio FSN($e^{j2\pi\Omega}$) (left) and amplitude characteristics $|E(e^{j2\pi\Omega})|^2$ of several minimum-weight error sequences (right) for bipolar channel.

Example 3.11 Appendix 3A.1 determines FSN($e^{j2\pi\Omega}$) for NRZ and Bi-Phase on a Lorentzian channel (see also Fig. 3A.1.e). With Bi-Phase, FSN($e^{j2\pi\Omega}$) is essentially flat at low information

densities D. Thus single bit-errors prevail and there is no loss with respect to the MFB. As D increases, low-frequency depressions of $\mathrm{FSN}(e^{j2\pi\Omega})$ increase to the point where multiple bit-errors come to prevail. Worst-case error sequences $\underline{e}$ depend on D but all have their energy concentrated at low frequencies. In this region losses with respect to the MFB increase rapidly with D (Fig. 3.26.b).

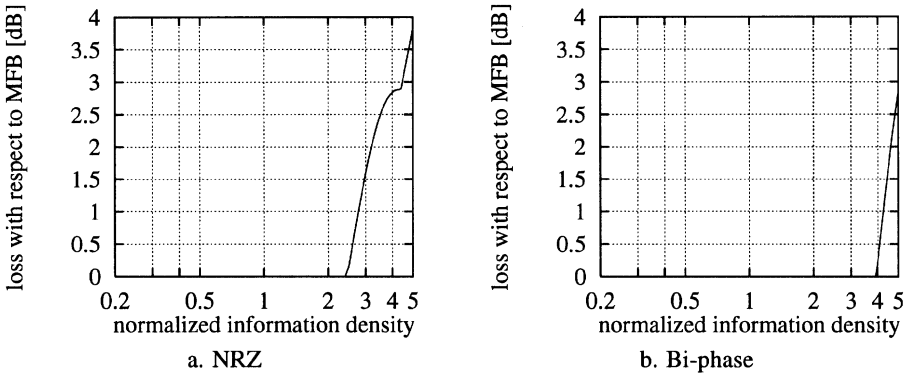


Figure 3.26: Loss $\rho_{\mathrm{ML}}/\rho_{\mathrm{MFB}}$ of maximum-likelihood receiver with respect to the matched-filter bound (MFB) as a function of the normalized information density D. The channel is Lorentzian with white Gaussian noise.

A similar behavior occurs with NRZ, be it that $\mathrm{FSN}(e^{j2\pi\Omega})$ now has its largest depression at high frequencies for large D. This causes worst-case error sequences to have a high-frequency nature. Problem 3.20 is concerned with an analysis of one of these sequences. The MFB is attainable for densities D up to $\sqrt{6} \simeq 2.45$, beyond which multiple bit-error prevail.

For the sake of completeness we note that the MFBs for NRZ are separated by a factor $3/(1+4D^2)$ in predetection signal-to-noise ratio (compare (3A.9) and (3A.10)). This factor must be accounted for in any comparison between Figs. 3.26.a and 3.26.b.

It is worth mentioning that the foregoing analysis applies to any linear discrete-time channel with additive Gaussian white noise, irrespective of whether this channel originates from the original continuous-time channel via a WMF or via a suboptimum demodulator. In the latter case noise n_k is not necessarily white. Appendix 3D analyzes the impact of colored noise on MLSD performance.

3.12 FURTHER READING

The notion of capacity has early precursors in the work of Hartley [13] and was put on a rigorous mathematical footing in the seminal paper of Shannon [25]. These papers, as well as [26], are well worth reading for their historical value and the important insights that they convey. Gallager's book [12] has long been a standard text on information theory and includes a detailed treatment of continuous-time channels. A more recent standard text is [7]. Evaluation of capacity tends to be hard, and several

categories of channels still defy analysis. For example, digital recording channels are often constrained to have binary input signals and this constraint has so far only permitted capacity to be bounded [14]. Practical modulators and reception techniques tend to be suboptimum and this results in a detraction from capacity that was first studied in the lucid paper of Price [23]. In Chapter 6 we will delve deeper into this issue.

The timeless book of Wozencraft and Jacobs [33] has popularized the use of equivalent discrete-time channel models that are derived from the actual continuous-time channel via optimum modulation and demodulation steps. Although that book deals primarily with wideband channels, several of the basic results carry over directly to channels with intersymbol interference. This was recognized, for example, by Forney and exploited in his development of the whitened matched filter (WMF) [8]. The optimality of the canonical form of the WMF as a prefilter for the decision feedback equalizer was first recognized by Price [23]. Several standard texts on data transmission have adopted equivalent discrete-time models as a basic ingredient of their treatment (see, for example, [24] and [17]). The origins of the matched filter, which is at the heart of all of these discrete-time models, can be traced back to work of North on radar reception [21] (see also [32]).

Maximum-likelihood reception is highly impractical when carried out in the 'brute-force' manner described above, and early work in this field has concentrated on equivalent recursive procedures for which processing effort grows only linearly in the message length [5, 1]. Soon thereafter it was recognized that the Viterbi algorithm, which was developed originally for decoding of convolutional codes [31], could also be applied to intersymbol interference channels [8, 22, 16]. Viterbi detection has been a hotbed of activity ever since; Chapter 7 describes the basics as well as some of the main lines of development.

Space restrictions have precluded a discussion of several concepts that are closely related to the subject matter of this chapter, including trellis-coded modulation, trellis precoding and multicarrier modulation. We recommend [29, 30], [11] and [4] as points of entry for these subjects.

APPENDIX 3A

Signal-to-noise ratios of NRZ and Bi-Phase on a Lorentzian channel

Digital magnetic recording channels often accept only binary input signals $s(t)$. Such signals are delivered by the NRZ and Bi-Phase modulation codes of Section 3.2.2 and Example 3.5. Both codes are nonredundant, i.e. the symbol interval T is equal to the time T_s between source data bits. Their transfer magnitudes $|C(\Omega)|$ are depicted in Fig. 3A.1.a. NRZ has a low-pass nature and spectral zeros at all integer multiples of the bit frequency; Bi-phase has a band-pass nature and a zero at DC. In Section 2.3.5 we have seen that the Lorentzian model for digital magnetic recording channels is char-

acterized by a normalized information density D and a transfer function

$$F(\Omega) = j\pi\Omega e^{-\pi D|\Omega|} \tag{3A.1}$$

for all Ω. The factor $j\pi\Omega$ accounts for a differentiation in the playback head and yields a spectral zero at DC (Fig. 3A.1.b). High-frequency roll-off is determined by D. At low densities there is much high-frequency content and Bi-phase is well matched to the channel (compare Figs. 3A.1.a and 3A.1.b). At high densities, on the other hand, the pass-band of the channel shifts to lower frequencies, where NRZ has significantly more content than Bi-Phase. Thus the transfer function $H(\Omega) = C(\Omega)F(\Omega)$ of the hybrid channel gets largest at low densities for Bi-Phase, and at high densities for NRZ (Figs. 3A.1.c and 3A.1.d). Since noise is white, this behavior carries over directly to the spectral signal-to-noise ratio $\mathrm{SN}(\Omega) = |H(\Omega)|^2/(T\mathcal{N}(\Omega))$.

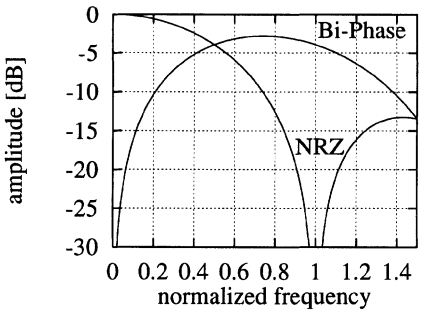
For NRZ, $\mathrm{SN}(\Omega)$ is zero via $F(\Omega)$ at DC, and via $C(\Omega)$ at all integer multiples of the symbol frequency. A direct consequence is that the folded signal-to-noise ratio $\mathrm{FSN}(e^{j2\pi\Omega})$ has a zero at DC (Fig. 3A.1.e). The upper half of the function exhibits a depression that is negligible for densities D below 1 and grows rapidly beyond that point. For Bi-Phase, $\mathrm{SN}(\Omega)$ is nonzero at the bit frequency, and this content folds back onto DC in Fig. 3A.1.e. The net effect is that $\mathrm{FSN}(e^{j2\pi\Omega})$ is virtually independent of frequency at low densities, but that its low-frequency portion deteriorates rapidly below that of NRZ as densities increase. We will see in the chapters ahead that even simple receivers can be close to optimum when $\mathrm{FSN}(e^{j2\pi\Omega})$ is more or less flat, as for Bi-Phase at low densities. Conversely, even complicated receivers tend to do a poor job when $\mathrm{FSN}(e^{j2\pi\Omega})$ exhibits deep depressions. These occur at high-densities in Fig. 3A.1.e for both codes.

Integration of $\mathrm{FSN}(e^{j2\pi\Omega})$ across the fundamental interval yields the per-bit channel signal-to-noise ratio SNR, which is depicted in Fig. 3A.1.f as a function of the normalized information density D. NRZ yields better SNRs at high densities than Bi-Phase, but roles reverse below $D = 1/\sqrt{2} \simeq 0.707$. The limiting gain of Bi-Phase over NRZ as $D \to 0$ amounts to a factor of 3, i.e. 4.77 dB. State-of-the-art systems use densities D in the order of 2. Here NRZ is to be preferred. The functions of Fig. 3A.1 can all be determined analytically (see Appendix 3A.1).

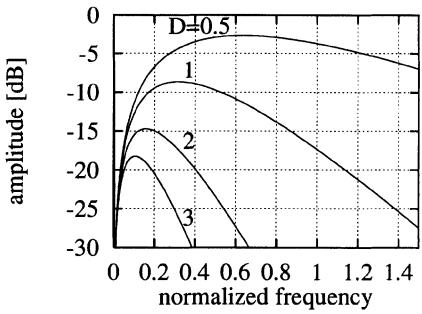
Exercise 3.3 Consider a periodic data sequence $a_k = \ldots + - + - + \ldots$ at the Nyquist rate. Show that the resulting transmit signal $s(t)$ is equal for NRZ and Bi-Phase, except for a delay. What is the implication for $\mathrm{SN}(\Omega)$ and $\mathrm{FSN}(e^{j2\pi\Omega})$? How is this visible in Figs. 3A.1.d and 3A.1.e?

3A.1 Analysis of NRZ and Bi-Phase on a Lorentzian channel

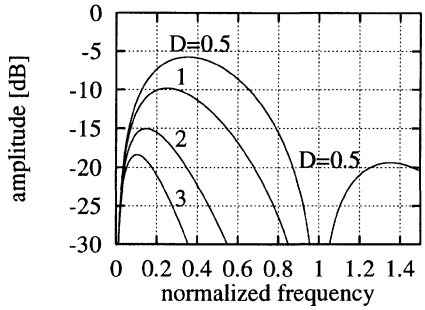
We proceed to analyze the characteristics of NRZ and Bi-Phase on a Lorentzian digital magnetic recording channel with differentiating playback head. Several of these functions were depicted in Fig. 3A.1. For future reference we take account of the effect



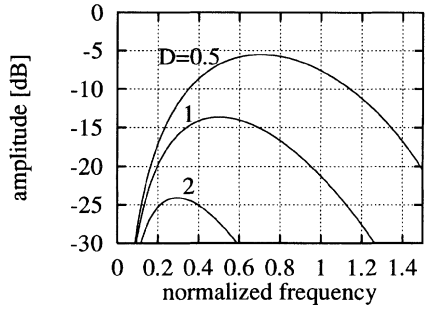
a. Modulator transfer magnitude $|C(\Omega)|$ versus normalized frequency Ω.



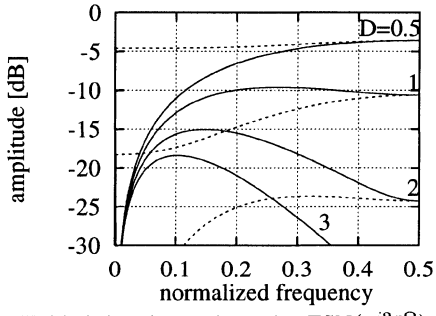
b. Channel transfer magnitude $|F(\Omega)|$ for several normalized information densities D.



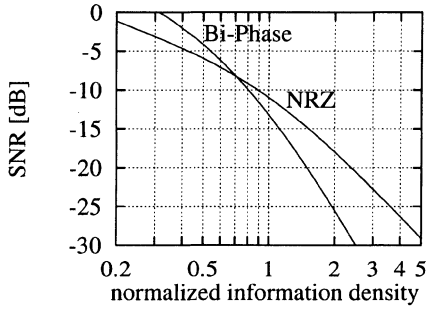
c. Transfer magnitudes $|H(\Omega)| = |C(\Omega)F(\Omega)|$ for NRZ.



d. Transfer magnitudes $|H(\Omega)| = |C(\Omega)F(\Omega)|$ for Bi-Phase.



e. Folded signal-to-noise ratios $\mathrm{FSN}(e^{j2\pi\Omega})$ for NRZ (solid) and Bi-Phase (dashed).



f. Signal-to-noise ratio SNR versus normalized information density D.

Figure 3A.1: Various functions pertaining to NRZ and Bi-Phase modulation codes when used in a Lorentzian digital magnetic recording system with differentiating playback head and white additive noise. Normalization along the vertical axis is in arbitrary units.

of a modulation code with rate R; the functions that were considered earlier on in this chapter all pertain to a rate $R = 1$. The modulator transfer functions $C(\Omega)$ associated with NRZ and Bi-Phase are

$$C(\Omega) = T\frac{\sin \pi\Omega}{\pi\Omega} \tag{3A.2}$$

and

$$C(\Omega) = jT\sin(\frac{\pi\Omega}{2})\frac{\sin(\frac{\pi\Omega}{2})}{(\frac{\pi\Omega}{2})}, \tag{3A.3}$$

respectively. The channel transfer function $F(\Omega)$ is given by

$$F(\Omega) = j\pi\Omega e^{-\pi D|\Omega|/R}. \tag{3A.4}$$

This generalizes (3A.1) to arbitrary code rates R. Noise $n(t)$ is white with power spectral density N_0. Spectral signal-to-noise ratios are

$$\mathrm{SN}(\Omega) = \frac{T}{N_0}\sin^2(\pi\Omega)e^{-2\pi D|\Omega|/R} \tag{3A.5}$$

for NRZ, and

$$\mathrm{SN}(\Omega) = \frac{4T}{N_0}\sin^4(\frac{\pi\Omega}{2})e^{-2\pi D|\Omega|/R} \tag{3A.6}$$

for Bi-Phase. Folded signal-to-noise ratios $\mathrm{FSN}(\Omega)$ may be determined by summing an infinite geometric series as in [3]. The result is

$$\mathrm{FSN}(e^{j2\pi\Omega}) = \frac{T}{N_0R}\sin^2(\pi\Omega)\frac{\cosh[2\pi\frac{D}{R}(0.5 - |\Omega|)]}{\sinh(\pi\frac{D}{R})} \tag{3A.7}$$

for NRZ, and

$$\mathrm{FSN}(e^{j2\pi\Omega}) = \frac{4T}{N_0R}\frac{\cos^4(\frac{\pi\Omega}{2})\cosh(2\pi\frac{D}{R}\Omega) + \sin^4(\frac{\pi\Omega}{2})\cosh[2\pi\frac{D}{R}(1 - |\Omega|)]}{\sinh(2\pi\frac{D}{R})} \tag{3A.8}$$

for Bi-Phase. These expressions are only valid in the fundamental interval $[-0.5, 0.5]$, beyond which symmetric extension applies. Signal-to-noise ratios $\mathrm{SNR} = \int_{-\infty}^{\infty}\mathrm{SN}(\Omega)d\Omega$ may be expressed in terms of standard integrals [3]. It is found that

$$\mathrm{SNR} = \frac{T}{N_0R}\frac{1}{2\pi\frac{D}{R}}\frac{1}{1 + (\frac{D}{R})^2} \tag{3A.9}$$

for NRZ, and

$$\mathrm{SNR} = \frac{T}{N_0R}\frac{1}{2\pi\frac{D}{R}}\frac{1}{1 + (\frac{D}{R})^2}\frac{3}{1 + 4(\frac{D}{R})^2} \tag{3A.10}$$

for Bi-Phase.

Exercise 3.4 Verify that the functions $C(\Omega)$ for NRZ and Bi-Phase do not depend on R.

APPENDIX 3B

Optimum distribution of transmit power across frequency

The optimum transmit power spectrum $\mathcal{S}(f)$ may be determined by a technique called *water-pouring* that is due to Gallager [12] (see also [11]). As shown in Fig. 3B.1, $\mathcal{S}(f)$ is nonzero over a band of frequencies $B = \{f : \mathrm{SN}^{-1}(f) < K_0\}$, called the *capacity-achieving band*, and within B is equal to

$$\mathcal{S}(f) = K_0 - \mathrm{SN}^{-1}(f) \tag{3B.1}$$

where K_0 is a threshold chosen so that $\mathcal{S}(f)$ satisfies the power constraint $\int_B \mathcal{S}(f)df = P_s$.

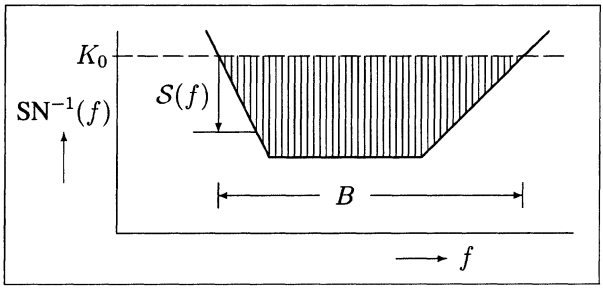


Figure 3B.1: Water pouring. Transmit power $\mathcal{S}(f)$ is 'poured' into the bowl $\mathrm{SN}^{-1}(f)$ until the prescribed total power P_s is obtained. This process determines the capacity-achieving band B, and the height K_0 determines $\mathcal{S}(f)$ within B according to $\mathcal{S}(f) = K_0 - \mathrm{SN}^{-1}(f)$.

With the aid of (3B.1) one readily ascertains that the channel capacity C of (3.17) amounts to

$$C = \int_B \log_2(K_0 \mathrm{SN}(f))df = B \log_2(K_0 G) \tag{3B.2}$$

where G denotes the geometric average of $\mathrm{SN}(f)$ across the capacity-achieving band, i.e.

$$G \triangleq \frac{1}{B} \exp \int_B \mathrm{SN}(f)df. \tag{3B.3}$$

Exercise 3.5 Prove (3B.2).

APPENDIX 3C

Maximum-likelihood detection

In this appendix we give a brief introduction to maximum-likelihood detection of signals in the presence of Gaussian noise. The first part of the appendix applies to signals in a variety of signal spaces. For this reason we shall initially not be specific about the duration and domain (time-continuous, time-discrete) of signals.

Let us assume that a signal s from a discrete set $S \triangleq \{s_1, \ldots, s_M\}$ is transmitted. Denote by $p(s_i)$ the *a priori* probability of occurrence of s_i, $i \in \{1, \ldots, M\}$. The received signal is a noisy version of s according to $r = s + n$ where n is noise that we will later take to be Gaussian. We wish to make a decision on which of the symbols $s_i \in S$ has been transmitted, based on observation of r (Fig. 3C.1).

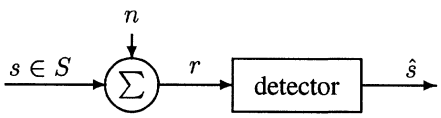


Figure 3C.1: Illustration of the maximum-likelihood detection problem.

An obvious approach would be to select the symbol $s_i \in S$ that has greatest *a posteriori* probability $p(s_i|r)$. This is referred to as maximum *a posteriori* probability (MAP) detection or Bayesian detection. With the help of Bayes' rule we may write

$$p(s_i|r) = \frac{p(r|s_i)p(s_i)}{p(r)} \tag{3C.1}$$

where $p(r|s_i)$ is the *likelihood* of r if s_i were transmitted and $p(r)$ is the *a priori* probability of r. Since $p(r)$ does not depend on s_i we can equivalently maximize $p(r|s_i)p(s_i)$ over all s_i. A disadvantage of this approach is that the *a priori* probabilities $p(s_i)$ must be known. If this is not so then a reasonable assumption may be used. For example, one might assume that all signals s_i have equal probability. Then $p(s_i)$ does not depend on i and maximization of $p(r|s_i)p(s_i)$ boils down to selecting the signal s_i with greatest likelihood $p(r|s_i)$. This is called *maximum-likelihood* (ML) detection.

In order to arrive at a constructive detection rule we must make use of the additive nature of noise. If a given signal $s_i \in S$ were transmitted, then noise n would have to equal $r - s_i$ in order to arrive at the observed received signal $r = s_i + n$. Thus the likelihood $p(r|s_i)$ is just the probability that noise n amounts to $r - s_i$. This probability can be computed if and only if the noise distribution function $p(n)$ is known. In the remainder of this appendix we shall restrict attention to Gaussian noise n. We first elaborate some particularly simple situations.

Example 3.12 Let s, n and r be scalars, and let n have zero mean and variance σ^2 so that

$$p(n) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{n^2}{2\sigma^2}\}. \tag{3C.2}$$

Then the likelihood function is given by

$$p(r|s_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{(r-s_i)^2}{2\sigma^2}\}. \tag{3C.3}$$

This function is to be maximized over all $s_i \in S$. This is obviously equivalent to minimizing $(r - s_i)^2$ over all s_i, i.e. to choosing the scalar $s_i \in S$ that is closest to r.

Exercise 3.6 Consider a set S with M unequal scalars $s_1, \ldots, s_M$. Show that maximum-likelihood detection is possible by means of a threshold detector with $M - 1$ thesholds. Where must these thresholds be located?

Example 3.13 Let s, n and r be discrete-time sequences of length N. To distinguish samples of these sequences from the complete sequences we shall add a subscript k if we refer to samples and we shall use boldface notation for complete sequences. Thus, for example, the kth sample of the noise sequence $\mathbf{n}$ is denoted n_k.

Let $\mathbf{n}$ be uncorrelated with zero mean and variance σ^2 so that

$$p(\mathbf{n}) = \prod_{k=1}^{N} p(n_k) = \prod_{k=1}^{N} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\{-\frac{n_k^2}{2\sigma^2}\} = (\frac{1}{\sqrt{2\pi\sigma^2}})^N \exp\{-\frac{|\mathbf{n}|^2}{2\sigma^2}\} \tag{3C.4}$$

where $|\mathbf{n}|^2 \triangleq \sum_{k=1}^{N} n_k^2$ denotes the *Euclidian weight* of $\mathbf{n}$. Then the likelihood of a given candidate sequence $\mathbf{s_i} \in \mathbf{S}$ is given by

$$p(\mathbf{r}|\mathbf{s_i}) = (2\pi\sigma^2)^{-N/2} \exp\{-\frac{|\mathbf{r} - \mathbf{s_i}|^2}{2\sigma^2}\}. \tag{3C.5}$$

This function is to be maximized over all $\mathbf{s_i} \in \mathbf{S}$. This is obviously equivalent to minimizing $|\mathbf{r} - \mathbf{s_i}|^2$ over all $\mathbf{s_i} \in \mathbf{S}$, i.e. to choosing the sequence $\mathbf{s_i} \in S$ that is closest to $\mathbf{r}$ in the Euclidian sense.

The simple outcome of this example depends crucially on noise being Gaussian and uncorrelated. The reader may wish to verify that maximum-likelihood detection no longer boils down to minimization of Euclidian distances if either of these properties does not hold (see also Problem 3.12).

Maximum-likelihood detection for continuous-time signals embedded in white Gaussian noise may be treated as a limiting case of the situation that was just discussed. Some regularity conditions must be imposed on the signals, but in practice these are always met. Not surprisingly, the result is that we must again select the candidate signal that is closest to the received one in the Euclidian sense. This is exactly the criterion on which the receiver of Fig. 3.13 is based.

APPENDIX 3D

Performance of the maximum-likelihood sequence detector in the presence of colored Gaussian noise

In our derivation of the MLSD we have assumed that noise at the detector input is white. In practice this assumption may not hold true, and the MLSD will no longer

be optimum.[7] Given the fact that it was designed for white noise, one might expect the MLSD to be equally sensitive to noise at all frequencies. In this appendix we will see that this is, in general, not true. Specifically, MLSD performance is determined by a weighted average of the noise PSD $\mathcal{U}(e^{j2\pi\Omega})$ at the detector input. In the usual event that single bit-errors prevail, the weighing function is $|G(e^{j2\pi\Omega})|^2$ where $G(e^{j2\pi\Omega})$ is the discrete-time Fourier transform of the target response g_k for which the MLSD is meant. Thus the MLSD is highly immune to noise at frequencies where $|G|$ is small, and highly sensitive at frequencies where $|G|$ is large.

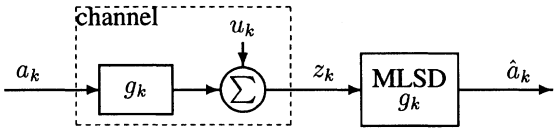


Figure 3D.1: Maximum-likelihood sequence detector (MLSD) operating on a sequence z_k with colored noise u_k. The detector is matched to the sytem response g_k.

The discrete-time channel of Fig. 3D.1 represents the cascade of the actual continuous-time channel, a prefilter and a synchronous sampling operation. The detector has input $z_k = (a * g)_k + u_k$ where a_k is the transmitted data sequence, g_k is the system response and u_k is the noise component of the detector input. Noise coloration may be due to both the continuous-time channel and the prefilter.

Let $\underline{a}^1$ and $\underline{a}^2$ be two admissible data sequences. Then the difference $\underline{e} \triangleq (\underline{a}^1 - \underline{a}^2)/2$ is called an *admissible error sequence*. Denote by $E(e^{j2\pi\Omega})$ the discrete-time Fourier transform of $\underline{e}$. Let S be the set of all admissible error sequences. For every $\underline{e} \in S$ we define a predetection signal-to-noise ratio

$$\rho(\underline{e}) \triangleq \frac{[\int_{-0.5}^{0.5} |E|^2 |G|^2 d\Omega]^2}{\int_{-0.5}^{0.5} |E|^2 |G|^2 \mathcal{U} d\Omega}. \tag{3D.1}$$

(For notational convenience we have suppressed the dependence of E, G and $\mathcal{U}$ on Ω.)

Theorem 3.2 *At sufficiently high channel SNRs, the effective predetection signal-to-noise ratio ρ_{MLSD} achieved by the MLSD of Fig. 3D.1 amounts to*

$$\rho_{MLSD} = \min_{\underline{e} \in S} \rho(\underline{e}). \tag{3D.2}$$

Proof: Conceptually, the MLSD first filters all admissible data sequences $\underline{a}^i$ by a filter with impulse response g_k (the index i serves to enumerates these sequences). The differences $z_k - (a^i * g)_k = ((a - a^i) * g)_k + u_k$ are squared and integrated. Out of the resulting path metrics

$$\lambda^i = \sum_k (((a - a^i) * g)_k + u_k)^2, \tag{3D.3}$$

[7]The designation 'MLSD' is then, strictly speaking, no longer warranted, but for convenience we will nevertheless stick to it.

the smallest one is selected, and the corresponding data sequence $\underline{\hat{a}}$ serves as the output of the VD. This is, in general, not a maximum-likelihood decision because noise u_k may be colored rather than white. Ideally the detector settles on the actually transmitted data sequence (i.e. $\underline{\hat{a}} = \underline{a}$), but in practice an erroneous sequence $\underline{\hat{a}}$ may be detected. Without loss of generality we restrict attention to the event of a first error occurring at instant $k = 0$. Then the error sequence $\underline{e} \triangleq (\underline{a} - \underline{\hat{a}})/2$ starts at $k = 0$ (i.e. $e_k = 0$ for $k < 0$ and $e_0 \neq 0$). For any first-error event starting at $k = 0$ and characterized by a given error sequence $\underline{e} \in S$ to occur, the following three conditions must be met simultaneously:

$\mathcal{C}_1$: No error event was in progress just before $k = 0$.

$\mathcal{C}_2$: The actual data sequence $\underline{a}$ must be such that $\underline{\hat{a}} \triangleq \underline{a} + 2\underline{e}$ is an allowable data sequence.

$\mathcal{C}_3$: A bad span of noise u_k causes $\underline{\hat{a}}$ to be more likely than $\underline{a}$.

The probability of $\mathcal{C}_1$ not being true is in the order of the symbol-error rate, so that $\Pr(\mathcal{C}_1) \simeq 1$ under normal operating conditions. The probability of $\mathcal{C}_2$ is typically also in the order of unity (see Appendix 7B for an analysis) and is therefore insignificant to the predetection SNR at realistic bit error rate levels. For $\mathcal{C}_3$ to hold, the path metric $\hat{\lambda}$ of $\underline{\hat{a}}$ should be smaller than that of $\underline{a}$, i.e.

$$\hat{\lambda} = \sum_k [((a - \hat{a}) * g)_k + u_k]^2 < \lambda = \sum_k u_k^2, \tag{3D.4}$$

where we have used (3D.3). Upon on realizing that $a_k - \hat{a}_k = 2e_k$ we obtain after some elementary rearrangements the equivalent condition

$$\sum_k [(e * g)_k^2 + u_k (e * g)_k] < 0. \tag{3D.5}$$

Due to its linear dependence on the Gaussian noise u_k, the left hand side of (3D.5) is a Gaussian stochastic variable. It has mean

$$\mathcal{M} = \sum_k (e * g)_k^2 = \int_{-0.5}^{0.5} |E(e^{j2\pi\Omega})|^2 |G(e^{j2\pi\Omega})|^2 d\Omega, \tag{3D.6}$$

and by analogy to Problem 1.18 its variance $\mathcal{V}$ amounts to

$$\mathcal{V} = \int_{-0.5}^{0.5} |E(e^{j2\pi\Omega})|^2 |G(e^{j2\pi\Omega})|^2 \mathcal{U}(e^{j2\pi\Omega}) d\Omega. \tag{3D.7}$$

It follows that

$$\Pr(\mathcal{C}_3) = Q(\sqrt{\frac{\mathcal{M}^2}{\mathcal{V}}}) = Q(\sqrt{\rho(\underline{e})}) \tag{3D.8}$$

where $Q(.)$ is the Q-function of Section 3.9. Based on this probability we can derive lower and upper bounds on the probability $\Pr[E]$ of a first-error event occurring at an arbitrary time instant k_1. For the sake of brevity we refer the reader to Appendix 7B for details of the involved reasoning. Both bounds become tight at high signal-to-noise ratios and are governed by the minimum of $\rho(\underline{e})$ over all admissible error sequences, i.e. by ρ_{MLSD}. □

Exercise 3.7 Assume that $G(e^{j2\pi\Omega})$ vanishes at DC. Argue that the MLSD is completely insensitive to DC offsets.

Specialization to single bit-errors In many instances ρ_{MLSD} will be governed by single bit-errors, for which $|E(e^{j2\pi\Omega})| = 1$. Then

$$\rho_{\text{MLSD}} = \frac{[\int_{-0.5}^{0.5} |G|^2 d\Omega]^2}{\int_{-0.5}^{0.5} |G|^2 \mathcal{U} d\Omega}. \tag{3D.9}$$

In the particular event that $|G(e^{j2\pi\Omega})|^2 = \text{FSN}(e^{j2\pi\Omega})$ and $\mathcal{U}(e^{j2\pi\Omega}) = 1$ as in Section 3.11 this becomes

$$\rho_{\text{MLSD}} = \int_{-0.5}^{0.5} \text{FSN}(e^{j2\pi\Omega}) d\Omega = \text{SNR} = \rho_{\text{MFB}}. \tag{3D.10}$$

This confirms that the MLSD of Section 3.11 will achieve the matched filter bound if single bit-errors prevail.

PROBLEMS

3.1 Consider L-PAM signaling for even values of L. Prove that the data variance σ_a^2 of a_k obeys $\sigma_a^2 = (L^2 - 1)/3$.

3.2 Prove (3.1).

3.3 Consider NRZ signaling with binary data $a_k \in \{-1, 1\}$. Determine the transmit power P_s via a time-domain argument. Use (3.1) to verify the result for the situation that data is uncorrelated ($\mathcal{A}(e^{j2\pi\Omega}) = 1$).

3.4 Show that the signal-to-noise ratio SNR_r of (3.11) may be viewed as a weighted average of $\text{FSN}(e^{j2\pi\Omega})$ according to

$$\text{SNR}_r = \int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega}) \text{FSN}(e^{j2\pi\Omega}) d\Omega. \tag{3P.1}$$

3.5 Assume that data a_k is uncorrelated with unit variance and that noise is white with power spectral density N_0. Prove that $\text{SNR}_r = \text{SNR} = P_d/P_n$ where P_d is the power of the data component of the received signal $r(t)$ and P_n is the noise power in a band of width $1/T$.

3.6 Voice-grade telephone lines have a one-sided bandwidth $W/2$ in the order of 3 kHz and an SNR in the order of 30 dB. Assume that the channel transfer characteristic is that of an ideal low-pass filter and that noise is white with power spectral density N_0, so that SNR $= P_s/(N_0 W)$ where P_s denotes the transmit power. Use (3.15) to estimate the capacity of this channel. Compare this with state-of-the-art voiceband modems, which use coded modulation and advanced reception techniques to achieve a bit rate of 14.4 kb/s.

3.7 In magnetic recording, information is stored along tracks of some width w. For simplicity we assume that the only disturbance of the recording channel is additive noise that is due to the medium, and we denote by SNR the signal-to-noise ratio of the channel.

1. Assume that the track is split into two halves of width $w/2$, that separate information is written on and read from each half-track, and that the noise components of both half-tracks are uncorrelated. Express the signal-to-noise ratio SNR$'$ for each half-track in terms of SNR.
2. Use (3.14) to compare the capacity $\mathcal{C}$ of the single-track recorder with the compound capacity $\mathcal{C}'$ for two half-tracks. To what extent is (3.14) applicable?
3. How far can capacity be increased relative to $\mathcal{C}$ by subdividing the single track into an arbitrary number of subtracks?

3.8 In this problem we are concerned with the minimum signal-to-noise ratio at which reliable signaling is possible for wideband channels. Consider the Hartley-Shannon capacity limit (3.15). The transmit power P_s is used to convey a total of $\mathcal{C}$ bits per second and therefore an average energy $E_b \triangleq P_s/\mathcal{C}$ is transmitted per bit.

1. Express $\mathcal{C}$ in terms of E_b/N_0. Divide this expression by W in order to obtain an implicit relation between $\mathcal{C}/W$ and E_b/N_0.
2. Assume that $\mathcal{C}/W$ is sufficiently small so that $\log_2(1 + \frac{\mathcal{C}}{W}\frac{E_b}{N_0}) \simeq \frac{1}{\ln 2}\frac{\mathcal{C}}{W}\frac{E_b}{N_0}$. Which signal-to-noise ratio E_b/N_0 is consistent with this approximation? How should this value be interpreted?

3.9 Extend the derivation of Section 3.9 to an L-AM data alphabet (L even) with equally likely symbols. Show in particular that ρ_{MFB} does not change and that the symbol error rate is given by

$$\Pr[E] = 2(1 - \frac{1}{L})Q(\sqrt{\rho_{\text{MFB}}}). \tag{3P.2}$$

How does the transmit power that is needed for a prescribed error rate depend on L at high SNRs?

3.10 Consider a channel with transfer function

$$H(\Omega) = \begin{cases} \sqrt{T}\cos \pi\Omega & \text{for } |\Omega| < \frac{1}{2} \text{ and} \\ 0 & \text{for } |\Omega| \geq \frac{1}{2}, \end{cases} \tag{3P.3}$$

and white additive Gaussian noise with power spectral density N_0. This channel has no excess bandwidth and is said to be of the *duobinary* type.

1. Compute the folded signal-to-noise ratio FSN$(e^{j2\pi\Omega})$ for this channel. Can you think of a very simple discrete-time channel with white noise and the same folded signal-to-noise ratio?
2. Compute the matched-filter bound.

3. Determine the equivalent discrete-time channels of Sections 3.10.1 and 3.10.2.

3.11 Rederive the matched-filter bound on the basis of the equivalent channel model of Fig. 3.22.

3.12 Consider the discrete-time channel model of Fig. 3.22 for the situation that noise n_k is white and has probability density function (PDF)

$$p_\nu(x) \triangleq \frac{\nu}{2\sqrt{2}\sigma\Gamma(1/\nu)} \exp[-|\frac{x}{\sqrt{2}\sigma}|^\nu] \tag{3P.4}$$

where ν is a non-negative constant, $\sigma = \sqrt{\int_{-\infty}^{\infty} x^2 p_\nu(x)dx}$ is the standard deviation and $\Gamma(.)$ is the gamma function. This PDF is from the *generalized exponential class* [15] and for $\nu = 2$ degenerates into a standard Gaussian PDF with mean 0 and variance σ^2. Generalize the maximum-likelihood receiver of Fig. 3.23 to this noise PDF. Hint: retrace the steps of Appendix 3C.

3.13 Consider the equivalent channel of Fig. 3.22. Show that

$$\text{SNR} = \frac{E_\gamma}{\sigma_n^2} \tag{3P.5}$$

where $E_\gamma \triangleq \sum_{k=-\infty}^{\infty} \gamma_k^2$ is the energy of γ_k and $\sigma_n^2 = 1$ is the variance of the noise n_k. Prove that (3P.5) remains valid when both γ_k and n_k are scaled by an arbitrary factor α.

3.14 Consider uncoded binary signaling across an equivalent discrete-time channel with impulse response $(h_0, h_1, h_2) = (0.5, \frac{1}{\sqrt{2}}, 0.5)$ of memory length $M = 2$ and white additive Gaussian noise n_k of unit power.

1. Sketch the amplitude characteristics of the channel. Are there any spectral zeros? If so, where are they located and what is their order?
2. Determine the worst error sequence $\underline{e}$, the predetection signal-to-noise ratio ρ_{ML}, and the loss $\rho_{\text{MFB}}/\rho_{\text{ML}}$ with respect to the matched-filter bound. Compare this loss with the 1.77 dB of Example 3.7.

3.15 Redo Problem 3.14 for the responses $(h_0, \ldots, h_3) = (0.38, 0.60, 0.60, 0.38)$ and $(h_0, \ldots, h_4) = (0.29, 0.50, 0.58, 0.50, 0.29)$ of memory length $M = 3$ and $M = 4$, respectively.

3.16 Consider L-PAM signaling, i.e. uncoded transmission using the L-AM alphabet. Determine the alphabet of the admissible error sequences. Are all error sequences within this alphabet admissible?

3.17 Consider a binary modulation code that produces sequences $\underline{a}$ of even length N. Assume that the code is *DC-free*, i.e. that valid sequences $\underline{a}$ all have $\sum_{k=0}^{N-1} a_k = 0$. Show that the maximum-likelihood receiver of Section 3.11 cannot make single bit-errors.
Hint: Prove that the error sequence $\underline{e}$ that corresponds to the single bit-error is not admissible.

3.18 Assume that the equivalent discrete-time channel of Fig. 3.22 is of the *Class IV partial-response* type, i.e. that $\gamma(\mathcal{D}) = 1 - \mathcal{D}^2$ (see Chapter 6 for an explanation of this terminology). Use a similar reasoning as in Example 3.10 to find the minimum-weight error sequence and the predetection signal-to-noise ratio. Is the matched-filter bound attainable? If not, then what is the loss incurred by a maximum-likelihood receiver? Use a picture like Fig. 3.25 to explain

these results. Answer the same questions for the *extended Class IV partial-response* channel, which has $\gamma(\mathcal{D}) = 1 + \mathcal{D} - \mathcal{D}^2 - \mathcal{D}^3$.

3.19 Show that ρ_{ML} may be determined directly from the spectral-to-noise ratio $\mathrm{SN}(\Omega)$ of the hybrid channel according to

$$\rho_{\mathrm{ML}} = \min_{\underline{e} \in S} \int_{-\infty}^{\infty} |E(e^{j2\pi\Omega})|^2 \mathrm{SN}(\Omega) d\Omega. \tag{3P.6}$$

3.20 Consider the Lorentzian channel of Appendix 3A with NRZ as modulation code. For what normalized information densities D are the double bit-errors with $e(\mathcal{D}) = 1 - \mathcal{D}$ more likely than single bit-errors? Quantify the corresponding loss with respect to the matched-filter bound and compare it with Fig. 3.26.a. Hint: Use (3A.9), (3P.6), and the integral identity

$$\int_0^{\infty} e^{-ax} \sin^4 bx dx = \frac{3}{8a} \frac{1}{(\frac{a}{4b})^2 + 1} \frac{1}{(\frac{a}{2b})^2 + 1}. \tag{3P.7}$$

REFERENCES

[1] K. Abend and B.D. Fritchmann, 'Statistical Detection for Communication Channels with Intersymbol Interference', *Proc. IEEE*, Vol. 58, pp. 779-785, May 1970.

[2] M. Abramowitz and I. Stegun, *Handbook of Mathematical Functions*. New York: Dover Publications, 1972.

[3] J.W.M. Bergmans, 'Discrete-Time Models for Digital Magnetic Recording', *Philips J. Res.*, Vol. 41, pp. 531-558, 1986.

[4] J.A.C. Bingham, 'Multicarrier Modulation for Data Transmission: An Idea Whose Time Has Come', *IEEE Communications Magazine*, pp. 5-14, May 1990.

[5] R.W. Chang and J.C. Hancock, 'On Receiver Structures for Channels Having Memory', *IEEE Trans. Inform. Theory*, Vol. IT-12, pp. 463-468, Oct. 1966.

[6] J.M. Cioffi, G.P. Dudevoir, M.V. Eyuboğlu and G.D. Forney, Jr., 'MMSE Decision-Feedback Equalization and Coding - Part II: Coding Results', *IEEE Trans. Commun.*, Vol. 43, No. 10, pp. 2595-2604, Oct. 1995.

[7] T.M. Cover and J.A. Thomas, *Elements of Information Theory*. New York: John Wiley, 1991.

[8] G.D. Forney, Jr., 'Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference', *IEEE Trans. Inform. Theory*, Vol. IT-18, pp. 363-378, May 1972.

[9] G.D. Forney, Jr., R.G. Gallager, G.R. Lang, F.M. Longstaff and S.U.H. Qureshi, 'Efficient Modulation for Band-Limited Channels', *IEEE J. Selected Areas Commun.*, Vol. SAC-2, No. 5, pp. 632-647, Sept. 1984.

[10] G.D. Forney, Jr., 'Coset Codes - Part I: Introduction and Geometrical Classification', *IEEE Trans. Inform. Theory*, Vol. IT-34, pp. 1123-1151, Sept. 1988.

[11] G.D. Forney, Jr., and M.V. Eyuboğlu, 'Combined Equalization and Coding Using Precoding', *IEEE Communications Magazine*, pp. 25-34, Dec. 1991.

[12] R.G. Gallager, *Information Theory and Reliable Communication.* New York: John Wiley & Sons, 1968.

[13] R.V.L. Hartley, 'Transmission of Information', *Bell Syst. Tech. J.*, Vol. 7, No. 3, pp. 535-563, July 1928.

[14] C. Heegard and L. Ozarow, 'Bounding the Capacity of Saturation Recording: The Lorentz Model and Applications', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 145-156, Jan. 1992.

[15] M.C. Jeruchim, 'Techniques for Estimating the Bit Error Rate in the Simulation of Digital Communication Systems', *IEEE J. Selected Areas Commun.*, Vol. SAC-2, No. 1, pp. 153-170, Jan. 1984.

[16] H. Kobayashi, 'Application of Probabilistic Decoding to Digital Magnetic Recording Systems', *IBM J. Res. Develop.*, Vol. 15, pp. 64-74, Jan. 1971.

[17] E.A. Lee and D.G. Messerschmitt, *Digital Communication.* Boston: Kluwer Academic Publishers, 1988.

[18] F.R. Magee, Jr., and J.G. Proakis, 'An Estimate of the Upper Bound on Error Probability for Maximum-Likelihood Sequence Estimation on Channels Having a Finite-Duration Pulse Response', *IEEE Trans. Commun.*, Vol. COM-21, pp. 699-702, Sept. 1973.

[19] J.L. Massey, 'The How and Why of Channel Coding', *Proc. 1984 International Zürich Seminar on Digital Communications*, pp. 67-73, Zürich, Switzerland, March 6-8, 1984.

[20] H. Meyr, M. Oerder, and A. Polydoros, 'On Sampling Rate, Analog Prefiltering, and Sufficient Statistics for Digital Receivers', *IEEE Trans. Commun.*, Vol. COM-42, No. 12, pp. 3208-3214, Dec. 1994.

[21] D.O. North, 'Analysis of the Factors which Determine Signal/Noise Discrimination in Radar', RCA Laboratories, Princeton, N.J., Rept. PTR-6C, June 1943.

[22] J.K. Omura, 'On Optimum Receivers for Channels with Intersymbol Interference', (Abstract), presented at the IEEE Int. Symp. Information Theory, Noordwijk, Holland, June 1970.

[23] R. Price, 'Nonlinearly Feedback-Equalized PAM VS. Capacity, for Noisy Filter Channels', in *Rec. Int. Conf. Commun.*, ICC-72, pp. 22.12-22.17.

[24] J.G. Proakis, *Digital Communications.* New York: McGraw-Hill, 1983 (2nd ed. 1989).

[25] C.E. Shannon, 'A Mathematical Theory of Communication', *Bell Syst. Tech. J.*, Vol. 27, pp. 379-423, 623-656, Oct. 1948.

[26] C.E. Shannon, 'Communication in the Presence of Noise', *Proc. IRE*, Vol. 37, No. 1, pp. 10-21, Jan. 1949.

[27] G. Ungerboeck, 'Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data-Transmission Systems', *IEEE Trans. Commun.*, Vol. COM-22, No. 5, pp. 624-636, May 1974.

[28] G. Ungerboeck, 'Channel Coding with Multilevel/Phase Signals', *IEEE Trans. Inform. Theory*, Vol. IT-28, pp. 55-67, Jan. 1982.

[29] G. Ungerboeck, 'Trellis-Coded Modulation with Redundant Signal Sets - Part I: Introduction', *IEEE Communications Magazine*, Vol. 25, No. 2, pp. 5-11, Feb. 1987.

[30] G. Ungerboeck, 'Trellis-Coded Modulation with Redundant Signal Sets - Part II: State of the Art', *IEEE Communications Magazine*, Vol. 25, No. 2, pp. 12-21, Feb. 1987.

[31] A.J. Viterbi, 'Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm', *IEEE Trans. Inform. Theory*, Vol. IT-13, pp. 260-269, Apr. 1967.

[32] J. van Vleck and D. Middleton, 'A Theoretical Comparison of the Visual, Aural and Meter Reception of Pulsed Signals in the Presence of Noise', *J. Appl. Phys.*, Vol. 17, pp. 940-971, Nov. 1946.

[33] J.M. Wozencraft and I.M. Jacobs, *Principles of Communication Engineering*. New York: Wiley, 1965.

4

BASEBAND MODULATION CODES

4.1 INTRODUCTION

Baseband modulation codes are widely applied in such diverse fields as digital line transmission [21, 28, 18, 116], digital optical transmission [111, 16], and digital magnetic and optical storage [105, 100, 106]. They act to translate the source data sequence d_n into a sequence a_k that is transmitted across the channel (Fig. 4.1; see also Chapter 3). The principal goal is to enable the receiver to produce reliable decisions $\hat{d}_n$ about d_n. Code design should, therefore, account for the characteristics of both channel and receiver. (Because modulation coding is the only type of coding that we consider in this chapter, we shall usually say 'coding' where we mean 'modulation coding'.)

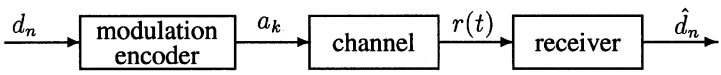


Figure 4.1: Transmission system with modulation encoder, hybrid channel and receiver.

Example 4.1 If the channel has restricted bandwidth then a multilevel modulation code may be used to minimize the bandwidth of the transmitted signal (see Section 4.4.3). Codes of this type are commonly used in cable transmission, and may help to increase the data throughput towards the capacity of the channel. The inevitable cost is a larger complexity than for an uncoded system, particularly for the receiver.

Example 4.2 If the receiver has restricted provisions for dealing with intersymbol interference (ISI), then it may be desirable to mitigate ISI via a code that suppresses those frequencies at which the channel has a poor transfer (see Section 4.3). Codes of this type have long been used in digital recording systems. They perform spectral shaping by adding redundancy to the source data symbols. This overhead, in turn, normally represents a step away from channel capacity.

These two examples show that code design can proceed in rather opposite directions. Design goals are manifold and the following ones are often mentioned [28, 29, 18, 99].

Reduction of intersymbol interference. Many channels have a poor or ill-predictable transfer in one or more frequency ranges. If the receiver cannot handle the ensuing ISI then it is desirable that the modulation encoder suppresses the unreliable frequencies. As a result, ISI will decrease and the system will be more tolerant to channel parameter variations (see Section 4.3).

Provision of timing information. It is common practice to derive the timing information from the received signal. The encoded data sequence should contain enough timing information to make this possible (see Section 4.5).

Transparency. The statistics of a_k should be well-defined irrespective of those of d_n. This facilitates the proper operation of the receiver.

Provisions for error monitoring. It is desirable that the modulation code facilitates the detection and monitoring of errors during system operation.

Provision of word alignment information. If data is encoded on a block-by-block basis then the received symbol sequence must be partitioned correctly into such blocks before decoding. This can be accomplished, for example, by looking for words that are prohibited at the proper word alignment, or by detection of predefined synchronization markers that are embedded into the data stream. Word synchronization is beyond the scope of this text (see [109], [79], [18], [14, ch. 10] and [101] for an introduction).

Addition of supplementary signals. The creation of a zero at DC (see Section 4.9) allows supplementary low-frequency signals to be conveyed to the receiver. Applications include power feeding across metallic lines [28], combined transmission of digital data and analog speech signals across local telephone lines [115], and servo signaling in digital optical recording [88].

Efficiency. The code should make optimum use of the channel, i.e. redundancy should be minimal, Euclidian distances should be high and the data rate should be close to capacity.

Cost. Any coding scheme is merely a part of a larger system and its cost must be in proportion to its importance within that system.

Novelty. An old code cannot be patented and new one may have commercial value even if its technical merits are questionable.

The latter two aims are non-technical but are often of major importance. As to the technical aims, much emphasis has historically been put on code properties that facilitate reception, such as reduction of ISI, enhancement of clock information, and ease of performance monitoring. Advances in data equalization, detection, timing recovery and adaptation, as described in Chapters 5 to 8, have paved the way for powerful receivers that require little if any 'reception-easing' overhead in the code. Continuous advances in VLSI technology, moreover, are making such receivers increasingly practical. This explains the current trend towards codes with small redundancy.

Example 4.3 Hard disk drives have long used run-length-limited modulation codes with redundancies in the order of 50 to 100%. These codes permit data reception with a minimum

of complexity (see Section 10.3.7). Modern high-end disk drives, by comparison, push net information densities up by around 30% by using codes with much less redundancy along with complicated receivers that incorporate such features as adaptive equalization, data-aided timing recovery and Viterbi detection [25].

The design of codes is often approached via simple mathematical yardsticks, such as the digital sum variation and run-length limits (see Sections 4.8 and 4.9 for an explanation of these terms). The practical relevance of these yardsticks depends heavily on the characteristics of channel and receiver. Their main advantage is that they turn code design into a well-defined mathematical problem. This problem has attracted many talented scientists over the past decades, and an abundance of codes and accompanying theories has evolved. It would be unfair to suggest that it is merely the academic beauty of the code design problem that has spurred all this activity. More significantly, practical transmission and storage systems must often adhere to governmental, industrial or *de facto* standards. Modulation codes are a basic ingredient of such standards and as such have great commercial value.

In our discussion of modulation codes we will restrict attention to some of the more relevant technical aspects and some of the more popular codes. For the sake of brevity we do not delve deeply into code design and analysis techniques. These techniques have reached a high level of sophistication, and several excellent expositions are available (see, for example, [3, 78, 61, 100, 55, 19, 46]). Section 4.2 introduces various notions that are related to ISI. These notions are used in Section 4.3 to elaborate the effect of coding on ISI. Section 4.4 assesses the effect of coding on per-symbol signal-to-noise ratios. This effect is often under-stressed, and yields valuable insights into attainable performances. In Section 4.5 we shall see how coding can be used to increase the timing content of the received signal and to improve code transparency. Common approaches for code mechanization are outlined in Section 4.7. Section 4.8 provides an overview of run-length-limited codes. This overview covers fundamentals as well as a small catalog of worked examples. A similar overview is given in Sections 4.9-4.11 for codes that introduce one or more spectral zeros. Both linear and nonlinear codes of this type are treated. The distinction between these two types is not often made but has important consequences for receiver design and transmission performance.

4.2 INTERSYMBOL INTERFERENCE, NYQUIST-1 FUNCTIONS AND EYE PATTERNS

Fig. 4.2 shows the system model that will take center stage throughout this chapter. Since most building blocks were already described in Section 3.2 we summarize their function only briefly.

A modulation encoder translates the source bit sequence d_n of rate $1/T_s$ into a data sequence a_k of rate $1/T$ that is transmitted via a linear pulse modulator with symbol response $c(t)$ (see Section 1.2.5 for a description of linear pulse modulation). The received signal $r(t)$ is a noisy and filtered version of the transmitted signal $s(t)$. The sym-

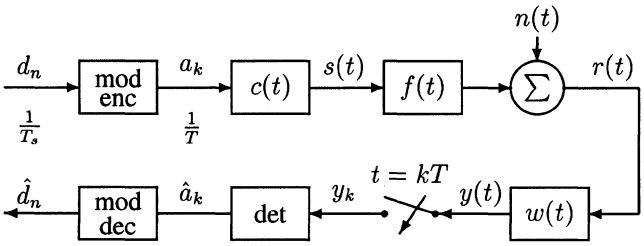


Figure 4.2: Transmission system with modulation encoder, hybrid channel, equalizer, sampler and detector.

bol response $h(t) \triangleq (c*f)(t)$ up until the channel typically has restricted bandwidth so that $r(t)$ depends on any given instant t on two or more transmitted data symbols. This effect is called *intersymbol interference* (ISI). The filter with impulse response $w(t)$ serves to handle ISI and suppress noise $n(t)$. These requirements conflict and therefore the output $y(t)$ of the filter will exhibit residual ISI and/or noise. This output is sampled in synchronism with the transmitted data. A detector operates on the samples y_k to produce symbol decisions $\hat{a}_k$ with respect to a_k. These are converted into bit decisions $\hat{d}_n$ by means of a modulation decoder.

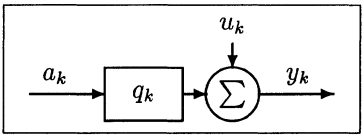


Figure 4.3: Equivalent discrete-time model for the cascade of modulator, channel, filter and sampler.

For future reference we note that the cascade of modulator, channel, filter and sampler may be regarded as a discrete-time channel as in Fig. 4.3. The impulse response q_k is a sampled version of the *system response* $q(t) \triangleq (h * w)(t) = (c * f * w)(t)$ and is referred to as the *sampled system response*. For the sampling moments $t = kT$ that are assumed in Fig. 4.2 we have $q_k = q(kT)$.[1] The noise component u_k is a filtered and sampled version of $n(t)$.

Example 4.4 The non-return-to-zero (NRZ) modulation code uses binary data $a_k \in \{-1, 1\}$, while $c(t)$ is a hold function of width T so that $s(t)$ is a binary waveform that directly reflects the underlying data sequence a_k (Fig. 4.4, top). Let us consider an idealized optical recording channel as in Section 2.4.3 This channel has a Gaussian low-pass characteristic $F(\Omega) = \exp\{-(\pi S\Omega/2)^2\}$. (We recall that Ω is a normalized measure of frequency, with $\Omega = 1$ corresponding to the symbol rate $1/T$.) For normalized information densities S of practical interest, the symbol response $h(t) = (c*f)(t)$ extends over several symbol intervals. This causes neighboring pulses to interfere with each other. In the middle portion of Fig. 4.4 (which pertains to

[1]We use the same symbol q for q_k and $q(t)$ in order to stress their close relationship. Distinction between them can be based on the argument. A similar remark applies to the signals $y(t)$ and y_k.

$S = 1.5$) this ISI is visible in that the channel output $r(t)$ is no longer binary at the instants $t = kT$. The amplitude of the replay signal is smallest for data patterns with the highest possible frequency, i.e. for alternating patterns.

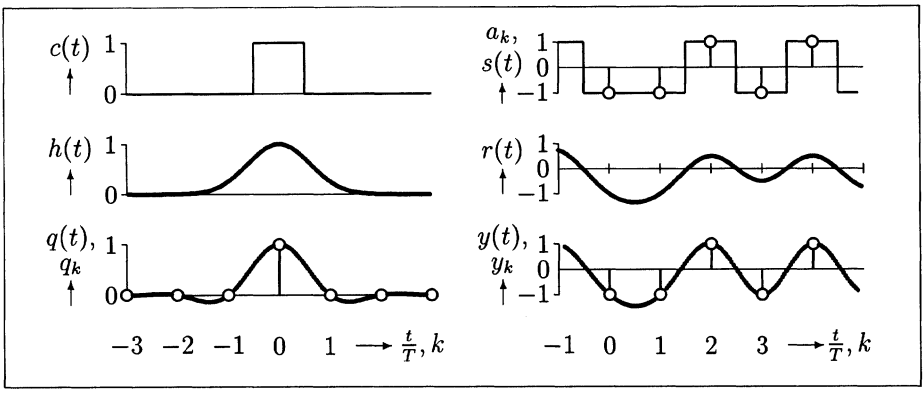


Figure 4.4: Example symbol responses and signal waveforms for the system of Fig. 4.2 in the absence of noise $n(t)$.

A classical technique to undo ISI is *full-response linear equalization*. Here $w(t)$ is dimensioned in order for the sampled system response q_k to degenerate into a Kronecker delta function ($q_k = \delta_k$). This requires a system response $q(t)$ for which

$$q(kT) = \begin{cases} 1, & k = 0, \\ 0, & \text{else.} \end{cases} \tag{4.1}$$

A function of this type is said to be a *Nyquist-1 function* and yields a detector input $y_k = (a * q)_k = a_k$ without ISI (Fig. 4.4, below). Bit decisions $\hat{a}_k$ can, therefore, be based on the polarity of y_k.

The effect of ISI is often depicted in the form of an *eye pattern* or *eye diagram*. Such a diagram appears when a data signal is displayed on an oscilloscope that is synchronized to the data clock. The effect of this synchronization is that the signal is cut into traces of one or more symbol intervals that are overlaid on the screen of the oscilloscope in a phase-aligned manner. The shape and size of the resulting 'eye' provide an indication of the margin of the system against various disturbances.

Example 4.5 Fig. 4.5 shows the eye patterns of the signals $r(t)$ and $y(t)$ of Fig. 4.4. The sampling instants $t = kT$ are aligned with the centers of the patterns. In principle, the polarity of $r(t)$ in the center of the 'eye' provides an indication of the underlying bits a_k. However, there is not much margin against added noise because the eye is nearly closed. The *eye height* or *vertical eye opening* decreases further towards the edges of the pattern. The *eye width* or *horizontal eye opening*, i.e. the interval across which there is any vertical eye opening, is a comfortable 90 percents. This indicates that there is considerable margin against timing errors. The equalized

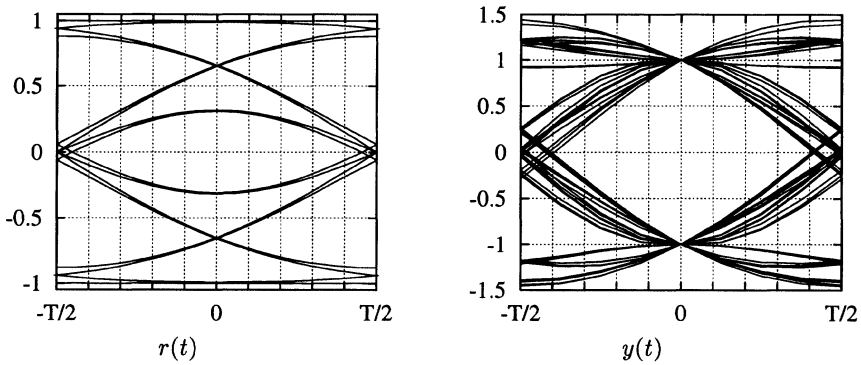


Figure 4.5: Eye patterns of the signals $r(t)$ and $y(t)$ of Fig. 4.4.

eye, by comparison, is fully open at the ideal sampling instants and there is still considerable eye width. It is worth mentioning that this width depends heavily on the bandwidth of the selected Nyquist-1 response. A larger eye-width necessitates more bandwidth and this normally causes noise to be enhanced. This tradeoff is discussed in detail in Chapter 5.

4.3 EFFECT OF MODULATION CODE ON INTERSYMBOL INTERFERENCE

Practical channels often have a poor or ill-predictable transfer in one or more frequency ranges. This poses several challenges to the receiver. We first illustrate this for two important effects, namely baseline wander and channel parameter variations. We then provide an analysis.

Example 4.6 The electronic circuits that form part of the transmission chain are often AC-coupled via capacitors. Furthermore, the transmitter and receiver may be AC-coupled with the channel via one or more transformers. In both cases the channel cannot pass DC, and the eye pattern will slowly wander up and down the screen of the oscilloscope. This *baseline wander* or *DC-wander* causes a low-frequency form of intersymbol interference (ISI) that may severely degrade performance. The movements of the eye pattern cannot be illustrated on paper, but a longer-term average can be depicted. This is done in Fig. 4.6 for the system of Example 4.4. It is assumed that the equalized system has the Nyquist-1 characteristic considered there, except for the lowest frequencies, which are suppressed due to a first-order high-pass filter with normalized cut-off frequency Ω_c.

Cut-off frequencies must be a small fraction of a percent of the signaling rate $1/T$ for baseline wander to be negligible. In Problem 4.1 it is shown that the RMS magnitude of the ISI is proportional to $\sqrt{\Omega_c}$ for small Ω_c. Thus the eye pattern approaches the limiting one of Fig. 4.5 only slowly as Ω_c is decreased.

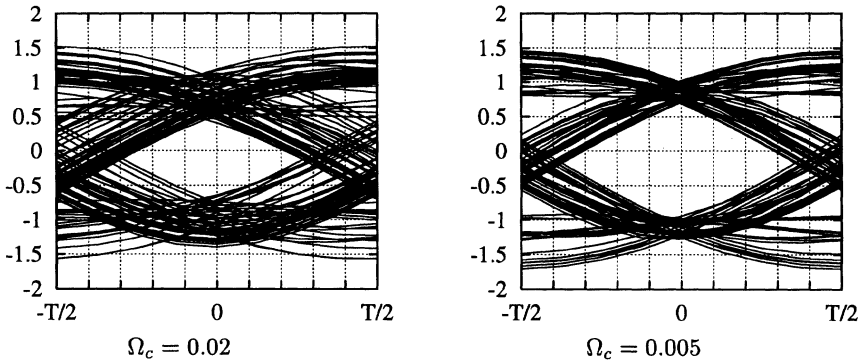


Figure 4.6: Eye patterns for a system that is Nyquist-1 except for baseline wander. The data sequence a_k is uncorrelated. The AC-coupling that causes baseline wander is due to a first-order high-pass filter with normalized cut-off frequency Ω_c.

Example 4.7 In digital magnetic recording systems, fluctuations of head-to-medium spacing cause the channel transfer function to become ill-defined, particularly at high frequencies. A similar effect occurs in digital optical recording systems as a result of dynamic defocusing errors. In both cases a fixed equalizer can compensate for imperfections of the *nominal* transfer function but cannot handle these fluctuations. As a result, the eye opening varies dynamically along with the fluctuations. This is illustrated in Fig. 4.7 for the system of Example 4.4. It is assumed that the equalizer is dimensioned for a normalized information density $S = 1.5$, but that the actual density can vary between $S = 1$ and $S = 2$.

Especially for $S = 2$ there is hardly any eye opening left. For the sake of completeness we mention that certain partial-response equalization techniques are less susceptible to these high-frequency fluctuations (see Chapter 6).

In principle, baseline wander and channel parameter variations can both be handled within the receiver by means of adaptive techniques for DC-restoration and equalization, respectively (see Chapter 8). These techniques may, however, be more complicated than one is willing to accept. An alternative way to go is to use a modulation code that de-emphasizes the problematic frequency bands. This causes the problematic features of the channel to be less visible in the received signal. As a result, less effort or no effort at all is required in the receiver to deal with them.

To make these notions more precise we must distinguish between the actual sampled response q_k of the system and the desired one which we denote as g_k. For full-response equalization we have $g_k = \delta_k$, while other *target responses* g_k are used in more powerful reception techniques (see Chapters 6 and 7). Deviations of q_k from g_k cause the actual detector input $(a * q)_k$ to deviate from the desired input $(a * g)_k$. The error $e_k \triangleq (a * q)_k - (a * g)_k$ is just the residual ISI that the detector faces at the decision instants (Fig. 4.8). Intuitively, we would expect the performance of the detector to degrade as the power of e_k increases. In forthcoming chapters we will see that, to a first-order approximation, this is indeed the case for receivers with symbol-by-symbol detection. Se-

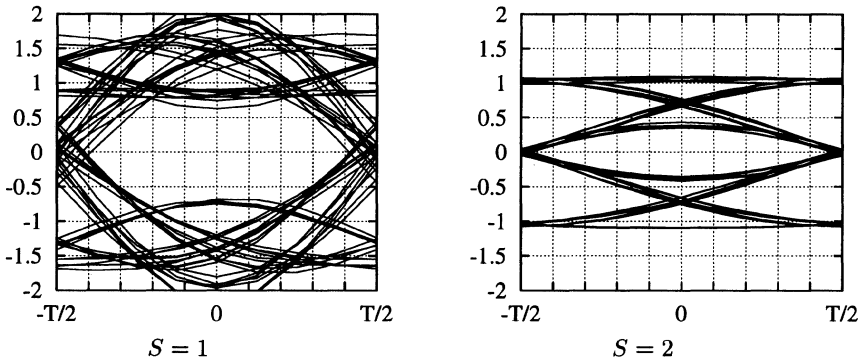


Figure 4.7: Eye patterns of the equalized signal $y(t)$ in the system of fig. 4.4 for the situation that the normalized information density S deviates from the nominal value $S = 1.5$. The eye pattern for $S = 1.5$ is depicted in Fig. 4.5.

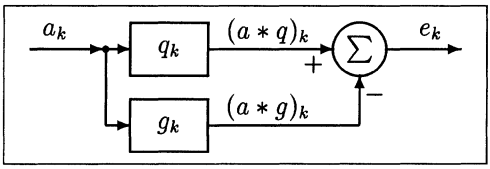


Figure 4.8: Residual intersymbol interference e_k arises when the sampled response q_k of the system deviates from the target response g_k.

quence detection induces a frequency-dependent sensitivity to disturbances (see, for example, Appendix 7C). To account for this sensitivity we can take a spectrally weighted version ε of the power of e_k as a crude measure of the performance degradation due to ISI. Let us denote the weighing function by $V(e^{j2\pi\Omega})$. With Fig. 4.8 we note that the power spectral density (PSD) of e_k is $\mathcal{E}(e^{j2\pi\Omega}) = \mathcal{A}(e^{j2\pi\Omega})|Q(e^{j2\pi\Omega}) - G(e^{j2\pi\Omega})|^2$ where $\mathcal{A}(e^{j2\pi\Omega})$ is the PSD of a_k, while $G(e^{j2\pi\Omega})$ and $Q(e^{j2\pi\Omega})$ denote the discrete-time Fourier transforms of g_k and q_k. Thus

$$\varepsilon = \int_{-0.5}^{0.5} \mathcal{E}V d\Omega = \int_{-0.5}^{0.5} \mathcal{A}V|Q - G|^2 d\Omega \tag{4.2}$$

where the dependencies of $\mathcal{E}$, V, $\mathcal{A}$, Q and G on Ω have been suppressed for notational convenience. In Examples 4.6 and 4.7 we have seen that Q may deviate significantly from G in specific frequency bands. To keep ε small in spite of these deviations we can either shape the data spectrum to have suppressed content in the 'unreliable' bands, or we can use a detector that has a reduced sensitivity to disturbances in those bands. The first measure is generally simple in terms of hardware but requires introduction of redundancy and is thereby wasteful of bandwidth. The second one is more complicated but requires no bandwidth expansion.

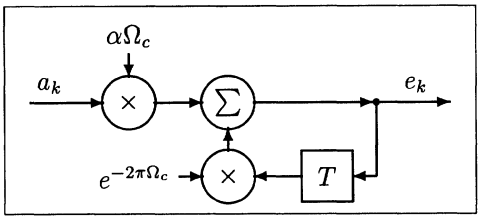


Figure 4.9: First-order discrete-time model of residual ISI due to baseline wander.

Example 4.8 Reduction of baseline wander is possible with the aid of a modulation code that suppresses frequencies near DC. The effect of such a code is quantified in Problem 4.1. Let us assume that baseline wander is due to a first-order high-pass filter with normalized cut-off frequency Ω_c. Appendix 4A shows that the resulting ISI e_k can be modeled as in Fig. 4.9. The data a_k is first scaled by a factor $\alpha\Omega_c$, where α is a constant that is typically in the order of 2π. It is then filtered by a first-order recursive low-pass filter with a time constant of $1/(2\pi\Omega_c)$ symbol intervals. Normally $\Omega_c \ll 1$ and therefore the filter approximately performs an integration of the scaled data sequence $\alpha\Omega_c a_k$. Correspondingly $e_k \simeq \alpha\Omega_c c_k$ where

$$c_k \triangleq \sum_{i \le k} a_k \tag{4.3}$$

is the *running digital sum* (RDS) or *charge* of a_k. Hence baseline wander can be suppressed by keeping the RDS within strict bounds. This is the distinguishing feature of *DC-free modulation codes*, which are discussed in Section 4.9. Fig. 4.10 shows resulting eye patterns for the same situation as in Fig. 4.6 and an RDS that is bounded between -3 and 3 (both inclusive). Baseline wander has clearly decreased, especially for $\Omega_c = 0.005$.

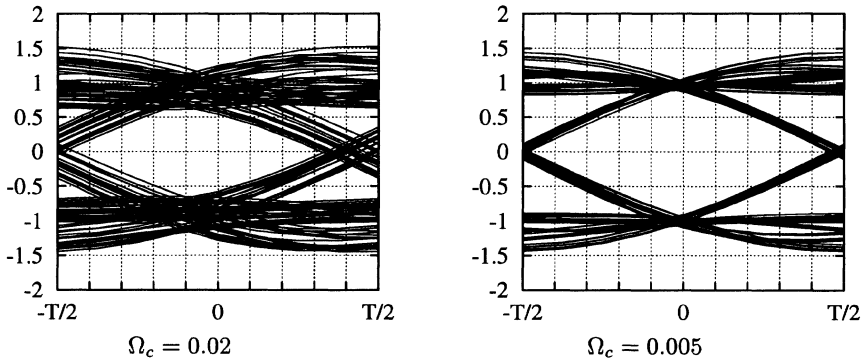


Figure 4.10: Eye patterns for the same system as in Fig. 4.6. Data a_k is constrained to have a running digital sum between -3 and $+3$ (both inclusive).

Example 4.9 In Chapter 6 we will discuss *partial-response techniques*. These are based on the use of simple integer-valued target responses g_k, such as $g_k = \delta_k - \delta_{k-1}$. For this particular response the sensitivity function $V(e^{j2\pi\Omega})$ of the maximum-likelihood sequence detector is found to be $V(e^{j2\pi\Omega}) = 4\sin^2 \pi\Omega$ (see Appendix 7C). This function has a second-order zero at DC. The implication is that the detector is fully insensitive to DC-offsets and highly insensitive to baseline wander.

Example 4.10 Run-length-limited (RLL) codes shape the data spectrum by constraining the minimum and maximum length of runs of identical symbols a_k. A typical spectrum is depicted in Fig. 4.11. It pertains to the so-called $(2,7)$ code (see Section 4.8). This code has rate $R = 1/2$, i.e. the baud rate $1/T$ is twice the source data rate $1/T_s$. Frequencies in Fig. 4.11 are normalized with respect to $1/T$.

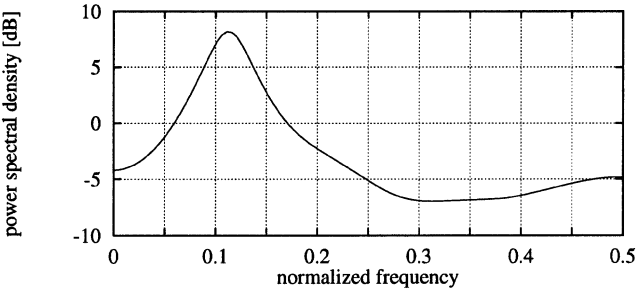


Figure 4.11: Power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of $(2,7)$ run-length-limited code.

The spectrum has a bandpass nature and this renders $(2,7)$ suited to systems with less reliable upper and lower band edges. We illustrate this for the system of Example 4.7. Fig. 4.12 depicts corresponding eye patterns at the output of the channel.

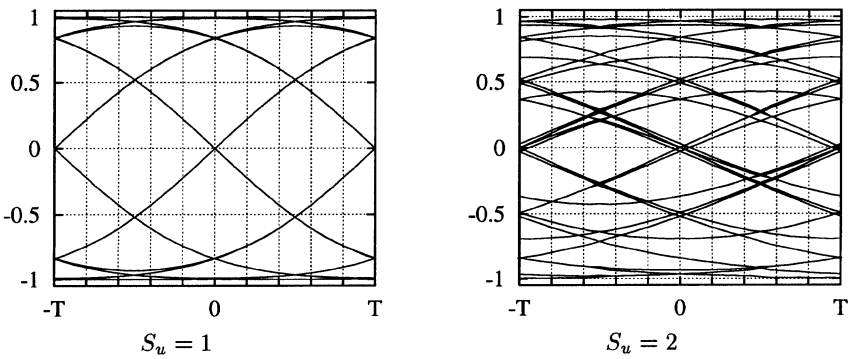


Figure 4.12: Eye patterns of the received signal $r(t)$ in the system of Fig. 4.4 for two different user densities $S_u = t_0/T_s$. The data sequence a_k is produced by a $(2,7)$ run-length-limited code with rate $R = 1/2$.

Because the symbol interval T has halved with respect to uncoded storage, two eyes now occupy the same time span as a single eye in Figs. 4.5 and 4.7. For the same reason normalized information densities $S = t_0/T$ have doubled, even though the underlying modulation transfer functions have not changed. It is, therefore, more natural to normalize densities with respect to the user bit length $T_s = T/R$, which does not depend on R. The eyes remain open across the considered density range even without equalization. This robustness comes at the cost of a decreased margin against noise, as evidenced in part by the small eye height (see Section 4.4 for an analysis of the effect of coding on signal-to noise ratios). The net effect of the code is somewhat akin to that of an equalizer and this explains the casual designation 'poor-man's equalizer' that is sometimes used for RLL codes. It is worth mentioning that the compact disc (CD) system uses an RLL code similar to $(2, 7)$, while the CD channel may be crudely described by the same model as considered here. The eye patterns of Fig. 4.12, therefore, are quite reminiscent of those in CD.

4.4 EFFECT OF MODULATION CODE ON SIGNAL-TO-NOISE RATIOS

In Section 4.3 we have established that shaping of the data spectrum $\mathcal{A}(e^{j2\pi\Omega})$ has various advantages but that only a flat spectrum ($\mathcal{A}(e^{j2\pi\Omega}) = 1$) comes for free in terms of redundancy. In all other cases we must increase the signaling rate $1/T$ relative to the situation without redundancy. This increment inevitably degrades per-symbol signal-to-noise ratios. We first review this degradation for the situation where noise $n(t)$ is white and a_k is binary.

4.4.1 Effect of coding on SNR

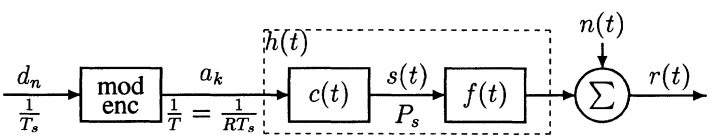


Figure 4.13: Cascade of encoder, modulator and continuous-time channel.

In Fig. 4.13, the encoded data sequence $a_k \in \{-1, 1\}$ has data rate $1/T = 1/(RT_s)$ where $1/T_s$ is the source data rate and R is the information rate of the modulation code. A transmitted signal $s(t)$ of constrained power P_s emerges via a linear pulse modulator with symbol response $c(t)$. This response is typically a hold function of width T and almost always a function whose duration is linked to the symbol interval T. The energy $E_c = \int_{-\infty}^{\infty} c^2(t)dt$, therefore, depends linearly on R.

The channel has impulse response $f(t)$ and additive noise $n(t)$. Neither of these quantities depends on R. The cascade of modulator and channel has symbol response $h(t) \triangleq (c * f)(t)$. The energy $E_b \triangleq \int_{-\infty}^{\infty} h^2(t)dt$ is the received data energy per transmitted bit. We recall that E_b determines the per-symbol channel signal-to-noise ratio

SNR according to SNR $= E_b/N_0$ provided that $n(t)$ is white (see (3.9)). The power spectral density N_0 of $n(t)$ does not depend on R, so that any dependence of SNR on R must be brought about via E_b. We first consider this dependence for a low-pass channel. Two limiting cases are of particular interest:

Wideband channel: If the channel has a sufficiently large bandwidth then none of the transmitted data energy E_c will be rejected by the channel. This means that E_b, just as E_c, must be linearly proportional to R.

Narrowband channel: If the bandwidth of the channel is much smaller than that of $c(t)$ then a decrement of R (which increases the bandwidth of $c(t)$) will cause an increasing fraction of the energy of $c(t)$ to be rejected. At the same time the energy of $c(t)$ decreases linearly. The net result is that E_b decreases faster than linearly with R. The decrement is often quadratic ($E_b \propto R^2$).

Practical low-pass channels are in between these two extremes but are often biased towards the narrowband case.

Example 4.11 Let us assume that $c(t)$ is a hold function of width T and that the channel has a low-pass nature such that $f(t)$ extends over many symbol intervals T_s. Fig. 4.14 depicts corresponding symbol responses $c(t)$ and $h(t)$ for $R = 1$ and $R = 0.5$.

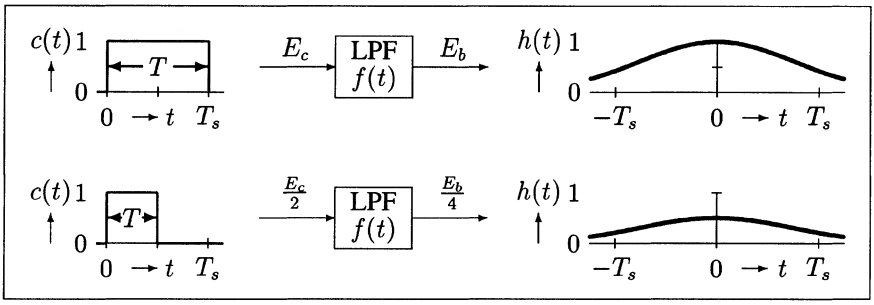


Figure 4.14: Illustration of the effect of signaling rate on received bit-energies for a narrowband channel. Top: $R = 1$; bottom: $R = 0.5$.

The hold function $c(t)$ is half as long for $R = 0.5$ as it is for $R = 1$ and has the same amplitude. For this reason it has only half the energy ($E_c/2$) of the $R = 1$ function. Even for $R = 1$ $c(t)$ is much shorter than $f(t)$ and therefore we can approximate the response of $f(t)$ to $c(t)$ by the response to a Dirac function of weight T. This leads to a symbol response $h(t) \simeq Tf(t)$ whose amplitude is linearly proportional to T. The energy E_b, therefore, is four times smaller for $R = 0.5$ than it is for $R = 1$.

The system in this example is somewhat reminiscent of a digital optical recording system and the limiting rates of decrement ($\propto R$ and $\propto R^2$) correspond to low and high normalized information densities S, respectively. Densities of practical interest are well into the R^2 range.

In Appendix 4C these limiting rates of decrement are bounded for a variety of channel transfer functions and noise PSDs. For low-pass channels, SNR degrades at least linearly in R. For band-pass channels, by comparison, the minimum rate of decrement may be zero or even negative, i.e. SNR can potentially increase as R decreases. The maximum rates of decrement are assessed in the Appendix for rectangular pulses $c(t)$ and rational code rates $R = 1/N$ ($N = 1, 2, 3, \ldots$) only. It is found that SNR degrades at most quadratically in R, irrespective of $f(t)$ and the noise PSD.

Example 4.12 The Lorentzian channel of Appendix 3A has a band-pass nature with a first-order zero at DC ($m = 1$ in the terminology of Appendix 4C). For this channel, $\mathrm{SNR}_R/\mathrm{SNR}_1 = [1 + D_u^2]/[1 + (D_u/R)^2]$ where the subscript denotes the code rate at hand and D_u denotes the information density normalized with respect to the 'user' bit interval T_s (see Appendix 3A). In Fig. 4.15, this ratio is depicted as a function of D_u for some practical information rates R.

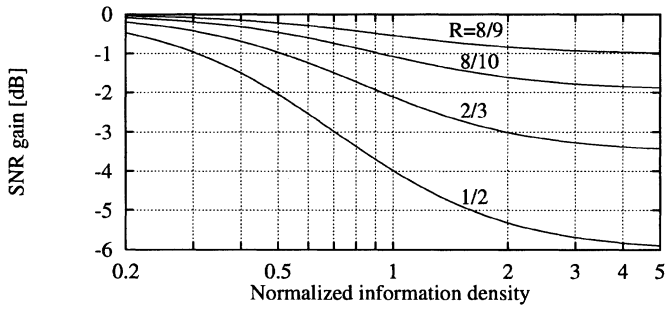


Figure 4.15: Signal-to-noise ratio gain $\mathrm{SNR}_R/\mathrm{SNR}_1$ versus normalized information density D for a Lorentzian channel at various information rates R.

For $D \ll R$, $\mathrm{SNR}_R \simeq \mathrm{SNR}_1$, i.e. there is no SNR penalty for adding redundancy. For $D \gg 1$, the case of greatest practical interest, SNR $\propto R^2$ and this is the highest rate of decrement that is possible according to Proposition 2 of the Appendix 4C.

4.4.2 Effect of coding on FSN$(e^{j2\pi\Omega})$

The folded signal-to-noise ratio FSN$(e^{j2\pi\Omega})$ governs the performance of a variety of receiver types and its behavior as a function of R is therefore of considerable interest. A first-order idea of this behavior can be obtained by noting that SNR may be regarded as the *arithmetic* average of FSN$(e^{j2\pi\Omega})$ (SNR $= \int_{-0.5}^{0.5} \mathrm{FSN}(e^{j2\pi\Omega})d\Omega$, see (3.7)). The foregoing results concerning SNR, therefore, also characterize the *average* degradation of FSN$(e^{j2\pi\Omega})$ as R decreases. It remains to determine the effect of R on the excursions of FSN$(e^{j2\pi\Omega})$ about this average. A detailed analysis of this effect is beyond the scope of this text. Instead we provide a qualitative discussion and we refer to Appendix 4C.3 for illustrative examples that pertain to idealized optical and magnetic recording systems, as well as to Appendix 4C.2 which provides a partial analysis for the situation that $c(t)$ is a hold function of width T. Again we distinguish between two limiting cases:

Narrowband channels: Here excursions of $\mathrm{FSN}(e^{j2\pi\Omega})$ about SNR tend to grow rapidly as R decreases. This causes the *geometric* and *harmonic* averages $\overline{\mathrm{FSN}}^G$ and $\overline{\mathrm{FSN}}^H$ of $\mathrm{FSN}(e^{j2\pi\Omega})$, as defined in Section 1.5, to drop ever further below SNR. Since SNR may degrade as much as quadratically with R, $\overline{\mathrm{FSN}}^G$ and $\overline{\mathrm{FSN}}^H$ may degrade even faster. In the chapters ahead we will see that this severely degrades the performance of various receivers.

Wideband channels: Here R tends to have little impact on the shape of $\mathrm{FSN}(e^{j2\pi\Omega})$, while SNR degrades at most linearly with R. Thus $\overline{\mathrm{FSN}}^G$ and $\overline{\mathrm{FSN}}^H$ degrade much less rapidly than for narrowband channels. This makes the addition of redundancy through the modulation code a much more acceptable proposition.

4.4.3 Effect of multilevel data

An increment of the size of the data alphabet permits the signaling rate to be lowered but will necessitate a smaller spacing between allowed data levels if the transmitted power P_s is to remain fixed. Extension of the above results to multilevel data alphabets requires that we account for this reduced spacing. We shall merely illustrate this extension here and leave a more detailed elaboration to the reader.

Example 4.13 Let us assume that $c(t)$ is a hold function of width T. For binary data $a_k \in \{-1, 1\}$ (2-AM), $c(t)$ must have amplitude 1 in order to arrive at a transmit power $P_s = 1$. By comparison, for uncorrelated quaternary data a_k in the 4-AM alphabet $\{-3, -1, 1, 3\}$ the amplitude must be $1/\sqrt{5}$ (see Table 3.1). This decreases SNR *a priori* by a factor 5. On the credit side, the symbol interval T has doubled with respect to binary data ($T = 2T_s$ versus $T = T_s$, i.e. $R = 2$ versus $R = 1$). For a wideband channel this will improve SNR by a factor of $2^1 = 2$ while a factor of $2^2 = 4$ is gained for a narrowband channel. The net effect of going from 2-AM to 4-AM is, therefore, an SNR loss that ranges between 1 and 4 dB. One might be tempted to conclude that going to 4-AM is a bad idea. This is, however, not necessarily true:

1. For a narrowband channel, $\mathrm{FSN}(e^{j2\pi\Omega})$ is bound to be much better-behaved for 4-AM than for 2-AM (see Section 4.4.2), and this may greatly improve receiver performance.
2. In our analysis we have assumed that the noise level of the channel does not depend on the transmitted signal. This assumption is not always valid. For example, twisted-pair cable-transmission systems suffer from crosstalk that is induced by capacitive coupling with neighboring wire pairs. This coupling is a high-frequency phenomenon and crosstalk can, therefore, be substantially reduced by lowering the signaling rate $1/T$. This is a principal reason why multilevel codes such as 2B1Q are favored for applications of this type.

4.5 EFFECT OF MODULATION CODE ON TIMING RECOVERY

Timing recovery schemes normally extract their timing information from the received signal $r(t)$. This approach is called *self-timing* and takes center stage in Chapters 9

and 10. There we will see that timing information resides in particular at high frequencies. This is intuitively reasonable. After all, a DC-wave conveys no timing information whatsoever, while an HF sinusoid contains many zero-crossings that may serve as a timing reference. It is, therefore, desirable that the transmitted signal $s(t)$ has significant HF content and does not become DC-like for prolonged periods of time. This will make it unlikely that timing recovery ever breaks down for lack of timing information. We shall discuss three techniques that fit into this context. An added advantage of the second and third technique is that both tend to improve *transparency*. The statistics of the encoded data a_k, therefore, tend to be relatively well-defined even when those of the user data d_n are not.

Transmit-side equalization The symbol response $c(t)$ of the linear pulse modulator in Fig. 4.2 can be modified to enhance the HF content of $s(t)$. Equivalently, the modulator can be extended with a *transmit filter* that enhances high frequencies. A fringe benefit of modifying $c(t)$ in this direction is that it acts to pre-compensate HF losses of the channel. This *transmit-side equalization* eases the burden on an equalizer at the receiving end of the system, and may also lead to better performances than with a receive-side equalizer alone (see Sections 5.7 and 6.2.3). On the debit side, a shift of the transmitted power towards high frequencies causes an increasing portion of this power to be rejected by the channel, and this, in turn, restricts per-symbol SNRs and the performance of optimum and near-optimum detection schemes (see Sections 3.3 and 3.9). In mathematical terms, it is shown in Section 10.2 that the *average* amount of timing information conveyed by the received signal $r(t)$ can be characterized by a merit factor

$$F \triangleq \int_{-\infty}^{\infty} (2\pi\Omega)^2 \mathrm{SN}_r(\Omega) d\Omega \tag{4.4}$$

where $\mathrm{SN}_r(\Omega)$ is the spectral signal-to-noise ratio of the received signal as defined in Section 3.2.4. An enlargement of F by 6 dB permits RMS jitter in the timing recovery subsystem to be halved at no expense to tracking properties. The weighing by Ω^2 reflects the fact that timing information is located especially at high frequencies.

Example 4.14 Digital recording channels usually accept only binary signals $s(t)$. These are produced by NRZ, for which $c(t)$ is a hold function of width T. One of the few possibilities to modify $c(t)$ while keeping $s(t)$ binary is to use a Bi-phase-like code as in Example 3.1 and Appendix 3A. The corresponding symbol response $c(t)$ and transfer function $C(\Omega)$ were shown along with those of NRZ in Fig. 3A.1. Bi-phase has no DC and much more HF content than NRZ. If the channel has little HF losses then most of this content will get across to the receiver, and Bi-phase will be preferable to NRZ from a timing standpoint. The roles reverse when HF losses are very large. A comparison may be based on the merit factor F of (4.4). For a Lorentzian channel this yardstick may be computed analytically (see Appendix 10A.2 and Problem 10.6). The ratio $F(\text{Bi-Phase})/F(\text{NRZ})$ is depicted in Fig. 4.16. A ratio of 3 dB means that Bi-phase tolerates 3 dB more channel noise than NRZ for the same average amount of received timing information.

Bi-phase is preferable to NRZ at low normalized information densities D, where HF losses are small. Roles reverse beyond $D \simeq 1.381$. By comparison, the cross-over point for SNR is

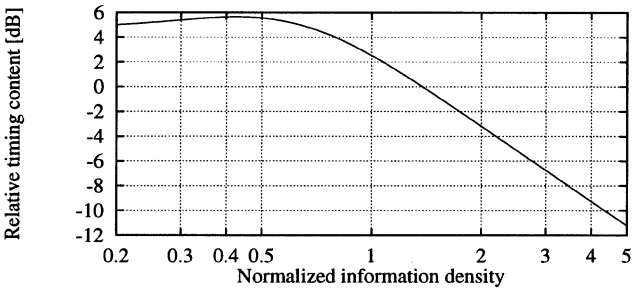


Figure 4.16: Relative timing content $F(\text{Bi-phase})/F(\text{NRZ})$ versus normalized information density D for Bi-phase and NRZ on a Lorentzian digital recording channel.

located at $D = 1/\sqrt{2} \simeq 0.707$ (see Appendices 3A and 3A.1). Thus Bi-phase remains superior for timing recovery long after it has ceased to be attractive from a detection standpoint.

Exercise 4.1 An NRZ signal $s(t)$ becomes DC-like when the data sequence a_k exhibits long strings of ones or zeros. What condition must be imposed on $c(t)$ to avoid this? Show that Bi-phase meets this condition. What is the longest interval between transitions for Bi-Phase?

Maximum-run-length constraint Data sources tend to produce long runs of ones or zeros in the 'idle' mode of operation. If these strings were transmitted directly, then $s(t)$ would not convey any timing information.[2] Digital recording systems often use run-length-limited (RLL) modulation codes that serve, among other purposes, to avoid this problem. RLL codes are discussed in more detail in Section 4.8. They have parameters d and k that stipulate the minimum and maximum run-lengths $d+1$ and $k+1$ of a_k (both expressed in symbol intervals T). Thus, a binary $(2, 7)$ code will produce sequences a_k with runs of at least 3 and at most 8 identical bits. The k constraint relates to timing recovery and guarantees that data transitions will occur regularly. This insurance has a price tag in the form of code redundancy. The ensuing increment of the signaling rate $1/T$ relative to the source data rate $1/T_s$ tends to lower signal-to-noise ratios (see Section 4.4). The information rate $R = T/T_s$ decreases with d, increases with k, and cannot exceed 1. The *capacity* C is the highest rate that can be achieved for a given pair (d, k) and should not be confused with the channel capacity of Chapter 3. Codes that achieve capacity are called *maxentropic*. Rates C and spectra $\mathcal{A}(e^{j2\pi\Omega})$ of maxentropic RLL codes are fully determined by d and k. Fig. 4.17 provides an illustration for $d = 0$.

Overhead is significant for the smallest values of k but C grows rapidly towards unity as k increases. Even for modest values of k (such as $k = 7$ or $k = 10$) the required redundancy is hardly worth mentioning and data spectra are virtually flat.

[2]The implicit assumption is that the condition of Exercise 4.1 is not met.

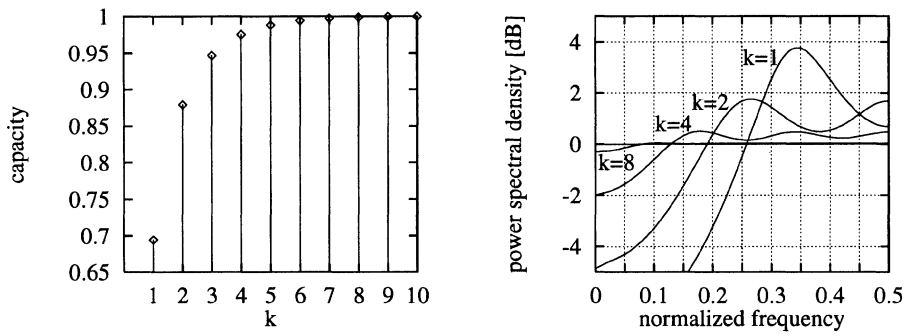


Figure 4.17: Capacity $\mathcal{C}$ and power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of maxentropic run-length-limited codes without minimum run-length constraint ($d = 0$).

Scrambling Long strings of ones or zeros may be avoided by randomizing or *scrambling* the data prior to transmission. Scrambling introduces no redundancy and therefore requires no bandwidth overhead. For this reason it is often used when bandwidth is at a premium, such as in cable and voiceband transmission. The debit side of this efficiency is that there will always be some source data sequence d_k that gets 'scrambled' into a DC-wave. This particular sequence d_k is, by design, highly random and thereby quite unlikely. Scrambling can, therefore, avoid long strings of ones and zeros with a high probability, but not with the absolute certainty of RLL codes. A basic topology is shown in Fig. 4.18. All sequences in Fig. 4.18 are binary and for mathematical convenience are taken to be in the alphabet $\{-1, 1\}$. Multiplications in this alphabet are equivalent to exclusive-or operations in the logical alphabet $\{0, 1\}$. The latter alphabet is more relevant from an implementation standpoint.

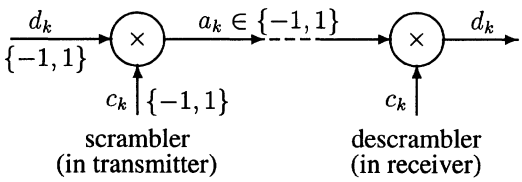


Figure 4.18: Frame-synchronized scrambler/descrambler combination.

In Fig. 4.18, scrambling is accomplished by multiplying the source data sequence d_k by an auxiliary sequence c_k. Descrambling requires a second multiplication with c_k. The result is a sequence $a_k c_k = d_k c_k^2 = d_k$, where the last step is due to the fact that $c_k \in \{-1, 1\}$. Hence d_k will be recovered correctly irrespective of c_k. This allows us to choose a sequence c_k with favorable scrambling properties. One commonly uses a so-called *pseudorandom sequence*. Such a sequence is *spectrally rich*, i.e. it contains a wide range of frequency components. This is desirable not only for timing recovery

but also for adaptation and detection. Appendix 4B describes some of the properties of pseudorandom sequences and shows how these sequences may be generated.

In order for the topology of Fig. 4.18 to work properly, the auxiliary sequences in transmitter and receiver must be in exact synchronism with each other. Systems in which data is transmitted block-by-block normally incorporate a frame-synchronization mechanism that can be used for this purpose. The combination of scrambler and descrambler is called *frame-synchronized* when it operates in this way [94]. To avoid the need for a synchronization mechanism, c_k may be generated locally on the basis of the scrambled sequence a_k, which is available to both transmitter and receiver. The *self-synchronized* arrangement of Fig. 4.19 is widely used in practice.

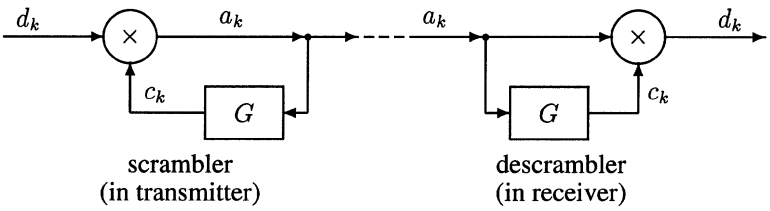


Figure 4.19: Self-synchronized scrambler/descrambler combination.

The mapping G which produces the auxiliary signal is normally such that a DC-wave d_k gets scrambled into a pseudorandom sequence. A disadvantage of self-synchronization is that the scrambler output a_k can sometimes become periodic with period P for a scrambler input d_k with period P. A further disadvantage is that the descrambler exhibits error propagation. Both problems can be alleviated by choosing G properly (see Appendix 4B).

4.6 EFFECT OF MODULATION CODE ON ADAPTATION

Adaptation techniques are aimed at adjusting the data receiver in such a manner as to minimize the harmful effects of ISI and noise (see Chapter 8). An elementary example is automatic gain control (AGC), which is performed in virtually any data receiver. More elaborate approaches include adaptation of transversal equalizers and multi-tap feedback filters.

Adaptation is similar in many respects to timing recovery. Specifically, adaptation schemes also extract control information from the incoming signal and generally cease to operate properly when this signal becomes DC-like for prolonged periods of time. For this reason, the measures of Section 4.5 can also be beneficial to adaptation. These measures are, however, not necessarily sufficient to ensure proper operation, especially when the number of adjusted parameters is large. We first consider a 2-parameter example and then dwell briefly on multi-parameter adaptation.

Joint adaptation of gain and slope Digital recording channels invariably have a gradual roll-off at high frequencies. The *slope* or degree of roll-off tends to be *a priori* uncertain and for this reason one often applies an adaptive slope equalizer in the data receiver. Variations of the gain of the channel are compensated for via an AGC circuit, while the slope equalizer compensates for unbalances between low and high frequencies. In order to be able to jointly determine the gain and slope, the received signal must contain at least two distinct frequency components. If the transmitted data has only a single component within the pass-band of the channel for a prolonged period of time then AGC and/or slope equalizer may misconverge. To avoid this we may impose constraints on the run-length of data patterns that give rise to such a component.

Practical recording systems always include one or more AC-couplings and at high densities they have negligible excess bandwidth. Their pass-band can, therefore, be considered to range from DC to the Nyquist frequency with both edge frequencies excluded. Table 4.1 lists all data patterns that have at most a single frequency component within this pass-band.

data pattern	period	constraint	spectral components at
$\ldots + + + + + + + + + + + + + \ldots$	$1T$	k	DC
$\ldots + - + - + - + - + - + - \ldots$	$2T$	k_2	$1/(2T)$
$\ldots + + - + + - + + - + + - \ldots$	$3T$	k_3	DC, $1/(3T)$
$\ldots + + - - + + - - + + - - \ldots$	$4T$	k_4^a	$1/(4T)$
$\ldots + + + - + + + - + + + - \ldots$	$4T$	k_4^b	DC, $1/(4T)$, $1/(2T)$
$\ldots + + + - - - + + + - - - \ldots$	$6T$	k_6	$1/(6T)$, $1/(2T)$

Table 4.1: Overview of data patterns with zero or one spectral components in $(0, 1/(2T))$. Per type of pattern only a single polarity is shown. Polarity reversal does not affect the magnitude of the spectral components.

The maximum run-length of the DC-pattern in Table 4.1 may be characterized by a k-constraint as in Section 4.5. In similar vein we can define constraints k_2, k_3, k_4^a, k_b^4 and k_6 that define the maximum run-lengths k_2+1, k_3+1, k_4^a+1, k_4^b+1 and k_6+1 of the patterns with period 2, 3, 4 and 6. Somewhat plastically we designate a modulation code that introduces these constraints an *anti-whistle code.* Anti-whistle codes may be regarded as a generalization of run-length-limited codes, which only impose constraints on 'whistles' at DC and the Nyquist frequency (and, in one example, on the $4T$ pattern of the k_4^a type [25]). In Section 4.8.8 we discuss a class of simple anti-whistle codes of the polarity-bit type.

Joint adaptation of multiple parameters As the number of adapted parameters increases, the transmitted data has to become increasingly *spectrally rich* in order for adaptation to work properly. As a rule of thumb, the minimum number of spectral components in the received signal should not be smaller than the number of adapted parameters (see Section 8.6.1 for more precise statements). This can, in principle, be achieved via additional run-length constraints akin to the ones discussed above. Un-

fortunately, the number of constraints that must be imposed increases rapidly with the number of adapted parameters and this renders code design unwieldy. Instead one often settles for a scrambler so as to ensure spectral richness in the statistical (as opposed to worst-case) sense.

4.7 CODE MECHANIZATION

In the sections ahead we will discuss numerous modulation codes in varying degrees of detail. Application fields and properties of these codes vary widely, but their mechanizations have many traits in common. We now discuss some of these traits along with related notions.

In order to keep implementation manageable, modulation encoders almost always have a *rational* code rate $R = p/q$, i.e. groups of p source bits d_n are encoded into *code words* of q symbols a_k. Sometimes p and q vary dynamically in such a manner that their ratio remains fixed. This is the distinguishing feature of *variable-length* codes. Most codes in use today, however, are of the *fixed-length* type, i.e. p and q do not change over time.

It is generally desirable that R be as large as possible. The highest achievable rate is the *capacity* $\mathcal{C}$ and depends on the alphabet of a_k as well as on any constraints that are imposed on a_k. Practical encoding schemes often have $R < \mathcal{C}$, especially when $\mathcal{C}$ is not rational. The ratio $R/\mathcal{C}$ is called the *efficiency* of the code.

Example 4.15 The 2B1Q line code maps groups of $p = 2$ source bits into a single quaternary symbol a_k and therefore has rate $R = 2$. It is clearly impossible to accommodate more than 2 bits in a single quaternary symbol, i.e. $\mathcal{C} = 1$ and 2B1Q has an efficiency of 100%.

Example 4.16 GCR is a binary code with rate $R = 4/5$ that limits the maximum run-length of a_k to 3 symbol intervals. In Section 4.8 we will see that the capacity $\mathcal{C}$ for this run-length constraint is $\mathcal{C} = 0.8972$ and GCR, therefore, has an efficiency of 89.17%.

Example 4.17 The category of $4B/3T$ codes (see Examples 4.37 and 4.38) maps groups of $p = 4$ source bits into code words of $q = 3$ ternary symbols a_k, i.e. $R = 4/3$. The maximum amount of information that can be conveyed with 3 ternary digits is $\mathcal{C} = 3\log_2 3 \simeq 4.755$ bits and the efficiency can be rated as 84.12%. The redundancy of 15.88% is used in part to keep the running digital sum within strict bounds and this figure can be compared with the minimum redundancy that will serve this purpose. Relative to this minimum, practical $4B/3T$ codes achieve efficiencies of around 90% (see Problem 4.23).

Conceptually, the simplest way to obtain a code with rate $R = p/q$ is via a look-up table that is addressed by groups of p bits at a time. For each address there is one position in the table that stores the corresponding code word. This approach, called *state-independent encoding*, is easily mechanized as long as p and q are not too large. Unfortunately, p and q must often be large if we insist on a reasonable efficiency, and therefore encoders are frequently of the *state-dependent* type (Fig. 4.20).

The encoder accepts input blocks of p source bits and generates a code word of length q depending on the input block and the current internal *state* S_n of the encoder.

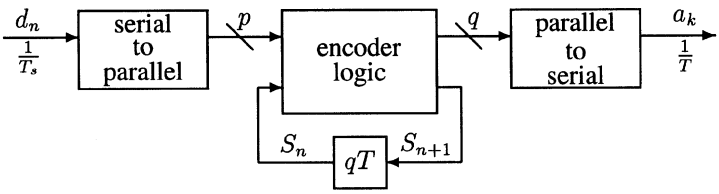


Figure 4.20: Finite-state encoder.

There are, of course, only finitely many states. This encoder structure is ideally suited to digital logic implementation. Its main advantage is that p and q can often be much smaller than they would be for a state-independent code with the same properties. The *raison d'être* of variable-length codes is pretty much the same.

Example 4.18 The $(1,7)$ RLL code of Section 4.8.6 has a simple 5-state encoder with word-lengths $p = 2$ and $q = 3$. A state-independent encoder for the same run-length constraints requires the prohibitive word-lengths $p = 22$ and $q = 33$ [34].

The variable-length $(2,7)$ code of Section 4.8.5 uses a simple encoding table with word-lengths $p = 2$ and $q = 3$. A fixed-length state-independent encoder for these run-length constraints would need $p = 17$ and $q = 34$ [34]. This is also impractical.

The action of any finite-state encoder can be described with the aid of an *encoding table* and also with a *finite-state transition diagram* (FSTD). This diagram is a directed graph in which vertices represent the states and edges represent possible transitions between states.

S_n	d_n	S_{n+1}	(a_{2n}, a_{2n+1})
$-$	$-$	$+$	$(+,-)$
$-$	$+$	$-$	$(-,+)$
$+$	$-$	$-$	$(-,+)$
$+$	$+$	$+$	$(+,-)$

Table 4.2: FM encoding table; '$+$' and '$-$' are shorthand notation for '$+1$' and '-1', respectively.

Example 4.19 FM is a binary modulation code with rate $1/2$ that maps binary data symbols $d_n \in \{-1,1\}$ into code words of two binary digits $a_k \in \{-1,1\}$. It has a single binary-valued state $S_n \in \{-1,1\}$. At any given instant n, the next encoder outputs (a_{2n}, a_{2n+1}) and the next state S_{n+1} are both fully determined by the incoming digit d_n and the current state S_n (Table 4.2).

The FSTD of Fig. 4.21 has two states and four edges that indicate the possible transitions between states. Labels alongside edges are of the form $d_n|a_{2n}a_{2n+1}$. Every edge is uniquely determined by the current state and the incoming symbol d_n, and in turn determines the next state and a code word $a_{2n}a_{2n+1}$. The designation 'FM' stems from the fact that the effect of the code is somewhat akin to analog frequency modulation (see Example 4.31).

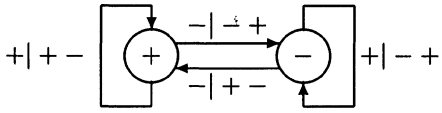


Figure 4.21: Finite-state transition diagram of FM encoder.

The source data sequence d_n can, in principle, be recovered from the code words by retracing the state sequence S_n. This *state-dependent decoding* will work well as long as the decoder operates on a valid encoded sequence. The decoder, however, normally resides in a receiver that is prone to decision errors. A single error at the input of a state-dependent decoder can cause the decoder to lose track of the encoder state sequence, with no guarantee of recovery and with the possibility of unbounded or *catastrophic* error propagation. This is obviously unacceptable and it is therefore common practice to design the encoder in such a manner as to permit the use of a *sliding-block* decoder (Fig. 4.22). A decoder of this type reconstructs a group of p source bits on the basis of the corresponding code word as well as a fixed number m of preceding words and a fixed number a of later words. The preceding symbols constitute the *memory* of the decoder and the following symbols are called the *anticipation.* A single error at the input to a sliding-block decoder can only affect the decoding of words that fall into a *decoding window* of at most $m + a + 1$ words. Design of the decoding logic can be cumbersome when done mathematically. A simple work-around involves the use of adaptive look-up tables [8]. This approach is applicable to a wide variety of codes and also to decoding techniques that are simplified via partial-response equalization.

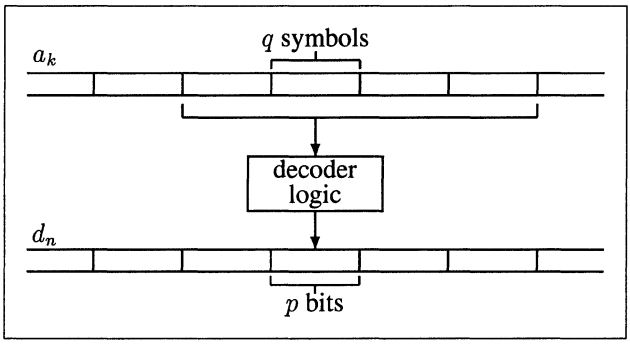


Figure 4.22: Sliding-block decoder.

4.8 RUN-LENGTH-LIMITED CODES

Run-length-limited (RLL) modulation codes have found widespread application in digital magnetic and optical recording systems. They were pioneered in the late 1960's by

P. Franaszek [34] and have since provided a fertile working ground for coding theorists. Major contributions include those of Tang and Bahl [112], Jacoby [56, 57], Franaszek [38, 39], Patel [87], Lempel and Cohn [69], and, more recently, Adler, Hassner and Coppersmith [3], Gallopoulos, Heegard and Siegel [46], and Hollmann [55]. These and other contributions have described numerous codes that have found their way into storage products, and have matured the subject area to the extent that systematic procedures are now available for the design of simple yet highly efficient encoders and decoders. The present section provides a concise review of this body of work. We refer to [105, 99, 100] and to the excellent tutorial of [78] for more extensive reviews.

The principal technical merit of RLL codes is their ability to ease life for the receiver, in terms of both reduced ISI and the provision of a certified amount of timing information. These two benefits, which were illustrated in Sections 4.3 and 4.5, carry a price tag in the form of redundancy that must be added to the data stream. To accommodate this overhead the signaling rate $1/T$ must be enlarged, and this, in turn, lowers per-symbol signal-to-noise ratios and thereby ultimately attainable performances and/or information densities (see Section 4.4; see also [89] for early observations in this context). Interestingly, RLL codes originally found their way into storage products because they were thought, among other advantages, to offer density improvements [105]. This thought continues to reverberate [117, 99].

RLL codes commonly have two parameters, d and k, that stipulate the minimum and maximum lengths of runs of consecutive ones and zeros in the data stream.[3] The d-constraint controls the high-frequency content of the data stream and the k-constraint controls the low-frequency content. In practice d is never larger than 2, while k ranges roughly between 2 and 10.

Example 4.20 GCR (Section 4.8.3) is a $(0, 2)$ code that is used in magnetic tape drives [58]. Most floppy disk drives and many low-end hard disk drives use MFM, which is also known as Delay Modulation (DM) and Miller Code. This $(1, 3)$ code is described in Section 4.8.4. Hard disk drives have long used the $(2, 7)$ and $(1, 7)$ codes that are described in sections 4.8.5 and 4.8.6. The compact disk system uses a $(2, 10)$ code called EFM (Eight-to-Fourteen Modulation).

4.8.1 dk-sequences

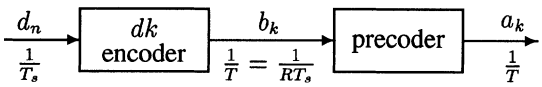


Figure 4.23: RLL encoder. The source data symbols d_n are first converted into a binary dk-sequence b_k, from which the RLL sequence a_k is derived via a precoder.

[3]Just as in other chapters we also use the symbol k as a time index in symbol intervals. It should be clear from the context which interpretation is meant.

The RLL-encoding process usually involves two steps (Fig. 4.23). The first and major step delivers a so-called dk-sequence b_k in the logical alphabet $\{0, 1\}$, which is then mapped into an RLL sequence $a_k \in \{-1, 1\}$ with the aid of a precoder. The dk sequence is closely related to the RLL sequence and satisfies two conditions:

1. two logical ones are separated by a run of at least d consecutive zeros (d-constraint);

2. no run of consecutive zeros lasts longer than k symbol intervals (k-constraint).

The logical ones in the dk sequence mark the positions where a transition ($+1 \rightarrow -1$ or $-1 \rightarrow +1$) occurs in the RLL sequence. In recording terminology this notation is sometimes called *NRZI* (non-return-to-zero-inverse).[4] The precoder performs *NRZI encoding*, i.e. it transforms the transition markers into actual transitions. Precoding is also applied in many linear modulation codes (see Section 4.10). The minimum and maximum run-lengths of the RLL sequence are $d + 1$ and $k + 1$, respectively.

Example 4.21 The dk-sequence 00101000001000100 has $d = 1$ and $k = 5$ and will, depending on the initial state of the precoder, be transformed into either $++--+++++++--$ $---+++$ or its inverse ($--++-------++++---$), where $+$ and $-$ are shorthand for $+1$ and -1.

For the sake of completeness we mention that the precoding step, while commonly used, is not strictly necessary. Instead, the coder can be designed to directly produce run-length-limited sequences (see [102] for an illustration). Key advantage of this approach is that there are twice as many code words available for a given code-word length.

4.8.2 Capacity of RLL codes

The conversion of the source data sequence d_n into a dk sequence is only possible at the cost of adding redundancy. This will increase the data rate $1/T$ of the dk sequence relative to the rate $1/T_s$ of d_n. The *information rate* $R \triangleq T/T_s$ of the encoder, therefore, cannot be larger than unity. The capacity C is the highest rate that is attainable for given values of d and k. This rate was derived in 1948 by Shannon [104] and is given by

$$C = \log_2 \lambda \tag{4.5}$$

where λ is the largest real-valued root of the following equations, depending on whether k is infinite or not:

$$\begin{aligned} x^{k+2} - x^{k+1} - x^{k+1-d} + 1 = 0, \quad k < \infty \\ x^{d+1} - x^d + 1 = 0, \quad k = \infty. \end{aligned} \tag{4.6}$$

Table 4.3 lists C for various values of d and k.

[4]This obscure terminology is rooted in the annals of telegraphy.

k	$d=0$	$d=1$	$d=2$	$d=3$
1	.6942			
2	.8791	.4057		
3	.9468	.5515	.2878	
4	.9752	.6174	.4057	.2232
5	.9881	.6509	.4650	.3218
6	.9942	.6690	.4947	.3746
7	.9971	.6793	.5174	.4057
8	.9886	.6853	.5293	.4251
9	.9993	.6888	.5369	.4376
10	.9996	.6909	.5418	.4460
∞	1.000	.6942	.5515	.4650

Table 4.3: Capacity $\mathcal{C}$ of run-length-limited codes for various values of d and k.

An RLL code that achieves capacity is called *maxentropic*. The run-lengths $T_m \in \{d+1, \ldots, k+1\}$ produced by a maxentropic RLL code form a stationary sequence whose probabilities $\mathrm{pr}(T_m)$ follow a truncated geometric distribution with parameter $\lambda = 2^{\mathcal{C}}$, i.e.

$$\mathrm{pr}(T_m) = \lambda^{-T_m} = 2^{-T_m \mathcal{C}} \text{ for } T_m \in \{d+1, \ldots, k+1\}. \tag{4.7}$$

The average run-length $\overline{T}$ amounts to

$$\overline{T} = \sum_{T_m=d+1}^{k+1} T_m \mathrm{pr}(T_m) = \sum_{i=d+1}^{k+1} i\lambda^{-i}. \tag{4.8}$$

Exercise 4.2 Prove that $\mathrm{pr}(T_m)$ defines a proper distribution function, i.e. that the sum of all probabilities amounts to one. Hint: Use (4.6).

The power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of maxentropic RLL sequences a_k was first derived by Foley (see, for example, [67]) and may be expressed as

$$\mathcal{A}(e^{j2\pi\Omega}) = \frac{1}{\overline{T}} \frac{1}{\sin^2 \pi\Omega} \frac{1 - |G(e^{j2\pi\Omega})|^2}{|1 + G(e^{j2\pi\Omega})|^2} \text{ where } G(e^{j2\pi\Omega}) \triangleq \sum_{i=d+1}^{k+1} \lambda^{-i} e^{j2\pi\Omega i}. \tag{4.9}$$

Some spectra with $d = 0$ were already shown in Fig. 4.17. Further spectra are depicted in Fig. 4.24 for $d = 1$ and $d = 2$. All spectra have a bandpass nature. This nature becomes more pronounced as d increases and as k decreases. An increment of d causes the bulk of the spectrum to shift towards lower frequencies. It should be noted that frequencies in Fig. 4.24 are normalized with respect to the signaling rate $1/T$, which is a function on the rate R of the code. A comparison in terms of absolute frequencies, therefore, would yield a different picture.

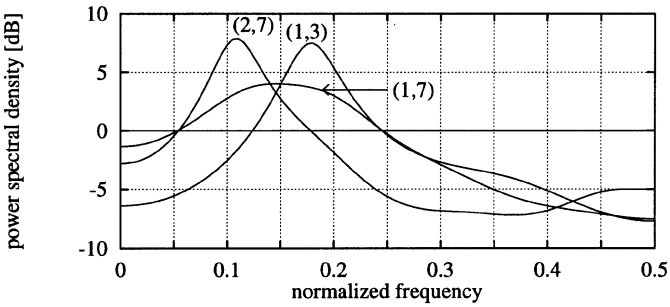


Figure 4.24: Power spectral densities of maxentropic run-length-limited sequences for some popular pairs (d, k).

Exercise 4.3 Use Table 4.3 to determine the absolute frequencies of the spectral peaks in Fig. 4.24. What is the lowest of these frequencies and to which code does it correspond?

A minimum transition spacing of $(d+1)$ symbol intervals T corresponds to $(d+1)R$ units T_s, as compared to a single unit T_s for an uncoded system. The product $(d+1)R$, called the *density ratio* of the code, determines whether transitions are spaced closer together $[(d+1)R < 1]$ or further apart $[(d+1)R > 1]$ than for an uncoded system. A higher density ratio tends to lower intersymbol interference and this, in turn, tends to improve performance of peak detectors that are commonly used with RLL codes.

Code	d	k	p	q	R	R/C (%)	$(d+1)R$
GCR	0	2	4	5	4/5	91	0.80
MFM	1	3	1	2	1/2	91	1.00
$(1,7)$	1	7	2	3	2/3	98	1.33
$(2,7)$	2	7	1	2	1/2	97	1.50

Table 4.4: Basic parameters of some commonly used run-length-limited codes.

Example 4.22 Table 4.4 summarizes the density ratio of some popular RLL codes. Also listed are the run-length constraints, the word lengths p and q of incoming and outgoing bits, the rate $R = p/q$, and the efficiency R/C of the codes. Density ratios range between 0.8 and 1.5; efficiencies are all above 90% and range up to 98%.

We proceed to discuss several examples of RLL codes. Our discussion follows chronological lines and reflects an evolution of somewhat heuristic design approaches towards highly systematic ones. We will draw repeatedly on the terminology that was introduced in Section 4.7, and the reader may wish to review that section before passing on.

4.8.3 Group-Coded Recording (GCR) Code

GCR is a fixed-length state-independent $(0, 2)$ code that is used in a variety of magnetic tape drives. It is also known as '4/5 code'. Encoding rules are listed in Table 4.5. Source bits are collected in groups of length $p = 4$ and mapped into code words of $q = 5$ bits. A subsequent precoding operation produces the data sequence a_k. It is easily verified that any concatenation of code words satisfies the $(0, 2)$ constraint. GCR has rate $R = 4/5$ and an efficiency of 91% (see Table 4.4). Its PSD closely approximates the $(0, 2)$ spectrum of Fig. 4.17. Since the encoder is state-independent, the decoder must also be state-independent and the encoding table can be re-used as a decoding table.

Source bits	dk code word	Source bits	dk code word
0000	11001	1000	11010
0001	11011	1001	01001
0010	10010	1010	01010
0011	10011	1011	01011
0100	11101	1100	11110
0101	10101	1101	01101
0110	10110	1110	01110
0111	10111	1111	01111

Table 4.5: Encoding/decoding table for GCR code.

4.8.4 Modified Frequency Modulation (MFM)

MFM is a $(1, 3)$ code with rate $R = 1/2$ that has found widespread application in floppy and hard disk drives. It was contrived by A. Miller as a modification to the FM code of Example 4.19 [82] (see also Section 4.10), and is also known as Miller code and Delay Modulation (DM). MFM maps every incoming source bit d_n into a code word (b_{2n}, b_{2n+1}) of length $q = 2$. The resulting sequence b_k is applied to a precoder to obtain the channel sequence a_k. The encoder may be described by the two-state transition diagram of Fig. 4.25.

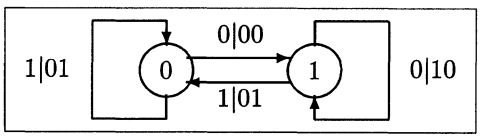


Figure 4.25: Two-state transition diagram of MFM encoder.

Labels along edges in Fig. 4.25 are of the form $d_n|b_{2n}b_{2n+1}$. It should be stressed that the diagram, unlike the one of Fig. 4.21, does not account for the precoding operation that completes the encoding process. It can be observed that the second digit of

every code word is just the incoming source bit and only the first one is 'invented' by the code. A code of this type, in which code words are obtained by simply inserting redundant bits between the incoming source bits, is said to be *systematic*. A principal advantage of systematic codes is that the decoder needs only skip the redundant bits in order to recover the source data sequence. For MFM, the 'redundant' bit is zero unless it is to be inserted between two zeros ($d_{n-1} = d_n = 0$), in which case it is set to one.

Exercise 4.4 Use the diagram of Fig. 4.25 to verify that MFM is indeed a $(1, 3)$ code.

With rate $R = 1/2$, MFM has 91% efficiency (see Table 4.4). The PSD of MFM may be determined analytically [53] and is found to be

$$\mathcal{A}(e^{j2\pi\Omega}) = \frac{3 + \cos 2\pi\Omega + 2\cos 4\pi\Omega - \cos 6\pi\Omega}{9 + 12\cos 4\pi\Omega + 4\cos 8\pi\Omega}. \tag{4.10}$$

This function closely resembles the $(1, 3)$ PSD of Fig. 4.24.

4.8.5 Rate $1/2$ $(2, 7)$ code

A code with these parameters is employed in several disk-drive products. The variable-length code that was invented by P. Franaszek [35] is a well-known example (Table 4.6).

Source bits	dk code word
10	0100
11	1000
000	000100
010	100100
011	001000
0010	00100100
0011	00001000

Table 4.6: Encoding/decoding table for rate $1/2$ $(2, 7)$ code.

The source data sequence is first parsed into two-, three- and four-bit partitions to match the entries of the table, and is then mapped into the corresponding dk sequence. For example, the source sequence 10110010010... would be parsed as 10 11 0010 010... and thus mapped into 0100 1000 00100100 100100.... This dk sequence obviously satisfies the $(2, 7)$ constraint. A precoding step completes the encoding procedure.

It can be observed that any given code word is not the beginning or *prefix* of any other code word. A code that satisfies this condition is said to be a *prefix code*. Prefix codes are *self-punctuating*, i.e. decoding can be accomplished without explicitly knowing where the blocks of variable length start or end. In the case at hand the blocks all have an even length and it will, therefore, be necessary to know which of the bits are the even ones. Decoding is possible with a simple sliding-block decoder [30]. Any code

digit b affects a total of 4 decoded source bits d and any bit-error, therefore, propagates to at most four decoded data bits. The above encoding table has been judiciously designed to limit error propagation to this extent.

The variable-length encoder according to Table 4.6 can be translated into an equivalent encoder of the fixed-length type [30]. That encoder can be described with a 4-state transition diagram. The power spectral density of the $(2,7)$ code under consideration has been derived analytically via the elegant method of [46], which is tailored specifically to RLL codes. This PSD closely resembles the $(2,7)$ PSD of Fig. 4.24 and is given by $\mathcal{A}(e^{j2\pi\Omega}) = P(e^{j2\pi\Omega})/[21Q(e^{j2\pi\Omega})]$ where $P(e^{j2\pi\Omega}) \triangleq 1956 + 1464\cos 2\pi\Omega + 424\cos 4\pi\Omega - 762\cos 6\pi\Omega + 54\cos 8\pi\Omega - 408\cos 10\pi\Omega + 8\cos 12\pi\Omega - 160\cos 14\pi\Omega + 32\cos 16\pi\Omega + 48\cos 18\pi\Omega + 24\cos 20\pi\Omega$ and $Q(e^{j2\pi\Omega}) \triangleq 162 + 16\cos 4\pi\Omega + 128\cos 8\pi\Omega + 32\cos 12\pi\Omega$.

4.8.6 Rate $2/3$ $(1,7)$ code

Rate $2/3$ $(1,7)$ codes are used today in many digital recording systems. Several codes with these parameters have been developed. Two of the more popular ones, developed independently by Cohn, Jacoby and Bates [26] and Adler, Hassner and Moussouris [4], were shown in [105] to differ only in the choice of data-to-codeword assignments. Encoding rules of the code of [26] are listed in Table 4.7. There are 5 states that are labeled A, B, C, D and V. Incoming bits are grouped into pairs ($p = 2$) and each pair, together with the current state of the encoder, determines a code word of $q = 3$ bits along with the new state. The decoder is similar to the one for $(2,7)$ and has a maximum error propagation of 5 source data bits. The smallest possible number of states for a $(1,7)$ code with rate $R = 2/3$ is 4 [77, 118].

	State				
Data	A	B	C	D	V
00	$101\|V$	$100\|A$	$001\|V$	$010\|A$	$000\|A$
01	$100\|V$	$100\|B$	$010\|V$	$010\|B$	$000\|B$
10	$101\|C$	$100\|C$	$001\|C$	$010\|C$	$000\|C$
11	$101\|D$	$100\|D$	$001\|D$	$010\|D$	$000\|D$

Table 4.7: Encoding table of $(1,7)$ code with rate $R = 2/3$. Entries in 'data' colum indicate pairs of source digits d_{2m}, d_{2n+1}. Entries in 'State' columns are of the form $XYZ|S$ where XYZ denotes the code word and S denotes the new state.

The power spectral density of the code of Table 4.7 closely approximates the $(1,7)$ PSD of Fig. 4.24 and is given by $\mathcal{A}(e^{j2\pi\Omega}) = U(e^{j2\pi\Omega})/[15V(e^{j2\pi\Omega})]$ where $U(e^{j2\pi\Omega}) \triangleq 15681 + 9182 * \cos 2\pi\Omega - 96\cos 4\pi\Omega + 11330\cos 6\pi\Omega + 2996\cos 8\pi\Omega - 1778\cos 10\pi\Omega + 1784\cos 12\pi\Omega + 816\cos 14\pi\Omega - 200\cos 16\pi\Omega$ and $V(e^{j2\pi\Omega}) \triangleq 1665 + 1936\cos 6\pi\Omega + 560\cos 12\pi\Omega + 64\cos 18\pi\Omega$ [46].

4.8.7 Rate $8/9$ $(0, G|I)$ codes

These codes are tailored to Viterbi detection on Class IV partial-response $(1 - \mathcal{D}^2)$ channels. They introduce maximum-run-length constraints that are aimed on the one hand at facilitating timing recovery and automatic gain control, and on the other hand at restricting the path memory length of the Viterbi detector. Digital magnetic recording systems constitute the principal application area.

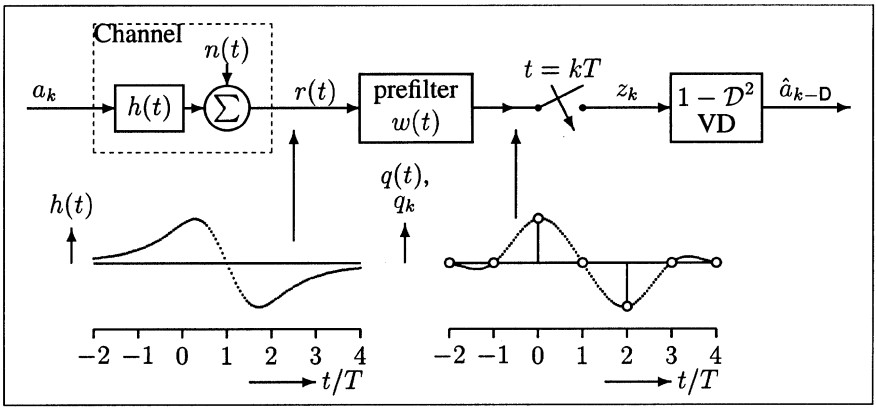


Figure 4.26: Partial-response maximum-likelihood (PRML) reception in a digital recording system.

The magnetic recording channel has a zero at DC and this renders full-response equalization as in Example 4.4 unattractive. Instead one often equalizes the channel into a so-called Class IV partial response (Fig. 4.26). The gist of partial-response techniques is that a controlled amount of ISI is left for the detector to handle. The Class IV response is characterized by a $\mathcal{D}$-transform $1 - \mathcal{D}^2$, i.e. the detector input z_k has a ternary data component $x_k = a_k - a_{k-2}$. The band-pass nature of Class IV resembles that of the magnetic recording channel quite well, and as a result noise enhancement is much smaller than it would be for a full-response equalizer. The combination of Class IV equalization and Viterbi detection is commonly designated PRML (Partial-Response Maximum-Likelihood) reception. In Chapters 6 and 7 we delve deeper into the intricacies of this approach.

Class IV has zeros at both DC and the Nyquist frequency. As a result, the data sequences '$\ldots + + + + \ldots$', '$\ldots - - - - \ldots$' and '$\ldots + - + - + - + - \ldots$' are completely rejected and cause x_k to become zero. In this event z_k contains no timing and/or gain information whatsoever, i.e. timing-recovery and/or gain-control circuits may misconverge. To prevent this anomaly from occurring for a prolonged amount of time, it is imperative that runs of the above three patterns have a restricted duration. A k constraint as in standard RLL codes will ensure this for the two DC-patterns. An additional constraint must be imposed on the run-length of the Nyquist-rate pattern '$\ldots + - + - + - \ldots$'. In Section 4.6 we have dubbed this a k_2-constraint. The combined

effect of the k and k_2 constraints is that x_k cannot be zero indefinitely. The maximum run-length of these zeros is denoted G.

An attractive feature of Class IV is that the even and odd data streams a_{2m} and a_{2m+1} can be detected independently by means of two separate Viterbi detectors that operate on the interleaved detector inputs z_{2m} and z_{2m+1}, respectively (see Section 7.6). Both detectors contain a path memory whose complexity can be restricted via a constraint on the maximum run-length of zeros in the interleaved Class IV sequences x_{2m} and x_{2m+1}. This length is denoted I.

Various existing PRML systems employ modulation codes with rate $R = 8/9$ that introduce both G and I constraints. These codes are designated $(0, G|I)$ where the '0' serves as a reminder that there is no minimum run-length constraint ($d = 0$). The choice $R = 8/9$ reflects a preference for byte-oriented processing and minimum code redundancy. Typical values for $G|I$ are 4|4 and 3|5. The rate $R = 8/9$ remains well below the capacity C for these constraints and for this reason encoders have only one or two states and are relatively simple [106, 78]. Furthermore, decoding involves a sliding-block decoder without memory and/or anticipation, i.e. there is no error propagation beyond code-word boundaries. Complexity and error-propagation properties become less attractive as G and I are reduced and C approaches R. Further details on the design and implementation of these codes may be found in [106], [78], [31] and [76].

In order to facilitate acquisition of the timing-recovery and automatic gain-control loops, it is sometimes desirable to supplement the G and I constraints with a constraint on the maximum run-length M of non-zero symbols x_k. Backgrounds of this constraint as well as the detailed design of an encoder and decoder with $(G, I, M) = (4, 4, 5)$ are described in [25]. The M constraint is closely related to the k_4 constraint of Section 4.6, and more in particular to the maximum run-length of the pattern '$\ldots + + - - + + - - \ldots$'.

4.8.8 Rate $8/9$ anti-whistle polarity-bit codes

These codes impose not only a k constraint but also k_2, k_3, k_4^a, k_4^b and k_6 constraints as defined in Section 4.6. These constraints are beneficial to timing recovery, gain control and slope control, particularly in the face of the severe bandwidth limitations that are encountered in high-density recording. Design of efficient anti-whistle codes can be based on a variety of systematic techniques, like those of [78] and [55]. If simplicity is more important than efficiency then other venues can be attractive. One option involves the use of the polarity-bit mechanism. Here the data stream is subdivided into words of $n - 1$ bits, which are either sent as they are or all inverted (i.e. multiplied by -1). Whether or not the bits have been inverted is communicated by means of a *polarity bit* at the end of the word. The polarity of the code word is classically chosen to obtain a spectral zero at DC (see Section 4.11.1), but can also be selected to 'break' whistles at the edges of code words.

The basic polarity-bit mechanism has the disadvantage that erroneous detection of a polarity bit results in misdetection of all $n-1$ associated data bits. A word-wise precod-

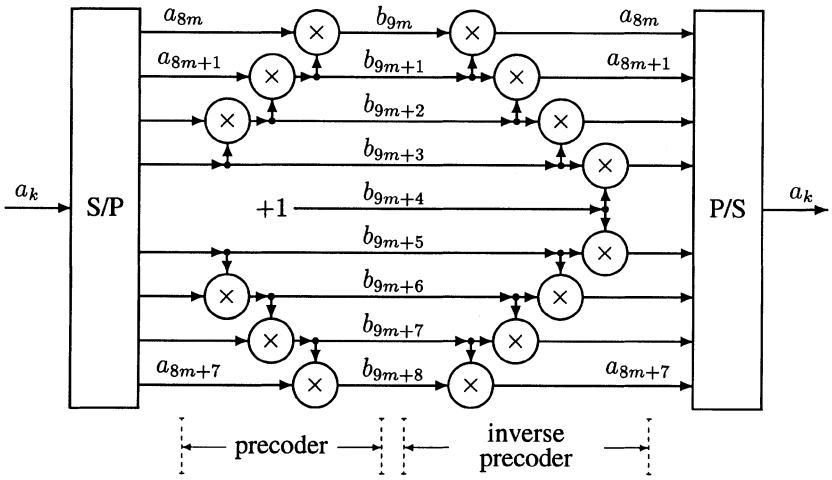


Figure 4.27: Byte-wise precoder and inverse precoder.

ing and inverse precoding operation can restrict error propagation to a single adjacent bit. Fig. 4.27 illustrates the approach for the situation that incoming bits are grouped into bytes ($n-1=8$). This corresponds to a code rate $R=(n-1)/n=8/9$.

The precoder is constructed in such a manner that all source bits can be reconstructed from two neighboring precoder outputs. The circuit of Fig. 4.27 achieves this objective with the smallest possible logical depth. An incoming data stream a_k is split into bytes $a_{8m}\ldots a_{8m+7}$ by means of a serial-to-parallel converter. The precoder produces a 9-bit word whose central bit b_{9m+4} is fixed at unity. The neighboring bits b_{9m+3} and b_{9m+5} are just the source bits a_{8m+3} and a_{8m+4}. As we move further away from the center, the precoder output becomes a product of ever more source bits. Reconstruction of source bits requires a pair-wise comparison of precoder outputs. For example, $b_{9m}=a_{8m}a_{8m+1}a_{8m+2}a_{8m+3}$ and $b_{9m+1}=a_{8m+1}a_{8m+2}a_{m+3}$ so that $b_{9m}b_{9m+1}=a_{8m}a^2_{8m+1}a^2_{8m+2}a^2_{8m+3}=a_{8m}$. A total of 8 binary multipliers suffices to reconstruct all source bits. This task is accomplished by the *inverse precoder* of Fig. 4.27. A key property of the inverse precoder is its polarity-insensitivity. Specifically, if we invert all bits $b_{9m},\ldots,b_{9m+8}$ simultaneously then the output of the inverse precoder will not change. This freedom can be used to create spectral zeros along the lines of Section 4.11.1, or to impose run-length constraints as in Fig. 4.28.

Precoder outputs $\underline{b}_m=b_{9m},\ldots,b_{9m+8}$ are multiplied by a polarity bit p_m to obtain code words $\underline{c}_m=c_{9m},\ldots,c_{9m+8}$. The polarity bit is determined by a polarity-bit generator (PBG) in such a manner as to 'break' the 'dangerous' patterns of Table 4.1 at the edges of code words. Since the inverse precoder is insensitive to word-wise polarity reversals, it will correctly recover the data bytes $\underline{a}_m$ irrespective of the precise polarity-bit sequence p_m. The structure of the inverse precoder is such that any single code digit affects at most two consecutive decoder outputs. It follows that error propagation

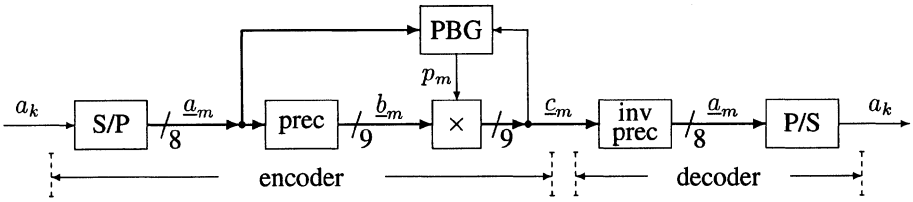


Figure 4.28: Anti-whistle polarity-bit encoder and decoder.

in the decoder is confined to at most two symbol intervals. Furthermore, errors do not propagate beyond byte-boundaries. Complexity of the PBG depends on the exact run-length constraints that are to be introduced and tends to be small. By way of illustration Fig. 4.29 depicts a PBG that consists of a single latch, three binary multipliers and an inverter. This generator is derived in [11] and restricts the run-length of all patterns in Table 4.1 to at most 14. Smaller values are possible at the cost of added complexity [11].

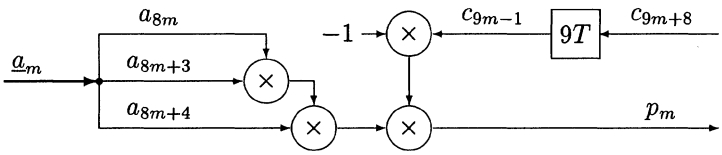


Figure 4.29: Polarity-bit generator that restricts the run-length of all patterns in Table 4.1 to at most 14.

4.9 MODULATION CODES WITH SPECTRAL ZEROS

Modulation codes can be used to create one or more spectral zeros in the transmitted data signal. The best known and most important example of this possibility is the class of DC-free codes. These codes lower baseline wander by imposing a charge constraint on the encoded data (see Section 4.3). Inherent to this constraint is that the transmitted signal cannot become DC-like for prolonged periods of time, and this is obviously beneficial for timing recovery (see Section 4.5). Many DC-free codes, in fact, have a strictly limited maximum run-length as in Section 4.8 (see Problem 4.14). An added advantage of the charge constraint is ease of performance monitoring. Violations of the constraint are readily detected in the receiver and provide a clear indication of transmission problems [28, 18]. It has more recently become apparent that the charge constraint can also be exploited towards a performance advantage on channels that are more or less matched to the code [121, 61]. This requires a comparatively complicated sequence de-

tector that keeps track of accumulated charge (see Section 4.9.6).[5] In similar vein, it is possible to create zeros at other frequencies (such as the Nyquist frequency $1/(2T)$) towards a performance advantage on channels that respond poorly to those frequencies. On top of performance advantages, spectral zeros at DC and/or the Nyquist frequency permit the addition of analog signals at either of these two frequencies for separate extraction in the receiver. Examples include power feeding over metallic transmission lines [28], combined transmission of digital data and analog speech signals across local telephone lines [115], servo signaling in digital recording systems [88, 97], and transmission of carrier or timing waves at the Nyquist frequency [91].

DC-free codes have long been used in cable transmission [1, 21, 20] and digital recording [44, 64, 83, 12], and a great variety of codes has evolved of which we shall present a representative sample. Zeros at frequencies other than DC are much less common and will receive only cursory attention. In Section 4.9.1 we introduce basic notions regarding DC-free codes. Capacity and spectra of these codes are discussed in Sections 4.9.2 and 4.9.3. Sections 4.9.4 and 4.9.5 extend the treatment to spectral zeros of arbitrary order at rational submultiples of the symbol frequency. Potential performance advantages of codes with spectral zeros are elaborated in Section 4.9.6 (see also Appendix 7A.1). After these preliminaries, Section 4.10 presents an overview of linear modulation codes with spectral zeros. This overview includes various examples of codes that have found practical application. A similar overview is provided for nonlinear codes in Section 4.11.

4.9.1 DC-free codes

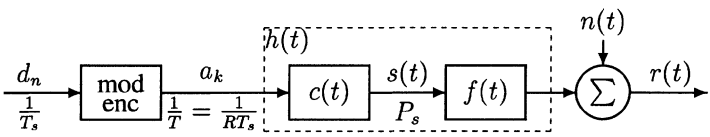


Figure 4.30: Cascade of encoder, modulator and channel.

For convenience we first recapitulate some of the notions of Section 3.2.4. In Fig. 4.30, a binary source data sequence d_n of data rate $1/T_s$ is encoded into a data sequence a_k of data rate $1/T = 1/(RT_s)$ where $1/T_s$ is the source data rate and R is the information rate of the modulation code. A linear pulse modulator with symbol response $c(t)$ transforms a_k into a transmitted signal $s(t)$ of constrained power P_s. The channel has impulse response $f(t)$ and additive noise $n(t)$; the cascade of modulator and channel has symbol response $h(t) \triangleq (c * f)(t)$.

We recall that the allocation of tasks between encoder and modulator is not necessarily unique. Specifically, one or more linear discrete-time operations may take place just before the conversion into continuous-time format, and these may be attributed to

[5]We shall see in Section 4.9.6 that linear modulation codes form a favorable exception to this rule.

either encoder or modulator. It is not uncommon to refer to a code as *linear* when its principal effect is due to a linear operation of this type [28, 29]. For linear modulation codes the distinction between the encoder and modulator is inherently vague, and we shall not attempt to be precise in what follows. It should be clear from the context which allocation we have in mind, and which signal is regarded as the output of the encoder.

Example 4.23 Bi-phase is a linear code with rate $R = 1/2$ or $R = 1$, depending on the point of view (see Example 3.1).

Both encoder and modulator can be used to obtain a DC-free transmitted signal $s(t)$. For the modulator this requires that $c(t)$ be DC-free, i.e. that the transfer function $C(\Omega)$ of the modulator vanishes at DC. In later chapters we will see that it can be advantageous to extend the modulator with a transmit filter in order to shape $c(t)$ in a desired fashion. A linear modulation code also acts to shape $c(t)$ and can as such be regarded as a special case of a transmit filter.

Example 4.24 When Bi-phase is regarded as having rate $R = 1$, its symbol response $c(t)$ is DC-free (see Fig. 3A.1.a). The spectrum of $s(t)$ is determined by the transfer magnitude $|C(\Omega)|$ of the modulator (Fig. 3A.1.b). It is DC-free and has maximum content at the signaling rate $1/T$. These spectral characteristics are well-matched, for example, to those of a Lorentzian channel at low densities. One might regard Bi-phase as a 'binary transmit filter' that achieves a net SNR advantage up to 4.77 dB over NRZ at these densities (see Appendices 3A and 3A.1).

DC in $s(t)$ can also be suppressed via the encoder. The modulation code is called *DC-free* (*charge-constrained, balanced*) if the encoder acts to constrain the *running digital sum* (RDS, *charge, unbalance*) $c_k \triangleq \sum_{i=0}^{k} a_k$ of the coded data symbols a_k between finite lower and upper limits C_l and C_u. The difference $C_u - C_l$ between the largest and smallest allowable charge is called the *digital sum variation* (DSV) [27, 28, 18].[6] The charge constraint is both necessary and sufficient for a_k to have a zero at DC [33], [124]. The effect of the constraint can be visualized in an *unbalance trellis diagram* that depicts the possible evolutions of charge c_k over time. Fig. 4.31 provides an illustration for binary data $a_k \in \{-1, 1\}$. Every allowable sequence a_k corresponds to a path in the trellis of Fig. 4.31 and vice versa. Specifically, upgoing and downgoing transitions in Fig. 4.31 have $a_k = +1$ and -1, respectively, i.e. $a_k = c_k - c_{k-1}$. Charge c_k is even at all even instants k, and odd at all odd instants. The zero at DC emerges by virtue of the finite height of the trellis. The DSV amounts to $C_u - C_l = 3 - (-3) = 6$.

Exercise 4.5 Consider Bi-phase as a modulation code with rate $R = 1/2$ as in Example 3.1. Construct the corresponding trellis diagram and show that Bi-Phase is DC-free. What is the DSV?

Exercise 4.6 Show that the transmitted signal $s(t)$ that arises with a DC-free code cannot have any content at the signaling rate $1/T$ and multiples thereof. Reconcile this fact with the observation that Bi-phase has maximum content at the signaling rate $1/T$ when viewed, as in Example 4.24, as having $R = 1$.

[6] This definition, while common, is not the only possible one. For example, DSV is sometimes defined as the total number of possible RDS values, i.e. as $C_u - C_l + 1$ [99, Chapter 6].

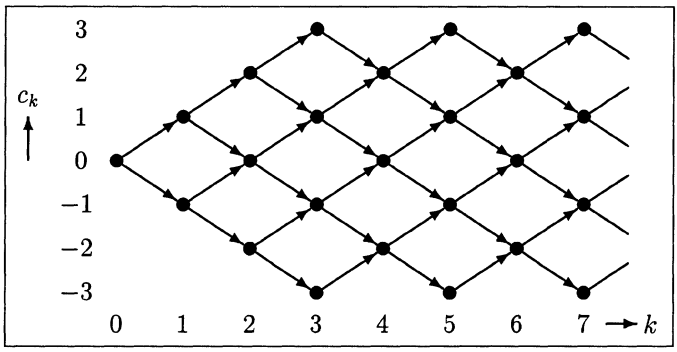


Figure 4.31: Unbalance trellis diagram of binary DC-free code with a digital sum variation (DSV) of 6. The initial charge c_0 is assumed to be zero.

For ternary DC-free codes, it is common practice to define the RDS and DSV relative to a data alphabet $\{-1, 0, 1\}$. This alphabet has a spacing of 1 between data levels, as opposed to the spacing of 2 that we have adopted by default in Chapter 3. DC-free codes with more than three data levels are of restricted practical interest.

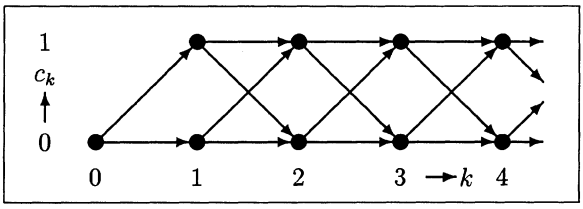


Figure 4.32: Unbalance trellis diagram of ternary DC-free code with a digital sum variation (DSV) of 1. The initial charge c_0 is assumed to be zero.

Example 4.25 In Section 4.10 we will encounter ternary codes in which nonzero symbols a_k are constrained to alternate in polarity. This gives rise to the unbalance trellis diagram of Fig. 4.32. Upgoing, horizontal and downgoing arrows correspond to digits $a_k = +1, 0$ and -1, respectively. There are two possible charges c_k that are arbitrarily taken to be 0 and 1. The DSV amounts to 1. In Chapter 7 we will see that the trellis diagram of Fig. 4.32 is central to Viterbi detection of Bipolar partial-response signals.

4.9.2 Capacity of DC-free codes

The capacity C of a DC-free code is the highest possible code rate R that can be achieved for a given DSV. Codes that achieve capacity are called *maxentropic*. Capacities of binary and ternary DC-free codes were first determined by Chien [24] and for

DSV$\geq$ 1 are given by

$$C_2 = \log_2 2\cos\frac{\pi}{\text{DSV}+2} \text{ bits per symbol interval} \tag{4.11}$$

and

$$C_3 = \log_2 2\cos[1+\frac{\pi}{\text{DSV}+2}] \text{ bits per symbol interval} \tag{4.12}$$

where the subscript denotes the cardinality of the data alphabet. Table 4.8 lists some numerical values.

DSV	1	2	3	4	5	6	7	8	9	∞
C_2	0	0.5	0.694	0.792	0.850	0.886	0.910	0.928	0.940	1
C_3	1	1.272	1.389	1.450	1.486	1.510	1.526	1.537	1.546	1.585

Table 4.8: Capacities C_2 and C_3 of binary and ternary DC-free codes for various values of the digital sum variation (DSV).

For binary codes, practical rates R compatible with DSVs of 2, 3, 4 and 5 are $R = 1/2, 2/3, 3/4$ and $4/5$, respectively. Ternary line codes are often of the 4B/3T type, i.e. blocks of 4 bits are translated into code words of 3 ternary digits. This yields a rate R of $4/3$ bits per symbol interval, and according to Table 4.8 DSVs can be as small as 3. By comparison, DSVs of popular 4B/3T codes are in the order of 5 to 7 (see Examples 4.37 and 4.38).

Exercise 4.7 Is Bi-phase maxentropic?

4.9.3 Spectra of maxentropic DC-free codes

The power spectral densities $\mathcal{A}(e^{j2\pi\Omega})$ of binary maxentropic DC-free codes were determined analytically for all even values of the DSV by Kerpez [62]. Expressions become quite complicated as DSV grows and for this reason we give only the formulas for DSV= 2, 4 and 6:

$$\mathcal{A}(e^{j2\pi\Omega}) = \begin{cases} 2\sin^2\pi\Omega & (\text{DSV}=2), \\ \frac{8}{3}\frac{7+6\cos 2\pi\Omega}{5-3\cos 4\pi\Omega}\sin^2\pi\Omega & (\text{DSV}=4), \\ 2\frac{47+33\sqrt{2}+(32\sqrt{2}+44)\cos 2\pi\Omega-\cos 4\pi\Omega}{16\sqrt{2}+23-(14\sqrt{2}+20)\cos 4\pi\Omega}\sin^2\pi\Omega & (\text{DSV}=6). \end{cases} \tag{4.13}$$

Fig. 4.33 provides an illustration.

The DSV= 2 spectrum is that of Bi-Phase. The width of the spectral notch, i.e. the frequency range across which $\mathcal{A}(e^{j2\pi\Omega})$ is small, is inversely proportional to DSV. A measure of this width is the *cut-off* frequency Ω_c, which is defined as the -3 dB point in Fig. 4.33. For large DSV it can be shown [100] that

$$\Omega_c \simeq \frac{6\ln 2}{\pi(\pi^2-6)}(1-C) \simeq \frac{1-C}{0.2923}. \tag{4.14}$$

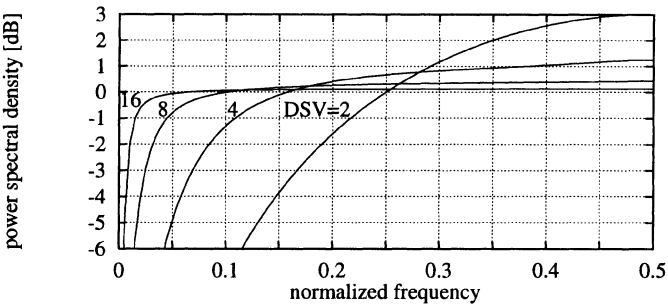


Figure 4.33: Power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of binary maxentropic DC-free codes for various values of the digital sum variation (DSV).

This expression relates Ω_c to the *redundancy* $1-\mathcal{C}$. To obtain a relation between Ω_c and DSV we make a series expansion for large DSV of (4.11) and use the result in (4.14). It follows that

$$\Omega_c \simeq \frac{2.44}{(\text{DSV}+2)^2}. \tag{4.15}$$

This approximation is accurate to within 0.5% for DSV$\geq$ 6.

Spectra of maxentropic DC-free codes for odd DSVs and ternary data alphabets are similar to those of Fig. 4.33 but are not known analytically. The only exception is the ternary code with DSV= 1, which is equivalent to the Twinned-Binary and AMI codes in Section 4.10.

4.9.4 Spectral zeros at rational submultiples of the symbol frequency

Spectral zeros can be created at any rational submultiple of the symbol rate $1/T$ by constraining a suitably modified version of the running digital sum. This is best illustrated with an example.

Example 4.26 Many channels respond poorly to high frequencies. For such channels it may be desirable for a_k to have a zero at the Nyquist frequency $1/(2T)$ ($\Omega = 0.5$). Consider a binary data sequence $a_k \in \{-1, 1\}$ of length N. This sequence has energy N, and therefore the average value of the amplitude spectrum $|A(e^{j2\pi\Omega})|^2$ of a_k must grow linearly with N. For large N the amplitude of a_k at the Nyquist frequency ($\Omega = 0.5$) will become negligible relative to the average amplitude if $A(e^{j\pi}) = \sum_{i=0}^{N-1} a_i(-1)^i$ does not grow with N. This suggests that we constrain the quantity $c_k \triangleq \sum_{i=0}^{k}(-1)^k a_k$ to be within within finite lower and upper limits C_l and C_u for any value of k. In [61] it is shown that this constraint will indeed deliver a zero at the Nyquist frequency, irrespective of the data alphabet.

The above reasoning is easily extended to obtain constraints that will produce spectral zeros at any rational submultiple of the symbol rate $1/T$ (see Problem 4.13 for a further illustration). In many instances these zeros do not come alone, i.e. a null at the

desired submultiple will automatically be accompanied by one or more nulls at other submultiples [61]. In practice constraints at DC and the Nyquist frequency are by far the most important choices, and for this reason we shall not dwell on the intricacies of constraints at other frequencies.

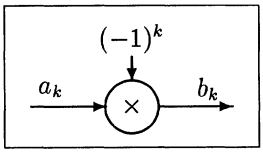


Figure 4.34: A simple modulation step suffices to transform a DC-free code into a code with a first-order zero at the Nyquist frequency.

Exercise 4.8 Consider the circuit of Fig. 4.34. Assume that the input sequence a_k is produced by a DC-free code. Show that the output b_k has a first-order zero at the Nyquist frequency.

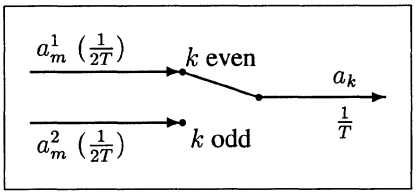


Figure 4.35: Interleaving of two DC-free sequences a_m^1 and a_m^2 of data rate $1/(2T)$ yields a sequence a_k of data rate $1/T$ with nulls at both DC and the Nyquist frequency.

It is sometimes desirable to obtain zeros at both DC and the Nyquist frequency. This is conveniently accomplished by interleaving two data sequences a_m^1 and a_m^2 of data rate $1/(2T)$ that are both DC-free (Fig. 4.35). The interleaved data sequence a_k satisfies

$$\sum_k a_k = \sum_m a_m^1 + \sum_m a_m^2 \tag{4.16}$$

and

$$\sum_k (-1)^k a_k = \sum_m a_m^1 - \sum_m a_m^2. \tag{4.17}$$

The sums on the right hand sides are bounded by virtue of the fact that a_m^1 and a_m^2 are both DC-free. The implication is that $\sum_k a_k$ and $\sum_k (-1)^k a_k$ are both bounded, i.e. that a_k must have a spectral null at both DC and the Nyquist frequency. Capacities of codes of this type were analyzed in [41].

4.9.5 Codes with higher-order spectral zeros

A code is said to have a Kth order zero at frequency Ω_0 if the power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of a_k and its first $2K-1$ derivatives all vanish at $\Omega = \Omega_0$. So far we have

restricted attention to first-order zeros. Higher-order zeros can be obtained by imposing additional constraints on a_k [61]. These constraints may yield added benefits in fields such as baseline wander and immunity to noise. On the debit side, they inevitably increase system complexity and decrease the rate R of the code. This decrement, in turn, may lower per-symbol signal-to-noise ratios to an extent that largely offsets any advantages. For these reasons only second-order zeros ($K = 2$) are of some practical interest, while orders $K \geq 3$ serve mainly academic purposes. Codes with a second-order zero at DC are sometimes called *DC^2-constrained* [96].

4.9.6 Reception techniques for codes with spectral zeros

Linear codes Here receivers based on partial-response techniques, decision feedback equalization or Viterbi detection can account fully for the combined effect of code and channel, irrespective of the precise DSV (see Chapters 6 and 7). Such receivers can be highly immune to baseline wander and it is, therefore, of restricted importance that the DSV be small (although it usually is). Performances may be assessed by regarding the symbol response $c(t)$ of the code as a part of the overall response of the system (see Chapters 5 to 7 and Section 3.11). It goes without saying that performance will benefit from a proper choice of $c(t)$ in relation to the parameters of channel and receiver. This may or may not favor the introduction of spectral zeros (see Appendix 3A for an illustration).

Nonlinear codes These codes have traditionally been used in conjunction with fairly simple receivers that account only for the effect of the channel. Such receivers tend to be sensitive to baseline wander and this favors a small DSV (see Example 4.8). We have mentioned earlier that it is, in principle, possible to exploit the spectral zeros of the code towards a performance advantage. This requires a relatively complicated receiver that keeps track, for example, of accumulated charge. By way of illustration we consider a binary DC-free code (see Fig. 4.31 for a corresponding trellis diagram). We recall that c_k denotes the charge of a_k, and that a_k is just the difference of successive charges ($a_k = c_k - c_{k-1}$). Conceptually, we may regard c_k rather than a_k as the transmitted data sequence. Optimum reception requires a maximum-likelihood receiver. In such a receiver, the received signal $r(t)$ of Fig. 4.30 is first applied to a whitened matched filter, whose output z_k may be described in terms of an equivalent discrete-time channel with impulse response f_k and additive white Gaussian noise n_k of variance 1 (see Section 3.10.2). A maximum-likelihood sequence detector (MLSD) operates on z_k in order to produce decisions $\hat{c}_k$ with respect to the charge sequence c_k. These are converted into bit decisions $\hat{a}_k$ via a difference operation. We arrive at Fig. 4.36.

The detector must distinguish the actually transmitted charge sequence c_k from other allowable charge sequences. Error performance of the detector is determined by the periods during which it fails to do this correctly. At the beginning of such an *error event*, the detected charge sequence $\hat{c}_k$ diverges from c_k, and at the end it remerges. The

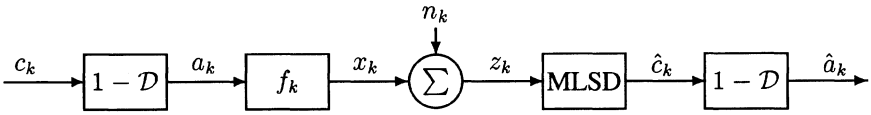
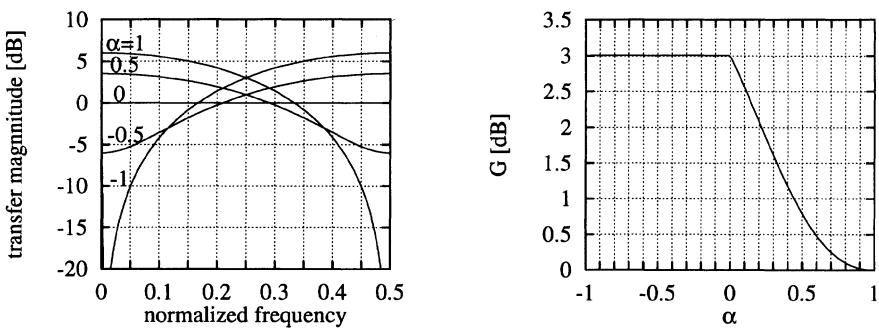


Figure 4.36: Equivalent discrete-time model of transmission system with charge-constrained modulation code and maximum-likelihood sequence detector (MLSD).

corresponding data sequence $\hat{a}_k$ will differ from a_k in at least two bit positions, and the error sequence $e_k = \frac{1}{2}(\hat{a}_k - a_k)$ will be DC-free. A direct consequence is that single bit-errors cannot occur, i.e. that performance is no longer governed by the matched-filter bound. This translates into a performance advantage for channels on which single bit-errors would otherwise prevail (see Section 3.11). The advantage depends on the characteristics of the channel in relation to those of the code and is generally largest when both have a zero at the same frequency. This is underscored in the designation 'matched spectral null' (MSN) that is used for certain codes with spectral zeros [61]. Even so, the channel does not need to have spectral zeros for an advantage to accrue.

Example 4.27 Consider an equivalent discrete-time channel without ISI ($f_k = \delta_k$). For this channel, the Euclidian weight of double bit-errors is twice as large as that of single bit-errors (see Section 3.11 for a definition of these terms). This corresponds to a gain of 3 dB in predetection signal-to-noise ratio relative to an uncoded system, where bit-by-bit detection would suffice. Appendix 7A.2 describes a simple two-state sequence detector, due to Wood [121], that achieves most of this gain.



a. Amplitude-frequency characteristics.

b. Gain G in predetection signal-to-noise ratio due to exploitation of the DC-constraint.

Figure 4.37: Two figures pertaining to first-order discrete-time channel with $f(\mathcal{D}) = 1 + \alpha\mathcal{D}$.

Example 4.28 Consider a first-order channel whose impulse response f_k has $\mathcal{D}$-transform $f(\mathcal{D}) = 1 + \alpha\mathcal{D}$. For positive α this is a low-pass channel, for negative α it is high-pass, and for $\alpha = 0$ it is all-pass (Fig. 4.37.a). Only for $\alpha = 0$ is there no ISI. MLSD performances

for a DC-free code on this channel are assessed in Exercise 7.6. Fig. 4.37.b. provides a graph of the attainable gain G in predetection signal-to-noise ratio for an MLSD that exploits both ISI and the DC-constraint relative to one that exploits only ISI. A gain of 3 dB is projected as long as the channel is high-pass or all-pass (i.e. for $\alpha \leq 0$). As α increases beyond 0, the gain decreases to vanish completely for $\alpha = 1$, i.e. for a low-pass channel of the Duobinary type.

These examples show that exploitation of a DC-constraint may be beneficial even if the channel does not have a zero at DC. A similar statement holds true for zeros at other frequencies. Performance merits of codes with spectral zeros were studied in [61] for a category of partial-response channels. Some of the main results of [61] are conveniently phrased in terms of the gain G of an MLSD that exploits both ISI and zero(s) of the code relative to one that only exploits ISI.

- For $f(\mathcal{D}) = 1 - \mathcal{D}$ (Bipolar) and $f(\mathcal{D}) = (1 - \mathcal{D})^2$, exploitation of a Kth-order zero at DC permits a gain $G \geq K + 1$. The actual gain can be strictly larger than this bound. For example, Bi-phase has $K = 1$ and achieves $G = 3$ on a Bipolar channel. This is 1.77 dB more than the gain that was projected in Example 4.28 for 'general' DC-free codes. The added gain accrues from additional code structure that is due to the linearity of Bi-phase.

- A gain $G \geq K+1$ is also achievable for $f(\mathcal{D}) = 1+\mathcal{D}$ (Duobinary) and $f(\mathcal{D}) = (1 + \mathcal{D})^2$ with a Kth-order zero at the Nyquist frequency, as well as for $f(\mathcal{D}) = 1 - \mathcal{D}^2$ (Class IV) with K^{th}-order zeros at both DC and the Nyquist frequency.

It is important to note that G does not account for the fact that creation of spectral zeros requires redundancy. To accommodate this redundancy it is necessary to increase the signaling rate $1/T$ relative to uncoded transmission, and this, in turn, affects per-symbol signal-to-noise ratios and the equivalent discrete-time system response f_k in Fig. 4.36. For a fair comparison we must use different responses f_k for the coded and uncoded systems [10]. In the literature on coding for partial-response channels it is not uncommon to assume that coding leaves the shape of f_k unaffected, and that the amplitude of f_k scales such that SNR $\propto R$, where R is the rate of the code [61, 106]. We have seen in Section 4.4 that these assumptions are, in many instances, overly optimistic and it is therefore crucial that redundancy be small if any net benefit is to remain. For DC-free codes this is, in principle, possible by allowing the DSV of the code to be large (see Section 4.9.2). With a large DSV, however, it will typically take a long time before the charge sequence c_k 'bumps' against the floor or ceiling of the trellis diagram, and the detector will have to defer its final decisions accordingly in order to benefit from the charge constraint (see Appendix 7A.3 for an illustration). This is obviously unattractive from an implementation standpoint.

In order to restrict detection delay to more manageable proportions it is possible to use a code with a *time-varying* trellis diagram. The key idea is to constrain the DSV to assume only one or at most a few possible values at given (and usually periodic) instants in time, while many DSV values are allowable in between. In this way the redundancy of the code can still be small, while the added constraints on the DSV force (or encourage) the detector to make early decisions. Cases in point include zero-disparity and

low-disparity codes (see Sections 4.11.2 and 4.11.3). Zero-disparity codes admit only a single DSV value at the end of each code word, and as a result the detection delay need not exceed the code word length. Low-disparity codes, by comparison, permit a small number of DSV values at the code word boundaries so that detection delay must, in general, be several times the code word length. Other examples of DC-free codes with a time-varying trellis diagram may be found in [42]. A sequence detector that is tailored to the latter codes is described in [93].

Soft-decision decoding It is worth mentioning that the code words of a zero-disparity code all have an even or odd Hamming weight, depending on whether the code word length is even or odd.[7] It is this property that induces added Euclidian distance. The same effect is obtained with a parity-check code. In such a code, a single parity-check bit is appended to each incoming data word such that each extended word has the same parity (either even or odd). Parity-check codes do not induce spectral zeros but the parity-check bit can nevertheless be exploited towards a performance advantage. A particularly simple technique involves the use of erasure flags along with bit flipping. Uncertain bit-decisions in each detected code word are flagged. At sufficiently high SNRs most detected words will not have any flag, and the ones that do will almost certainly have just one, provided that bit-errors do not occur in bursts (as they can, for example, in the decision feedback equalizer (see Chapter 6). If the parity check fails then the flagged bit -which is almost certainly present- is flipped so as to correct the detection error. This trick is easily amended towards modulation codes with spectral zeros. It is among the simplest examples of *soft-decision* decoding, in which side-information from the bit detector is used along with code redundancy in order to achieve performance benefits. More powerful -and inevitably more complicated- approaches include errors-and-erasures decoding of BCH codes [13, Section 9.2] and soft-decision decoding of BCH codes [106].

4.10 LINEAR MODULATION CODES

Having discussed some of the basic properties of codes with spectral zeros, we are now ready for a survey of various codes of practical interest. The distinction between linear and nonlinear codes will serve as a borderline in our review. For chronological reasons we treat linear codes first. By definition, the principal effect of these codes is due to a linear operation. The archetypal codes are fully linear and either binary or ternary (Fig. 4.38.a).

A source bit sequence $d_n \in \{-1, 1\}$ of sampling rate $1/T_s$ is upsampled by an integer factor L, i.e. $L-1$ zeros are inserted between successive bits. It is then applied to a discrete-time filter with impulse response w_k of sampling rate $L/T_s = 1/T$. The filter output is the encoded data sequence a_k. One commonly selects w_k such that a_k is

[7]The Hamming weight of binary code word is defined as the number of ones minus the number of digits of opposite polarity.

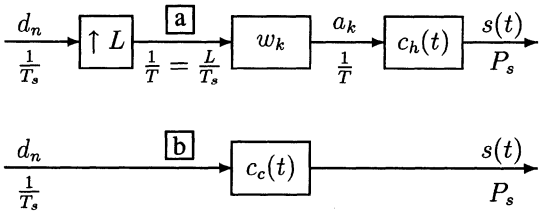


Figure 4.38: a. Transmitter with linear encoder and linear pulse modulator whose symbol response $c_h(t)$ is a hold function of width T. b. Equivalent linear pulse modulator with symbol response $c_c(t)$.

binary or ternary. A continuous-time transmitted signal $s(t)$ emerges via a hold function $c_h(t)$ of width T. Evidently a_k is a linear function of d_n. Because of this linearity we may express $s(t)$ as

$$s(t) = \sum_{n=-\infty}^{\infty} d_n c_c(t - nT_s) \tag{4.18}$$

where $c_c(t)$ is the compound symbol response of the cascade of encoder and modulator, given by

$$c_c(t) = \sum_{k=-\infty}^{\infty} w_k c_h(t - kT). \tag{4.19}$$

The implication is that we can generate $s(t)$ directly from d_n by means of a linear pulse modulator with symbol response $c_c(t)$ as in Fig. 4.38.b. A further implication is that the receiver can directly detect d_n with no need to first reconstruct a_k. For this reason the power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of a_k is of restricted interest but the symbol response $c_c(t)$ is all the more relevant to system performance.

Code linearity does not imply that a_k has spectral zeros and indeed there is no fundamental reason why linear modulation codes must exhibit such zeros. Even so, virtually all practical examples of these codes have a zero at DC, the Nyquist frequency, or both.

Example 4.29 Fig. 4.39 depicts symbol responses $c_c(t)$ of three linear DC-free codes. Bi-phase (also called *Diphase* or *WAL-1*) has $L = 2$ and was the subject of Examples 3.1 and 3.5. Bi-phase seems to originate in the paper of Sunde [110], who refers to it as dipulse transmission. Twinned Binary (also called *Dicode*) originates in cable transmission [81]. It has $L = 1$ and will resurface in the guise of the Bipolar $(1 - \mathcal{D})$ partial response in Chapter 6. Crank-shaft was also conceived as a line code [48]. It has $L = 4$ and is at the heart of the Quadra-Phase and Hedeman-2 codes [12, 122].

The symbol response of Bi-phase extends over one symbol interval T_s, i.e. successive pulses $d_n c_c(t - nT_s)$ do not overlap and the transmitted signal $s(t)$ is binary. For Twinned binary, by comparison, $c_c(t)$ extends across $2T_s$ seconds and 'controlled' ISI arises. As a result, $s(t)$ consists of rectangular pulses of width T_s and ternary amplitude $a_k \in \{-2, 0, 2\}$. Furthermore, nonzero symbols a_k alternate in polarity as in Example 4.25. For Crank-shaft, $c_c(t)$ is

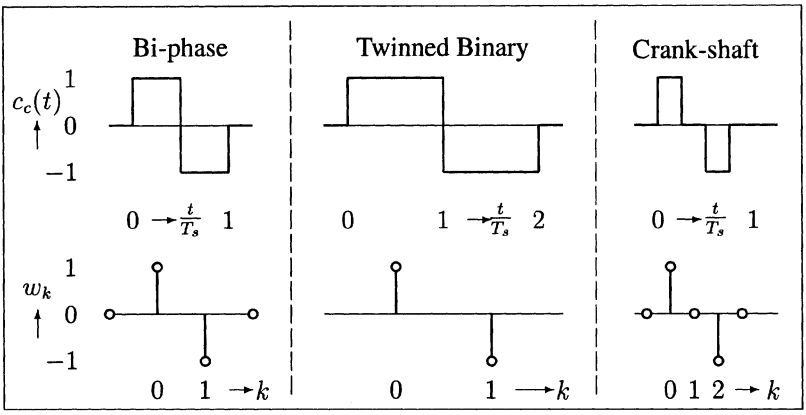


Figure 4.39: Symbol responses $c_c(t)$ of Bi-phase (a.), Twinned Binary (b.) and Crank-Shaft (c.). Corresponding impulse responses w_k are shown underneath.

only $3T_s/4$ seconds long and there is no ISI. Furthermore, $s(t) \in \{-1, 1\}$ in the first and third quarters of each symbol interval and $s(t) = 0$ elsewhere.

In effect, linear modulation codes act as a transmit-side equalizer and in digital recording they are sometimes referred to as write equalizers [95, 120]. Bi-phase is the simplest possible example and some published write equalizers are, in fact, just clever disguises of Bi-Phase [95, 54]. A derivation of all symbol responses w_k for which a_k is binary may be found in [54]. The effect of transmit-side equalizers is studied in Chapters 5 and 6.

Staggered linear codes The model of Fig. 4.38 does not cover all possible linear codes. An illustrative counterexample is Quadra-Phase (QP) [12]. This is essentially an interleaved version of Crank-Shaft. The transmitted signal $s(t)$ of Crank-Shaft is zero during the second and fourth quarters of every symbol interval T_s, and a second ('quadrature') Crank-Shaft signal that is delayed by $T_s/4$ seconds with respect to the first one can exactly fill these 'gaps' without interfering with the 'in-phase' signal [48]. The resulting QP signal is fully binary and conveys two bits of information every T_s seconds. QP is linear but the model of Fig. 4.38 does not apply because of the 'staggered' nature of the in-phase and quadrature components. QP has various advantages over Bi-Phase that are significant to magnetic recording [12, 7]. One of these is a narrower spectrum. For an honest comparison we must double the symbol duration of QP relative to Crank-Shaft. Then QP will convey one bit per T_s seconds, just as Bi-Phase. For uncorrelated data d_n, power spectral densities $P_s(\Omega)$ of $s(t)$ are identical to the amplitude spectra $|C_c(\Omega)|^2$ of $c_c(t)$. Fig. 4.40 provides a comparison that also includes NRZ. Because of its narrower spectrum, QP is better suited for high recording densities than Bi-Phase. Its principal advantage over NRZ is that it is DC-free.

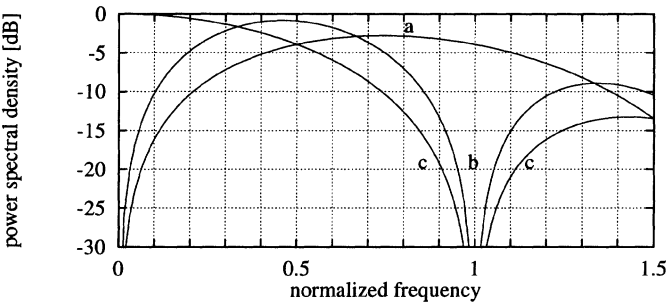


Figure 4.40: Power spectral densities $P_s(\Omega)$ of the transmitted signal $s(t)$ for three binary modulation codes. a. Bi-Phase; b. Quadra-Phase; c. NRZ.

Precoding The applicability of Twinned Binary is restricted by error propagation. We explain this problem with the help of Fig. 4.41.

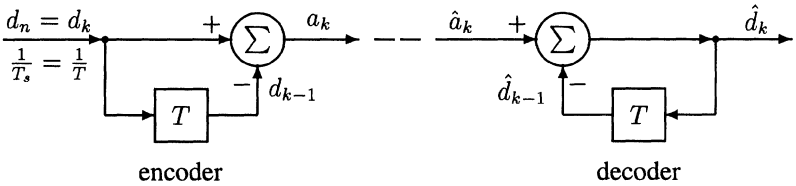


Figure 4.41: Cascade of Twinned-Binary encoder and recursive decoder.

The encoder delivers a ternary sequence $a_k = d_k - d_{k-1}$ that is detected in the receiver. If the previous source bit d_{k-1} is known then the current encoded symbol a_k fully determines the current source bit d_k according to $d_k = a_k + d_{k-1}$. This recursion, which is at the heart of the decoder of Fig. 4.41, describes a first-order pole at DC which compensates for the first-order zero that is introduced by the encoder. Occasionally the receiver makes erroneous decisions. Assume that a single bit-decision $\hat{d}_k$ is wrong while decisions $\hat{a}$ are all correct. Then all subsequent decisions $\hat{d}_{k+1}, \hat{d}_{k+2}, \ldots$ will also be wrong (in fact, they be outside the alphabet $\{-1, 1\}$ half of the time and this enables error monitoring and simple forms of error correction [59]). One way to avoid this *catastrophic error propagation* is to interweave the detection and decoding steps (see Chapters 6 and 7). Another possibility involves the use of a first-order *precoder*. This is a nonlinear circuit that transforms d_k into a binary sequence b_k according to the recursion $b_k = d_k b_{k-1}$. Since d_k and b_{k-1} are both in the alphabet $\{-1, 1\}$, their product b_k must also be in this alphabet.[8] The cascade of a first-order precoder and a Twinned-Binary encoder yields a so-called *AMI encoder* (Fig. 4.42).

[8]An actual implementation will normally use the logical alphabet $\{0, 1\}$, and this will transform the multiplication of d_k and b_{k-1} into a modulo-two addition, i.e. into an exclusive-or operation.

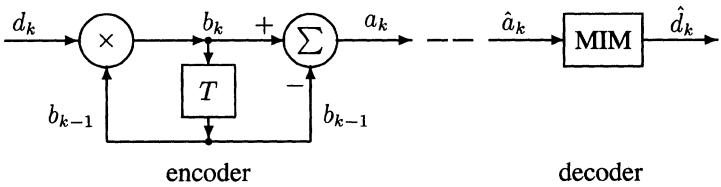


Figure 4.42: AMI encoder with memoryless decoder.

The acronym AMI stands for Alternate Mark Inversion and highlights the fact that nonzero symbols a_k alternate in polarity, just as for Twinned Binary. AMI (also called *Bipolar*) was devised by Barker [5] and has long been used in line transmission [1]. Its principal advantage over Twinned Binary is that decoding is possible with a memoryless inverse mapping (MIM) that totally avoids error propagation. To derive the decoding rule we observe from Fig. 4.42 that

$$a_k = b_k - b_{k-1} = (d_k - 1)b_{k-1}. \tag{4.20}$$

If $d_k = 1$ then $a_k = 0$ irrespective of b_{k-1}, while $a_k \in \{-2, 2\}$ otherwise. Thus an AMI signal may be generated by transmitting a 'space' (no pulse) when $d_k = 1$ and a 'mark' (pulse) of alternating polarity when $d_k = -1$. The decoder just observes whether or not a space is received and sets $\hat{d}_k = +1$ or -1 accordingly. This rule is obviously memoryless and therefore no error propagation can occur. For uncorrelated binary data d_k with zero mean it can be shown that the precoded data sequence b_k is also uncorrelated and has zero mean (Problem 4.15). The transmitted signal $s(t)$ then has the same spectrum as for Twinned Binary. Because of this 'spectral neutrality', the principal effect of the encoder is due to the linear (Twinned-Binary) portion of the encoder. For this reason it is common practice to call AMI a linear code.

A disadvantage of AMI is that the line signal can exhibit long runs of zeros (see also Problem 4.16). Such strings may hamper timing recovery and automatic gain control. *Modified AMI* codes are sometimes used to overcome this problem. These codes do not strictly exhibit alternate mark inversion but each block of n successive zeros is replaced by a sequence that violates the AMI rule. Well-known examples are the B6ZS and HDB3 codes [18].

Pseudoternary codes The ternary data signal produced by AMI is really a binary signal in a 'filtered' disguise. For this reason AMI is sometimes said to be a *pseudoternary* code. There are several other such codes.

Example 4.30 Duobinary is a pseudoternary code with a zero at the Nyquist frequency $1/(2T)$ that was invented by Lender [70]. It is obtained if we replace the subtraction in Fig. 4.42 by an addition. This yields an impulse response w_k with $\mathcal{D}$−transform $1 + \mathcal{D}$. Modified Duobinary was invented by van Gerwen [47] and is better known as *Class IV partial-response* after the catalog of Kretzmer [66]. It emerges if we double the delay in Fig. 4.42 to $2T$ seconds; this is

equivalent to putting $w(\mathcal{D}) = 1 - \mathcal{D}^2$. The result is a ternary transmitted signal $s(t)$ with a zero at both DC and the Nyquist frequency (Fig. 4.43).

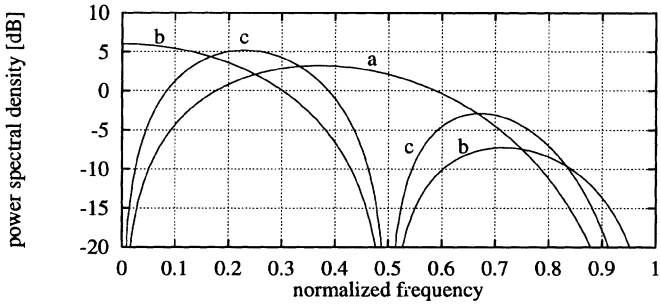


Figure 4.43: Power spectral densities of ternary partial-response codes. a. AMI, Twinned Binary $(1-\mathcal{D})$; b. Duobinary $(1+\mathcal{D})$; c. Modified Duobinary or Class IV partial response $(1-\mathcal{D}^2)$.

Duobinary is evidently well suited for low-pass channels but is not DC-free. Class IV has a narrower spectrum than AMI and is still DC-free. All three codes of Fig. 4.43 can be regarded as forms of *partial-response signaling*. Partial-response techniques are discussed in Chapter 6.

Exercise 4.9 Determine the compound symbol responses $c_c(t)$ of Duobinary and Modified Duobinary. Show that the MIM of Fig. 4.42 continues to work properly for Modified Duobinary and that it requires a minor modification for Duobinary.

Other applications of precoding It may seem from the above development that precoding is only useful in conjunction with codes that introduce ISI. This is, however, not quite true. Error propagation is indeed not an issue for codes without ISI but precoding may have desirable effects in other fields.

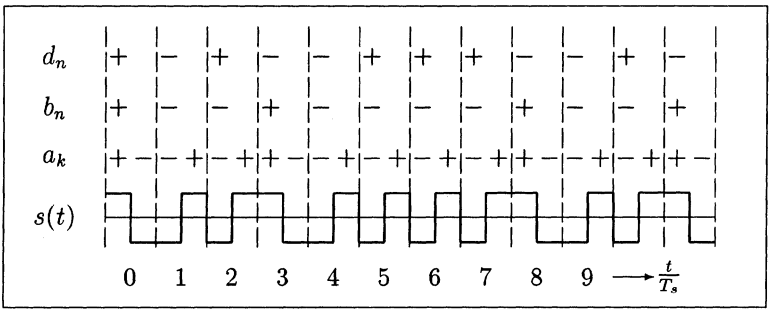


Figure 4.44: Example sequences and waveforms for precoded Bi-phase; '−' and '+' are shorthand for −1 and +1, respectively.

Example 4.31 Bi-phase does not introduce ISI yet a first-order precoder is sometimes added. Fig. 4.44 illustrates the encoding process. The precoded sequence b_n exhibits a transition whenever $d_n = -1$. The line signal $s(t)$ is obtained from b_n via a Bi-phase mapping and a hold operation of width $T = T_s/2$. It can be observed that $s(t)$ alternates, depending on the polarity of d_n, between square-wave segments of frequency $1/T_s$ and $1/(2T_s)$. Thus precoded Bi-phase is similar to analog frequency modulation and is, in fact, often called FM in recording jargon. On top of this aesthetic advantage, precoding offers the more tangible benefit of permitting certain detection techniques [114, 8]. We have already encountered FM in Example 4.19. Other designations include Frequency Doubling, Conditioned Diphase and Two-Level AMI Class II coding [29]. The general class of Bi-phase-like codes (either precoded or non-precoded) is sometimes known as Manchester.

4.11 NONLINEAR CODES WITH SPECTRAL ZEROS

This section provides a survey of nonlinear modulation codes with spectral zeros. Both binary and ternary codes are considered; typical application areas are digital recording and line transmission, respectively. The survey is more or less chronological and covers four basic varieties: polarity-bit codes, zero-disparity codes, low-disparity (bi-mode) codes and multi-mode codes. Some codes have a zero at the Nyquist frequency, and the large majority is DC-free.

4.11.1 Polarity-bit codes

Here incoming bits are collected into groups of $n - 1$ bits, which are sent either as they are or all inverted (i.e. multiplied by -1). Whether or not the bits have been inverted is communicated by means of a *polarity bit* at the end of the block. The polarity of the code word is chosen to make the digital sum variation as small as possible. This technique was invented (in analog form) by Bowers [15]. A digital version was first described by Carter [20]. The technique can be extended to punch a hole anywhere in the spectrum by basing the choice of polarity bits on a Fourier analysis [40]. A similar extension permits the generation of embedded pilot tones for servo purposes [97, 60].

Main attraction of the polarity-bit code is its simplicity. No look-up tables are required for encoding or decoding. The encoder contains a counter that keeps track of the running digital sum and passes or inverts groups of $n - 1$ bits conditioned upon the RDS. The decoder just passes or inverts groups of $n - 1$ bits depending on the polarity bit. A word synchronization mechanism is needed to demarcate the boundaries of code words. Erroneous detection of the polarity bit causes a complete group of $n-1$ data bits to be misinterpreted. Decision errors at other bit-positions do not propagate. This non-uniform error-propagation behavior can be uniformized via the word-wise precoder and inverse precoder of Section 4.8.8.

The polarity-bit code has rate $R = 1 - \frac{1}{n}$. The RDS trellis is periodic with a period of 1 code word when n is even and 2 code words when n is odd. The RDS cannot

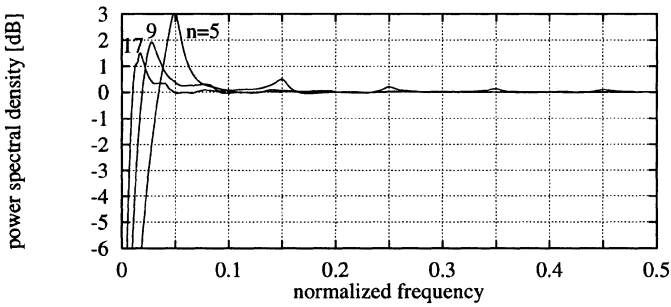


Figure 4.45: Approximate power spectral densities of polarity-bit codes of various word-lengths n. After Greenstein [50].

exceed $2n$ at the edges of code words, and $3n$ or $3n-1$ in the middle of code words according as n is even or odd [50]. An analytical approximation to the power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ is due to Greenstein [50] and serves as a basis for Fig. 4.45. For even n it can be shown that the normalized cut-off frequency Ω_c is approximately equal to

$$\Omega_c \simeq \frac{3}{4\pi}\frac{1}{2n-1} = \frac{3}{4\pi}\frac{1-R}{2+(1-R)}. \tag{4.21}$$

For rates R close to 1 (i.e. for large n) we have

$$\Omega_c \simeq \frac{3}{8\pi}(1-R) \simeq \frac{1-R}{8.378}. \tag{4.22}$$

Comparison of this expression with (4.14) reveals that the cut-off frequency that is obtained for a given redundancy $1-R$ is almost a factor 3 smaller than for maxentropic codes. The simplicity of polarity-bit codes, therefore, commands a premium in the form of a relatively poor suppression of low frequencies.

Improved suppression of low frequencies requires the flexibility to transform each incoming data word into more than two distinct code words. This is possible via a technique that is called *guided scrambling* [32]. Here each incoming data word is preceded by L bits $b_0, \ldots, b_{L-1}$ before being scrambled by a self-synchronizing scrambler. Distinct bit-combinations $b_0, \ldots, b_{L-1}$ yield distinct scrambled words. All 2^L combinations are tried and the scrambled word that minimizes the DSV is transmitted. Recovery of the data word requires a descrambler. Error propagation is confined to a single code word because scrambler and descrambler are re-initialized after each word. Complexity of the approach increases exponentially with L so that small values of L (notably $L=2$) are most attractive.

4.11.2 Zero-disparity codes

In these codes, code words have a fixed length n and are constrained to have zero disparity. The *disparity* or *unbalance* of a code word is the surplus number of ones

relative to minus-ones in the word. A disparity of zero means that there are equally many ones and minus-ones and for binary codes this is clearly only possible if n is even. The unbalance trellis diagram of zero-disparity codes admits only a single charge ($c_k = 0$) at the end of every code word ($k = 0, n, 2n, \ldots$).

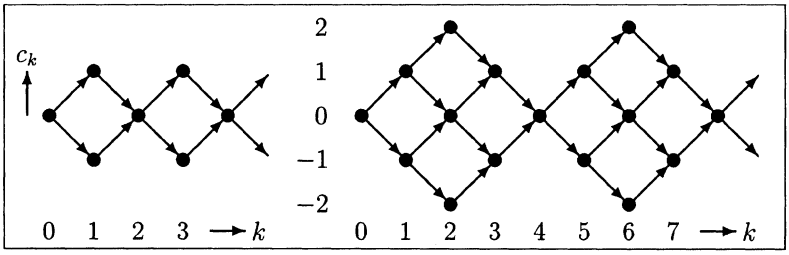


Figure 4.46: Unbalance trellis diagrams of Bi-Phase and FM (left) and Quadra-Phase (right).

Example 4.32 Bi-phase and its precoded version FM (see Section 4.10) may both be regarded as zero-disparity codes with rate $R = 1/2$ and word length $n = 2$. Quadra-Phase (see Section 4.10) may be viewed as a zero-disparity code with $R = 1/2$ and $n = 4$. Unbalance trellis diagrams of both codes are depicted in Fig. 4.46. DSVs amount to 2 and 4, respectively, and unbalances c_k are zero at the edges of code words.

Zero-disparity codes were introduced by Cattermole [21], although a few simple earlier examples do exist (notably Bi-phase and FM [44]). They are almost always binary (among the exceptions is [84]). For binary data and even n, the number N_0 of distinct zero-disparity code words of length n is given by the binomial coefficient

$$N_0 = \binom{n}{n/2} = \frac{n!}{(\frac{n}{2}!)^2}. \tag{4.23}$$

These N_0 words can be distinguished with a total of $\log_2 N_0$ bits, i.e. the highest possible rate $\hat{R}$ of a zero-disparity code is

$$\hat{R} = \frac{1}{n}\log_2 N_0 = \frac{1}{n}\log_2 \binom{n}{n/2}. \tag{4.24}$$

Unfortunately, N_0 is not normally a power of 2 and it is therefore desirable to drop some zero-disparity words from the code book in order to arrive at a practical coding arrangement.

Example 4.33 For $n = 8$ we have $N_0 = 70$. If we drop 6 zero-disparity words then we are left with a pruned code book of $2^6 = 64$ words. This would permit an encoder in the form of a look-up table that translates incoming words of 6 bits into 8-bit-wide code words. The code has rate $R = 6/8 = 0.75$, as opposed to $\hat{R} = 0.766$.

For practical values of n, look-up tables for encoding and decoding tend to become quite large. Hardware complexity can then be lowered significantly by resorting to enumerative techniques for encoding and decoding [73].

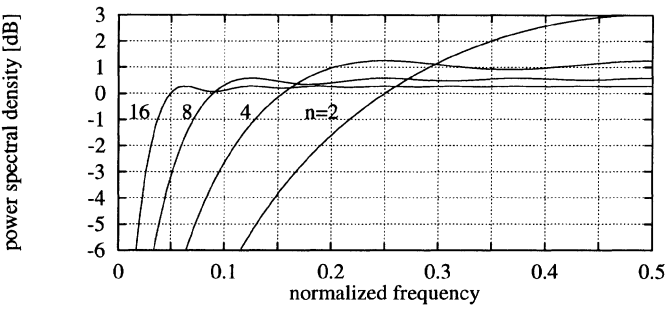


Figure 4.47: Power spectral densities of full-set zero-disparity codes of various word-lengths n.

The power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of zero-disparity codes with a full code-book was first determined by Franklin and Pierce [40] and is given by

$$\mathcal{A}(e^{j2\pi\Omega}) = \frac{n}{n-1}[1 - (\frac{\sin \pi n\Omega}{n \sin \pi\Omega})^2]. \tag{4.25}$$

Fig. 4.47 provides an illustration. The spectra of practical codes with a pruned code-book normally match those of Fig. 4.47 quite well. For large n the spectrum is essentially flat with magnitude $n/(n-1)$ until it drops sharply at the lower end, with a cut-off frequency $\Omega_c \simeq 1/(2.265n)$. This approximation, together with (4.24), allows us to assess the cut-off frequency Ω_c that is obtained for a given redundancy $1 - \hat{R}$. It is found that Ω_c is generally larger than for polarity-bit codes (eq. (4.22)) but strictly smaller than for maxentropic codes (eq. (4.14)). A quantitative comparison will be made in Section 4.11.3 (Fig. 4.49).

4.11.3 Low-disparity codes

These codes also have words of a fixed length n, but code words of low disparity are allowed on top of words with zero disparity. The latter words are uniquely assigned to incoming data words, while words with nonzero disparity are selected in pairs of opposite polarity and are distributed across two encoding tables according to the sign of their disparity. During transmission, the encoder switches between both tables in such a manner as to keep the running digital sum (i.e. the accumulated disparity) as small as possible. Thus the encoder can be in one of two modes and this explains the designation *bi-mode codes* that is sometimes used for low-disparity codes. The decoder is mode-independent and needs no side-information from past or future code words to decode the current word. Thus error propagation cannot occur. Binary low-disparity codes are due to Cattermole [21] and Griffiths [51]. Ternary low-disparity codes were first described by Sipress [108].

Example 4.34 Pair-Selected Ternary (PST) [108] is a ternary low-disparity code whose coding rules are summarized in Table 4.9.

Binary Word	Code word Mode A	Mode B	Disparity
$--$	$-+$	$-+$	0
$-+$	$0+$	$0-$	± 1
$+-$	$+0$	-0	± 1
$++$	$+-$	$+-$	0

Table 4.9: Encoding rules of Pair-Selected Ternary (PST).

Incoming data words are grouped into pairs. The pairs $--$ and $++$ are both mapped into a zero-disparity code word independently of the mode of the encoder. The other two pairs $-+$ and $+-$ are mapped into code words with disparity 1 or -1 according to the mode of the encoder, which alternates upon transmission of these words. PST has rate $R = 1$, digital sum variation DSV= 3 and an efficiency R/C_3 of 72% (compare Table 4.8). The code is somewhat akin to AMI, which also has $R = 1$ yet DSV= 1 (see Section 4.10). An advantage of PST over AMI is that the maximum run-length of zeros is only two symbol intervals, and this facilitates automatic gain control and timing recovery. A disadvantage is that PST is nonlinear. This complicates reception and may lead to inferior bit-error rates.

PST is an example of a *unit-disparity* code. This subclass of low-disparity codes is characterized in that only code words with disparity -1, 0 and 1 are permissible. For binary codes with even length n the disparity of code words is necessarily even, and the code is still said to be of the unit-disparity type if only words with disparity -2, 0 and $+2$ are used. The power spectral density of binary 'full-set' unit-disparity codes may be determined analytically for even n [100] and is given by

$$\mathcal{A}(e^{j2\pi\Omega}) = \frac{n+2}{n+1}[1 - \frac{n^2}{1 - 2(n+1)\cos(2\pi n\Omega) + (n+1)^2}(\frac{\sin \pi n\Omega}{n \sin \pi\Omega})^2]. \quad (4.26)$$

Fig. 4.48 provides an illustration. For a given value of n, spectral notches are all narrower than for zero-disparity codes, yet code rates $\hat{R}$ are larger by virtue of the fact that there are more code words available. Specifically, the total number of code words of unit disparity amounts to $N_1 = \binom{n}{1 + n/2}$ and these words are added in one of both encoding tables on top of the N_0 words of zero disparity. The other table includes $N_{-1} = N_1$ words of disparity -1 and also contains $N_0 + N_1$ words. Thus

$$\hat{R} = \frac{1}{n}\log_2[N_0 + N_1] = \frac{1}{n}\log_2[\binom{n}{\frac{n}{2}} + \binom{n}{\frac{n}{2}+1}]. \quad (4.27)$$

Just as for zero-disparity codes, pruned code-books are commonly used in order to arrive at a practical encoding arrangement. The resulting spectra tend to match those of Fig. 4.48 quite well.

Example 4.35 Digital magnetic recording systems sometimes employ a binary unit-disparity code that translates incoming bytes into code words of 10 bits [43, 113]. Generic designations

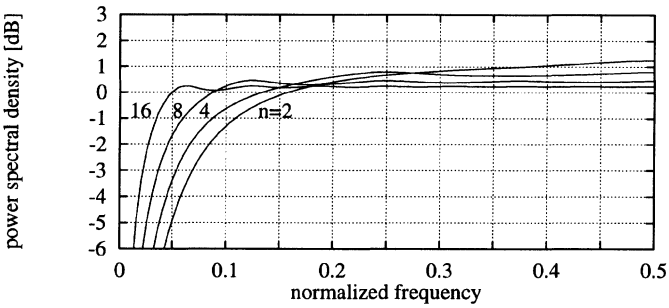


Figure 4.48: Power spectral densities of binary full-set unit-disparity codes of various word-lengths n.

for codes of this kind include 8/10, 8-to-10 and 8b10b (8-bits-to-10-bits). A zero-disparity 8-to-10 code is clearly impossible in view of the fact that there are only $N_0 = 252 < 2^8 = 256$ zero-disparity words of length 10 (see (4.23)). A bi-mode unit-disparity code of length 10 has at most $N_0 + N_1 = 252 + 210 = 462$ code words and this is much more than the 256 words that can be addressed with one byte. There is, therefore, considerable freedom in the construction of the encoding tables and this freedom can be used, among other purposes, to minimize the DSV or to incorporate maximum run-length constraints. Many 8b10b codes have been described in the literature. A comparison is beyond the scope of this text but may be found, for example, in [113].

The cut-off frequency Ω_c that full-set unit-disparity codes achieve for a given redundancy $1 - \hat{R}$ is compared in Fig. 4.49 with that of other DC-free codes. Polarity-bit codes lag furthest behind maxentropic codes. Unit-disparity codes clearly outperform zero-disparity codes and closely approach maxentropic codes for redundancies in excess of about 10%.

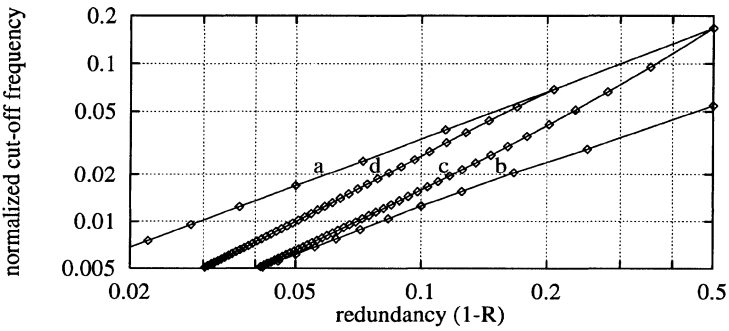


Figure 4.49: Cut-off frequency Ω_c versus redundancy $1 - R$ for various binary DC-free codes: maxentropic (a.), polarity-bit (b.), full-set zero-disparity (c.) and full-set unit-disparity (d.).

A straightforward extension of the above development leads to bi-mode codes whose disparity has a magnitude of at most K ($K \geq 1$). For binary codes of this type it is shown in Problem 4.22 that the digital sum variation is bounded by DSV$\leq 2K+n-2$ provided that n is even. We recall that K is necessarily even in this case. Only the case $K = 2$ (i.e. unit-disparity codes) is of practical interest. Here DSV$\leq n+2$. For ternary codes we have

$$\text{DSV} \leq 2(K + (n \text{ div } 2)) - 1. \tag{4.28}$$

This bound is valid for all integers $n, k \geq 1$ (see Problem 4.22). Unlike binary codes, several values of K are of practical interest.

Example 4.36 PST has $K = 1, n = 2$ and DSV$= 3$ (see Example 4.34) and thus meets (4.28) with equality.

Example 4.37 Codes that translate groups of 4 Bits into code words of 3 Ternary digits are collectively referred to as 4B/3T. A well known-example of a 4B/3T-code is 4B-3T [28, 18] (note the subtle difference in nomenclature). This is a low-disparity code with $K = 3$ that obeys the encoding rules of Table 4.10.

Binary Word	Code word Mode A	Code word Mode B	Dis-parity	Binary Word	Code word Mode A	Code word Mode B	Dis-parity
$----$	$+0-$	$+0-$	0	$+---$	$++-$	$--+$	±1
$---+$	$-+0$	$-+0$	0	$+--+$	$-++$	$+--$	±1
$--+-$	$0-+$	$0-+$	0	$+-+-$	$+-+$	$-++$	±1
$--++$	$+-0$	$+-0$	0	$+-++$	$+00$	-00	±1
$-+--$	$++0$	$--0$	±2	$++--$	$0+0$	$0-0$	±1
$-+-+$	$0++$	$0--$	±2	$++-+$	$00+$	$00-$	±1
$-++-$	$+0+$	$-0-$	±2	$+++-$	$0+-$	$0+-$	0
$-+++$	$+++$	$---$	±3	$++++$	$-0+$	$-0+$	0

Table 4.10: Encoding rules of 4B-3T.

The encoding table for Mode A contains words with zero and positive disparity, and for Mode B it contains words of zero and negative disparity. The running digital sum at the end of a code word is -3, -2 or -1 in Mode A, and 0, 1 or 2 at the end of Mode B. The DSV amounts to 7 and this is the largest value permitted by (4.28). In Example 4.38 we will encounter 4B/3T-codes that stays well within this bound. There we will also consider the spectrum of 4B-3T.

4.11.4 Multi-mode codes

In the nomenclature of Section 4.7, low-disparity codes have a two-state encoder and a sliding-block decoder with no memory and no anticipation. A logical extension leads to codes that require encoders with more than two states or *modes*, along with sliding-block decoders that may have memory and/or anticipation. Ternary codes of this type have long been known [33, 18]. Binary codes have appeared more recently (an early example is [82]), but their design has attained a high level of mathematical

rigor. We shall not attempt to provide an in-depth treatment of the design techniques that underly these codes, but rather call to attention the paper of Karabed and Siegel [61] which may serve as an introduction to this subject. Here we shall merely present three examples that are meant to illustrate some typical codes and their properties.

Example 4.38 MS43 is a 4B/3T-code that was devised by Franaszek as an improvement of 4B-3T [33]. The code has 3 modes (Table 4.11). Just as in unit-disparity codes, switching between modes is controlled by the running digital sum (RDS) at the end of each code word. Specifically, Mode A is chosen if RDS= -1, Mode B if RDS= 0 or 1, and Mode C if RDS= 2. Throughout transmission the DSV is kept less than or equal to 5, as opposed to 7 for 4B-3T.

Binary Word	Code word Mode A	Mode B	Mode C	Binary Word	Code word Mode A	Mode B	Mode C
$----$	$+++$	$+-+$	$-+-$	$+---$	$+-+$	$+-+$	$---$
$---+$	$++0$	$00-$	$00-$	$+--+$	$00+$	$00+$	$--0$
$--+-$	$+0+$	$0-0$	$0-0$	$+-+-$	$0+0$	$0+0$	$-0-$
$--++$	$0-+$	$0-+$	$0-+$	$+-++$	$0+-$	$0+-$	$0+-$
$-+--$	$0++$	-00	-00	$++--$	$+00$	$+00$	$0--$
$-+-+$	$-0+$	$-0+$	$-0+$	$++-+$	$+0-$	$+0-$	$+0-$
$-++-$	$-+0$	$-+0$	$-+0$	$+++-$	$+-0$	$+-0$	$+-0$
$-+++$	$-++$	$-++$	$--+$	$++++$	$++-$	$+--$	$+--$

Table 4.11: Encoding rules of MS43.

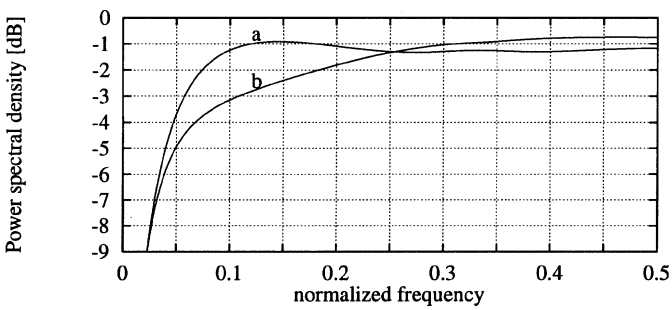


Figure 4.50: Power spectral densities of 4B-3T (a.) and MS43 (b.).

All code words uniquely identify a 4−bit data word, i.e. the decoder is state-independent with no memory and no anticipation. Because of its smaller DSV, MS43 has a better rejection of low frequencies than 4B-3T (Fig. 4.50). Formulas for the power spectral densities of 4B-3T and MS43 may be found in [18]. That paper also provides a detailed comparison of these and other 4B/3T codes.

Example 4.39 The binary MFM code of Section 4.8.4, contrived by A. Miller, is run-length-limited with rate $R = 1/2$ and run-length parameters $(d, k) = (1, 3)$. MFM is not DC-free

but has been modified to that effect by J.W. Miller [83]. The resulting code has been adopted as a standard in certain magnetic tape recording systems [17]. It has the same rate and run-length parameters as MFM and is commonly known as Miller-squared (M^2). The encoder can be described with the state diagram of Fig. 4.51 [121].

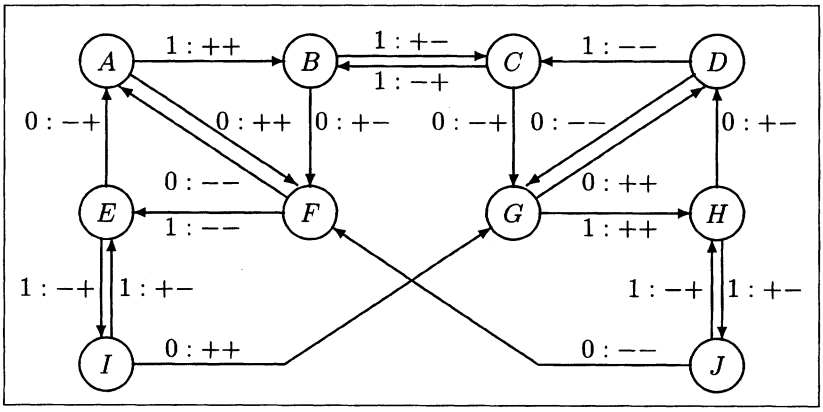


Figure 4.51: State diagram of M^2 code. Annotation along transitions is of the form a_n : $b_{2n}b_{2n+1}$, with + and − denoting +1 and −1, respectively.

Each incoming bit a_n is mapped into a pair of encoded digits (b_k, b_{k+1}) with $k = 2n$. The ten encoder states can be grouped into three subsets $S_1 = \{A, E, I\}$, $S_2 = \{B, C, F, G\}$ and $S_3 = \{D, H, J\}$ according to the possible values of the running digital sum at the end of each encoding step. If we denote the respective values by s_1, s_2 and s_3 then $s_1 = s_2 - 2$ and $s_3 = s_2 + 2$. The DSV is determined by the running digital sums in the middle of encoding steps (i.e. including b_k but excluding b_{k+1}) and amounts to DSV= $(s_3 + 1) - (s_1 - 1) = 6$. The unbalance trellis diagram is that of Fig. 4.31. It can be verified by inspection from Fig. 4.51 that the minimum and maximum-runlengths amount to $d+1 = 2$ and $k+1 = 4$. The power spectral density of M^2 was computed in [71] and resembles the $(1, 3)$ spectrum of Fig 4.24 except for a notch near DC. Decoding involves a simple sliding-block decoder with an anticipation of 2 code words and no memory [121].

Miller-squared is an example of a code that is both charge-constrained and run-length-limited. An analysis of such codes may be found in [85] and [63]. Other examples include Zero-Modulation [87], EFM (Eight-to-Fourteen Modulation; used in Compact Disc [88]) and EFMPlus (used in DVD) [103].

Example 4.40 Even Mark Modulation (EMM) is a binary code with rate $R = 2/3$ that produces a spectral zero at the Nyquist frequency [61]. Incoming bits a_n are grouped into pairs $\underline{a}_m = (a_{2m}, a_{2m+1})$ and mapped into code words $\underline{b}_m = (b_{2m}, b_{2m+1}, b_{2m+2})$ of three bits. The code has 8 states $S_m \in \{A, B, \ldots, H\}$. The incoming bit pair $\underline{a}_m$ and the current state S_m together determine the code word $\underline{b}_m$ and the next state S_{m+1} according to Table 4.12.

An interesting feature of EMM is that it incorporates asymmetrical run-length constraints. Specifically, consecutive marks ($b_n = 1$) are constrained to occur in even numbers, while spaces ($b_n = -1$) have no minimum run-length. This asymmetrical constraint is thought to

S_m	$\underline{b}_m/S_{m+1}$			
	$\underline{a}_m = --$	$\underline{a}_m = -+$	$\underline{a}_m = +-$	$\underline{a}_m = ++$
A	$-++/A$	$-++/B$	$++-/A$	$++-/B$
B	$--+/E$	$--+/F$	$++-/C$	$-++/G$
C	$---/A$	$---/D$	$+++/E$	$+++/F$
D	$--+/E$	$--+/F$	$+++/E$	$+++/F$
E	$+--/A$	$+--/B$	$+-+/E$	$+-+/F$
F	$+++/A$	$+++/B$	$+--/C$	$+++/H$
G	$---/A$	$---/B$	$+++/E$	$+++/F$
H	$---/A$	$---/B$	$+++/E$	$---/C$

Table 4.12: Encoding rules of EMM.

be of interest for certain optical recording systems [61, 72]. A further asymmetry is in the maximum run-length constraints, which amount to $k = 11$ for marks and $k = 7$ for spaces. The power spectral density of EMM may be determined analytically and is given by $\mathcal{A}(e^{j2\pi\Omega}) = 8[1 + \cos 2\pi\Omega]/[\sqrt{5}(3 + 2\cos 2\pi\Omega)]$. The spectrum declines gradually from DC on and has a first-order zero at the Nyquist frequency $(1/(2T))$. For this reason EMM is best suited for low-pass channels as encountered, for example, in digital optical recording. EMM may be decoded with a sliding-block decoder that has a decoding window of 12 bits. This implies that a single code-bit error can propagate to at most eight data bits a_k. This is attractive in view of the byte orientation of commonly employed error-correction codes.

4.12 FURTHER READING

Techniques for the design and spectral analysis of modulation codes have matured over a period of several decades. For code design we draw attention to the classical *Bounded-Delay* (BD) design method that was developed by Franaszek in a sequence of papers [36]-[39]. This method constructs codes that employ *look-ahead encoding* (or, equivalently, introduce an encoding delay) and aims at minimizing the decoding window. A drawback is that code construction may involve large and complicated search problems. The *state-splitting* algorithm of Adler, Coppersmith and Hassner [3], based on work of Marcus [75], has brought code design a new level of mathematical rigor and generality. This algorithm can be used to design codes at any rate R that does not exceed capacity (see [78] for an excellent exposition). The algorithm has low complexity but allows much freedom of choice and it is not always clear how this freedom can be used to obtain codes with a small decoding window. Both the BD-method and the state-splitting algorithm produce codes by transforming a given finite-state transition diagram that represents the code constraints. The state-splitting technique does so by a process which essentially differentiates states according to future. The BD method, on the other hand, differentiates states according to history. Both methods were recently unified and extended by Hollmann [55]. The resulting approach typically requires less design effort and leads to smaller decoding windows than either of its predecessors.

As to the power spectrum of codes, we mention the general analysis techniques of [19, 71, 45] as well as the elegant approach of [46], which is tailored to run-length-limited codes but can be extended in other directions [62].

Early examples of the exploitation of code redundancy towards a performance advantage are [121] for channels without ISI, and [119] for partial-response channels. In the latter category, Bi-Phase is an example *avant la lettre* [6, 103]. More recent contributions in this field include [61], [52] and [42].

Within the scope of a single chapter we cannot possibly do justice to the vast body of literature on modulation codes that has accumulated over time. Coding has been and remains an area of intense activity, and we refer the reader to two books [22], [100] and a number of review articles [23], [65], [28], [18], [29], [105], [99], [61], [78], [106] as points of entry for further reading. Also, the July 1983 issue of the *International Journal of Electronics* is entirely devoted to modulation codes and contains surveys of basics as well as of various application areas. We should finally mention Part II of the May 1991 issue of *IEEE Transactions on Information Theory*, which focuses on coding for storage devices.

APPENDIX 4A

Analysis of baseline wander

Denote the nominal continuous-time system response by $g(t)$ and its actual version by $q(t)$. Denote the Fourier integrals of $g(t)$ and $q(t)$ by $G(\Omega)$ and $Q(\Omega)$, respectively. We are concerned with the situation that $q(t)$ deviates from $g(t)$ as a result of AC-coupling due to a first-order high-pass filter with normalized cut-off frequency Ω_c. Correspondingly

$$Q(\Omega) = \frac{j\frac{\Omega}{\Omega_c}}{1+j\frac{\Omega}{\Omega_c}}G(\Omega) = [1 - \frac{1}{1+j\frac{\Omega}{\Omega_c}}]G(\Omega). \tag{4A.1}$$

Now $1/[1 + j\Omega/\Omega_c]$ is the transfer function of a low-pass filter with cut-off frequency Ω_c. In the usual case that $\Omega_c \ll 1$ and that $G(\Omega)$ is a smooth function of frequency, $G(\Omega) \simeq G(0)$ throughout the pass-band of this LPF. This leads to the approximation

$$Q(\Omega) \simeq G(\Omega) - \frac{G(0)}{1+j\frac{\Omega}{\Omega_c}}. \tag{4A.2}$$

In time-domain notation this expression reads

$$q(t) \simeq g(t) - \alpha\Omega_c \exp[-2\pi\Omega_c t/T]u(t) \tag{4A.3}$$

where $\alpha \triangleq 2\pi G(0)/T$ and $u(t)$ denotes the unit step function. The discrete-time system responses q_k and g_k are obtained from their continuous-time counterparts by

sampling at the instants $t = kT$. Thus $q_k = g_k - \alpha\Omega_c \exp[-2\pi\Omega_c k]u_k$ where u_k denotes the discrete-time unit step function. The second term on the right hand side characterizes baseline wander and can be modeled by a scaling factor $\alpha\Omega_c$ followed by a first-order recursive filter as in Fig. 4.9.

Exercise 4.10 Assume that $g(t)$ is a Nyquist-1 function with less than 100% excess bandwidth. Show that $G(0) = T$ so that $\alpha = 2\pi$. Hint: Translate (4.1) to the frequency domain. The resulting condition is Nyquist's first criterion for distortionless transmission (see Chapter 5).

APPENDIX 4B

Scramblers and pseudorandom sequences

In this appendix we discuss several aspects of scramblers and pseudorandom sequences. A mathematical development of these issues would require an elementary introduction to the theory of finite fields and this would carry us too far afield. The interested reader is referred to [49], [90] and [80] for such a development. Here we restrict ourselves to a summary of the main results. In Section 4B.1 we describe how self-synchronizing scramblers may be constructed on the basis of *maximum-length shift registers*. These registers produce pseudorandom sequences and the properties of these sequences are summarized in Section 4B.2.

4B.1 Self-synchronized scramblers

The scrambler and descrambler in Fig. 4.19 are fully determined by the mapping G. For reasons of causality, G must be some function of *past* scrambled digits a_{k-i}, $i \geq 1$. The descrambler normally operates on decisions $\hat{a}_k$ that may occasionally be incorrect. Every erroneous descrambler input may induce several erroneous outputs. To minimize this *error extension* it is desirable that c_k depends on only a few scrambled data symbols a_{k-i}, $i \geq 1$. This will also restrict complexity.

Example 4.41 The CCITT V.22.bis modem uses the mapping $c_k = a_{k-14}a_{k-17}$. Here a single erroneous descrambler input at instant k will cause erroneous outputs at instants k, $k + 14$ and $k + 17$. Error extension increases accordingly if c_k depends on more past digits.

Exercise 4.11 The topology of Fig. 4.19 remains self-synchronizing if we generate c_k on the basis of d_k rather than a_k. Explain why this is a bad idea.

One commonly constructs c_k by multiplying a small number of past digits a_{k-i}. To see how this works out, we first consider in more detail the situation that the source data signal d_k is a DC-wave $d_k = +1$. This leads to the *linear feedback shift register* (LFSR) of Fig. 4B.1.

The adjective 'linear' stems from the fact that the multiplications translate into modulo-2 additions in the alphabet $\{0, 1\}$. Only some of the shift register taps are used to construct c_k and these taps are normally specified in terms of a *connection polyno-*

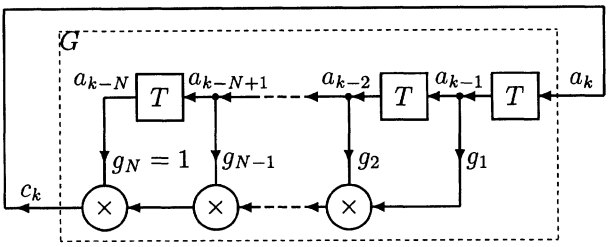


Figure 4B.1: Linear feedback shift register.

mial $G(\mathcal{D})$. This polynomial has binary weights $g_i \in \{0, 1\}$ that specify whether or not the ith tap is used. The zeroth coefficient g_0 is one by definition.

Example 4.42 The polynomial $G(\mathcal{D}) = 1 + \mathcal{D}^{14} + \mathcal{D}^{17}$ corresponds to the mapping $c_k = a_{k-14}a_{k-17}$ as in Example 4.41.

The order N of the polynomial determines the length of the shift register. Its *weight* determines the number of taps and thereby the amount of error extension. The sequence a_k that is generated by the LFSR is periodic. For a given value of N the period p can be at most $2^N - 1$. A sequence a_k with this maximum period is called a *maximum-length sequence* or *pseudorandom sequence*. In Section 4B.2 we will discuss various properties of pseudorandom sequences and we will see which generator polynomials are needed to generate them.

We have now established that the scrambler produces a maximum-length sequence in response to the all-one sequence d_k. It remains to consider the effect of other data sequences. This is most easily done by reverting to the alphabet $\{0, 1\}$, so that the multiplications in Figs. 4.19 and 4B.1 become modulo-2 additions. These are linear operations when considered in the field of binary numbers, and in this field the scrambler and descrambler are both linear building blocks. A treatment of finite-field theory is beyond the scope of this text, but we can get a reasonable idea of the effect of scrambler and descrambler by regarding them as linear in the usual sense. The descrambler is then a transversal structure to which we can associate a transfer function $G(\mathcal{D})$, and the scrambler is recursive with transfer function $1/G(\mathcal{D})$. The cascade of both has transfer function 1, i.e. the descrambler output is just the scrambler input, as desired. The scrambler output a_k can be decomposed into a steady-state response that is due to the source data sequence d_k and a transient response that is due to the all-zero input $d_k = 0$ (or, equivalently, to $d_k = +1$ in the alphabet $\{-1, 1\}$). The transient response is just the maximum-length sequence and does not die out as it would in a normal filter. It is this response that gives us the desired scrambling action.

Self-synchronizing scramblers can, in principle, have problems with periodic input sequences, more so than frame-synchronized ones. More specifically, when the data d_k has period p_0 then there is exactly one initial scrambler state that causes the scrambler output a_k to have period p_0 [94]. This is obviously undesirable when p_0 is small. To minimize the likelihood that this situation will ever occur one should take a large order

N for the scrambler/descrambler combination. For all other initial states the period of the output will be at least $p = 2^N - 1$ provided that p is selected to be a prime number. It is common engineering practice to adhere to this constraint.

4B.2 Pseudorandom sequences

The shift register of the LFSR of Fig. 4B.1 contains N bits $a_{k-1}, \ldots, a_{k-N}$ that together form the *state* of the LFSR at time k. There are 2^N distinct states and there is a single state from which the LFSR will never escape, viz. the all-one state ($a_{k-1} = a_{k-2} = \ldots = a_{k-N} = 1$).[9] To prevent the LFSR from producing the DC-wave $a_k = 1$ it is necessary that this be precluded. The state of the LFSR will then change every symbol interval. The sequence of states is periodic and the period p cannot be larger than the number of permissible states $2^N - 1$.

The LFSR is said to be a *maximum-length shift register* (MLSR) if it produces a state sequence of period $p = 2^N - 1$. The corresponding output sequence a_k is called a *maximum-length sequence* and also has period $p = 2^N - 1$. To obtain an MLSR it is necessary and sufficient that the connection polynomial $G(\mathcal{D})$ be *primitive* (see [90] for backgrounds of this term). Primitive polynomials exist for any order N and there are, in fact, often many possible choices. For a given order N, it is common to pick a polynomial with the smallest possible weight. This narrows the possibilities down considerably, though generally not to a single polynomial (see Problem 4.6). Table 4.1 lists a sample of these possibilities.

The N most recent bits $a_{k-1}, \ldots, a_{k-N}$ of the MLSR output together describe the current shift register state. The MLSR traverses all $2^N - 1$ permissible states and therefore if we look at N-bit segments of a maximum-length sequence we will see all possible n-bits words except for the all-one word. This causes the sequence to look rather random, even though it really is deterministic and periodic. For this reason a maximum-length sequence is often said to be *pseudorandom*. Indeed pseudorandom sequences display many of the properties of random sequences and they are often used for random number generation. For example, the time-average mean and autocorrelation function of a pseudorandom sequence are very similar to the mean and average of an uncorrelated binary data sequence (see Problems 4.7 and 4.8). Because of their periodicity, pseudorandom sequences have a line spectrum rather than a continuous PSD. This spectrum is flat except for a reduced DC component (see Problem 4.9), and thereby closely resembles the PSD of an uncorrelated random sequence.

We conclude this appendix by noting that scramblers and pseudorandom sequences are used in several other applications than the one considered here. For example, in Section 4.10 and Chapter 6 we will see that a *precoder*, which is basically a low-order self-synchronizing scrambler, can be used to avoid error propagation in partial-response systems. As another example, transmission in Fig. 4.19 of the all-one sequence d_k turns the scrambler into a pseudorandom data generator and permits the error rate of the sys-

[9] We recall that we take digits a_k to be in $\{-1, 1\}$. The all-one state becomes the all-zero state when translated to the logical alphabet $\{0, 1\}$.

N	$G(\mathcal{D})$	N	$G(\mathcal{D})$
1	$1+\mathcal{D}$	18	$1+\mathcal{D}^7+\mathcal{D}^{18}$
2	$1+\mathcal{D}+\mathcal{D}^2$	19	$1+\mathcal{D}+\mathcal{D}^2+\mathcal{D}^5+\mathcal{D}^{19}$
3	$1+\mathcal{D}+\mathcal{D}^3$	20	$1+\mathcal{D}^3+\mathcal{D}^{20}$
4	$1+\mathcal{D}+\mathcal{D}^4$	21	$1+\mathcal{D}^2+\mathcal{D}^{21}$
5	$1+\mathcal{D}^2+\mathcal{D}^5$	22	$1+\mathcal{D}+\mathcal{D}^{22}$
6	$1+\mathcal{D}+\mathcal{D}^6$	23	$1+\mathcal{D}^5+\mathcal{D}^{23}$
7	$1+\mathcal{D}^3+\mathcal{D}^7$	24	$1+\mathcal{D}+\mathcal{D}^2+\mathcal{D}^7+\mathcal{D}^{24}$
8	$1+\mathcal{D}^2+\mathcal{D}^3+\mathcal{D}^4+\mathcal{D}^8$	25	$1+\mathcal{D}+\mathcal{D}^{25}$
9	$1+\mathcal{D}^4+\mathcal{D}^9$	26	$1+\mathcal{D}+\mathcal{D}^2+\mathcal{D}^{26}$
10	$1+\mathcal{D}^3+\mathcal{D}^{10}$	27	$1+\mathcal{D}+\mathcal{D}^2+\mathcal{D}^5+\mathcal{D}^{27}$
11	$1+\mathcal{D}^2+\mathcal{D}^{11}$	28	$1+\mathcal{D}^2+\mathcal{D}^{28}$
12	$1+\mathcal{D}+\mathcal{D}^4+\mathcal{D}^6+\mathcal{D}^{12}$	29	$1+\mathcal{D}^3+\mathcal{D}^{29}$
13	$1+\mathcal{D}+\mathcal{D}^3+\mathcal{D}^4+\mathcal{D}^{13}$	30	$1+\mathcal{D}+\mathcal{D}^2+\mathcal{D}^{30}$
14	$1+\mathcal{D}+\mathcal{D}^6+\mathcal{D}^{10}+\mathcal{D}^{14}$	31	$1+\mathcal{D}^3+\mathcal{D}^{31}$
15	$1+\mathcal{D}+\mathcal{D}^{15}$	32	$1+\mathcal{D}+\mathcal{D}^2+\mathcal{D}^{23}$
16	$1+\mathcal{D}+\mathcal{D}^3+\mathcal{D}^{12}+\mathcal{D}^{16}$	33	$1+\mathcal{D}^{10}+\mathcal{D}^{33}$
17	$1+\mathcal{D}^3+\mathcal{D}^{17}$	34	$1+\mathcal{D}+\mathcal{D}^2+\mathcal{D}^{27}+\mathcal{D}^{34}$

Table 4.1: Minimum-weight primitive polynomials $G(\mathcal{D})$ of orders 1 to 34. After [90].

tem to be estimated in a very simple manner by observing the output of the descrambler. For an accurate measurement it is necessary to account for error extension in the descrambler [2]. Further applications are found in such diverse fields as estimation of nonlinearities [123] and spread spectrum communications [107, 125].

APPENDIX 4C

Effect of coding on signal-to-noise ratios

In this appendix we study the effect of coding on the per-symbol signal-to-noise ratio measures SNR (Sections 4C.1 and 4C.2) and FSN($e^{j2\pi\Omega}$) (Section 4C.3). Sections 4C.1 and 4C.2 use frequencies f in Hz rather than normalized frequencies Ω so as to be able to clearly distinguish between the uncoded and coded situations. In this notation, the channel has transfer function $F(f) = \int_{-\infty}^{\infty} f(t)e^{-j2\pi ft}dt$ and the frequency f should not be confused with the symbol $f(t)$ that we use for the impulse response of the channel. Noise $n(t)$ is not necessarily white and its PSD is denoted $\mathcal{N}(f)$. The development follows the lines of [10]. The reader is referred to that paper for a more detailed discussion and an extension of the results to discrete-time partial-response channels.

4C.1 Minimum rate of decrement

Let us consider a transfer function $G(f)$ whose magnitude $|G(f)|$ is monotonically non-increasing in $|f|$. This is a low-pass function and obviously

$$|G(Rf)|^2 \geq |G(f)|^2 \text{ for all } f \tag{4C.1}$$

for any $R \in (0,1]$. Let us assume that the channel satisfies $|F(f)|^2/\mathcal{N}(f) = f^{2m}|G(f)|^2$ for some non-negative integer m and some function G that satisfies (4C.1). For $m = 0$, $|F(f)|^2/\mathcal{N}(f)$ is a low-pass function, while a band-pass function with a $(2m)^{th}$-order zero at DC emerges for $m > 0$. The Lorentzian channel (see Section 2.3.5 and Appendix 3A) has $m = 1$. For the class of channels thus defined we claim that

$$\text{SNR}_R \leq R^{1-2m}\text{SNR}_1 \tag{4C.2}$$

where the subscript denotes the code rate at hand. To prove this claim we start from the SNR definition of (3.5) and (3.4). In terms of non-normalized frequencies f we may write

$$\text{SNR}_R = \int_{-\infty}^{\infty} |C_R(f)|^2 f^{2m} |G(f)|^2 df. \tag{4C.3}$$

Now coding compresses $c(t)$ according to $c_R(t) = c_1(t/R)$ and therefore $C_R(f) = R\,C_1(Rf)$ for all f. Correspondingly

$$\begin{aligned}
\text{SNR}_R &= R^2 \int_{-\infty}^{\infty} |C_1(Rf)|^2 f^{2m} |G(f)|^2 df \\
&\leq R^{2-2m} \int_{-\infty}^{\infty} |C_1(Rf)|^2 (Rf)^{2m} |G(Rf)|^2 df \\
&= R^{1-2m} \int_{-\infty}^{\infty} |C_1(\mu)|^2 \mu^{2m} |G(\mu)|^2 d\mu
\end{aligned} \tag{4C.4}$$

where the inequality is due to (4C.1). It remains to recognize the last integral as SNR_1.

For the low-pass case $m = 0$, SNR degrades at least as fast as for wideband channels. For the band-pass case $m \geq 1$, (4C.2) allows SNR to increase as R decreases. Even so, this increment may not materialize in practice, witness the Lorentzian channel of Example 4.12. Here $m = 1$, but $\text{SNR}_R \leq \text{SNR}_1$ with equality holding only at vanishing information densities D.

4C.2 Maximum rate of decrement

We now restrict $c(t)$ to be a rectangular pulse of width T ($c(t) = 1$ for $t \in [0,T]$ and 0 else) and require the code rate R to be rational according to $R = 1/N$ for $N = 1, 2, \ldots$. There are no restrictions on the impulse response $f(t)$ and the noise PSD $\mathcal{N}(f)$. For the category of systems thus defined we will analyze the dependence of the folded signal-to-noise ratio $\text{FSN}(e^{j2\pi f T_s})$ on R, and by taking the analysis one

step further we will arrive at a limit on the maximum rate of decrement of SNR with R. For future reference we define a function $Q_N(e^{j2\pi fT}) \triangleq \sin^2(\pi f T_s)/\sin^2(\pi f T_s/N)$ where T_s is the temporal spacing of source digits d_n and T_s/N is just the symbol interval T. Clearly Q_N is periodic in f with a period of $N/T_s = 1/T$ Hz. Within the fundamental interval $[-\frac{1}{2T}, \frac{1}{2T}]$, Q_N is a low-pass function with a maximum $Q_N(0) = N^2$. The following proposition relates the folded signal-to-noise ratio $\mathrm{FSN}_N(e^{j2\pi fT})$ for rate $R = 1/N$ to its counterpart $\mathrm{FSN}_1(e^{j2\pi fT_s})$ for rate $R = 1$.

Proposition 1: For all $N \in \{1, 2, 3, \ldots\}$ and all $f \in \mathbb{R}$ we have

$$\mathrm{FSN}_1(e^{j2\pi fT_s}) = \frac{1}{N}\sum_{n=0}^{N-1} Q_N(e^{j2\pi(f+\frac{n}{T_s})T})\mathrm{FSN}_N(e^{j2\pi(f+\frac{n}{T_s})T}). \tag{4C.5}$$

Proof: The rectangular pulse $c_1(t)$ of width T_s can be constructed by 'gluing' N pulses of width $T = T_s/N$ together according to

$$c_1(t) = \sum_{i=0}^{N-1} c_N(t - iT). \tag{4C.6}$$

In frequency-domain notation this expression becomes $C_1(f) = C_N(f)\sum_{i=0}^{N-1} e^{-j2\pi ifT}$. The right-hand side contains a geometric series that may be summed to yield

$$C_1(f) = C_N(f)\frac{1 - e^{-j2\pi fT_s}}{1 - e^{-j2\pi fT}}. \tag{4C.7}$$

Consequently $|C_1(f)|^2 = Q_N(e^{j2\pi fT})|C_N(f)|^2$ and with (3.4) this means that

$$\mathrm{SN}_1(f) = \frac{1}{N}Q_N(e^{j2\pi fT})\mathrm{SN}_N(f). \tag{4C.8}$$

In order to arrive at a relation between $\mathrm{FSN}_1(e^{j2\pi fT_s})$ and $\mathrm{FSN}_N(e^{j2\pi fT})$ we must perform a spectral folding operation in accordance with (3.6):

$$\begin{aligned}
\mathrm{FSN}_1(e^{j2\pi fT_s}) &= \sum_{i=-\infty}^{\infty} \mathrm{SN}_1(f + \frac{i}{T_s}) \\
&= \frac{1}{N}\sum_{n=0}^{N-1} Q_N(e^{j2\pi(f+\frac{n}{T_s})T}) \sum_{j=-\infty}^{\infty} \mathrm{SN}_N(f + \frac{n}{T_s} + \frac{j}{T}) \\
&= \frac{1}{N}\sum_{n=0}^{N-1} Q_N(e^{j2\pi(f+\frac{n}{T_s})T})\mathrm{FSN}_N(e^{j2\pi(f+\frac{n}{T_s})T})
\end{aligned} \tag{4C.9}$$

where the last equality is also due to (3.6). This concludes the proof.

Proposition 2: $\mathrm{SNR}_N \geq N^{-2}\mathrm{SNR}_1$.

Proof: We have

$$\mathrm{SNR}_1 = \frac{1}{T_s}\int_{-\frac{1}{2T_s}}^{\frac{1}{2T_s}} \mathrm{FSN}_1(e^{j2\pi fT_s})df \tag{4C.10}$$

and together with Proposition 1 we find after some elementary manipulations that

$$\mathrm{SNR}_1 = \frac{1}{NT}\int_{-\frac{1}{2T_s}}^{\frac{1}{2T_s}} Q_N(e^{j2\pi fT})\mathrm{FSN}_N(e^{j2\pi fT})df. \tag{4C.11}$$

Now $Q_N \leq N^2$ for all f and therefore

$$\mathrm{SNR}_1 \leq \frac{N}{T}\int_{-\frac{1}{2T_s}}^{\frac{1}{2T_s}} \mathrm{FSN}_N(e^{j2\pi fT})df = \frac{N^2}{T}\int_{-\frac{1}{2T}}^{\frac{1}{2T}} \mathrm{FSN}_N(e^{j2\pi fT})df = N^2\mathrm{SNR}_N. \tag{4C.12}$$

4C.3 Effect of coding on FSN$(e^{j2\pi\Omega})$

In this section we illustrate how coding affects the folded signal-to-noise ratio FSN$(e^{j2\pi\Omega})$. Two examples are given that pertain to idealized optical and magnetic recording systems. Both examples are taken from [9].

Example 4.43 Let us reconsider the idealized optical recording system of example 4.4 for the situation that signaling occurs with rectangular pulses $c(t)$ of width $T = RT_s$. The encoded data sequence a_k includes redundancy and conveys only $1/T_s$ bits of user information per second. For this reason we take T_s as a basis for normalization of the information density, i.e. we take $D = t_{50}/T_s$. This definition is compatible with the one of Example 4.4 because $R = 1$ in that example. Let us assume that $n(t)$ is white and that its level is such that SNR = 30 dB at $D = R = 1$. Fig. 4C.1 depicts corresponding folded signal-to-noise ratios.

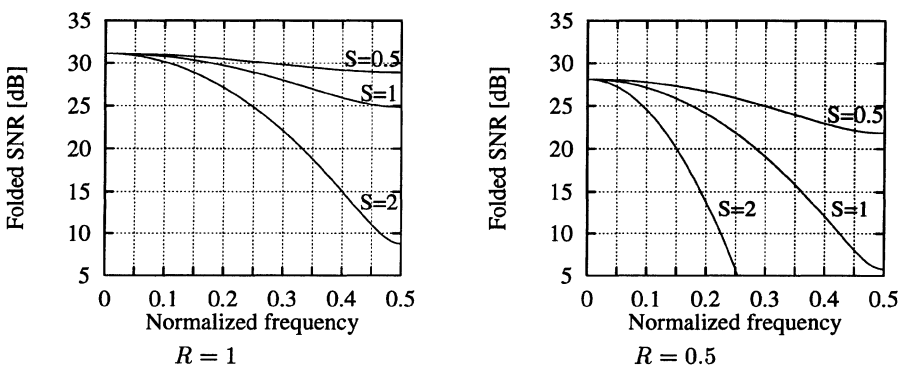


Figure 4C.1: Folded signal-to-noise ratio FSN$(e^{j2\pi\Omega})$ versus normalized frequency Ω for idealized optical recording channel at various normalized information densities D. Two different information rates R are considered.

At low densities ($D \ll R$) we essentially have a wideband channel. Here FSN($e^{j2\pi\Omega}$) is more or less flat and SNR (i.e. the average level of FSN($e^{j2\pi\Omega}$)) decreases linearly with R. High densities ($D \gg R$), on the other hand, correspond to a narrowband channel and here SNR decreases quadratically (see also Example 4.11). Since the low-frequency portion of FSN($e^{j2\pi\Omega}$) still decreases only linearly, the high-frequency portion must decrease much faster than quadratically and this is reflected in the deep depressions near the upper band edge. Reduction of R, therefore, causes excursions of FSN($e^{j2\pi\Omega}$) about SNR to grow progressively.

Exercise 4.12 Explain why FSN($e^{j2\pi\Omega}$) decreases only linearly in R for frequencies Ω near DC. Show that this remains true irrespective of the channel parameters whenever $c(t)$ is a hold function of width T. Hint: Consider the Fourier transform of $c(t)$ at multiples of the symbol frequency $1/T$.

Example 4.44 We next reconsider the Lorentzian channel of Appendix 3A and Example 4.12. The information density D is again normalized with respect to T_s. Noise $n(t)$ is white and its level is fixed such that SNR =20 dB at $D = R = 1$. Folded signal-to-noise ratios may be computed analytically (see Appendix 3A.1) and are depicted in Fig. 4C.2.

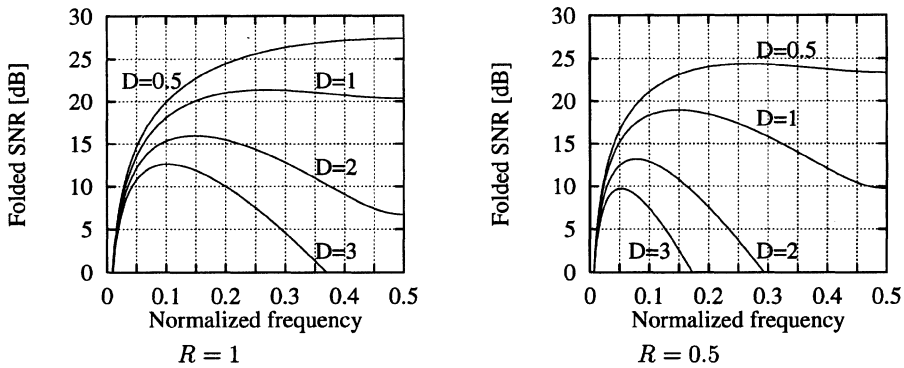


Figure 4C.2: Folded signal-to-noise ratio FSN($e^{j2\pi\Omega}$) versus normalized frequency Ω for Lorentzian channel at various normalized information densities D. Two different information rates R are considered.

For any density D, FSN($e^{j2\pi\Omega}$) vanishes at DC. At low densities ($D \ll R$) the function becomes independent of R. This reconfirms that there is no SNR penalty for reducing R. For $D \gg R$, the case of greatest practical interest, FSN($e^{j2\pi\Omega}$) deteriorates rapidly at high frequencies and this causes SNR to decrease in proportion to R^2 (see Example 4.12).

PROBLEMS

4.1 In this problem we are concerned with the effect of the modulation code on baseline wander. Consider the equivalent discrete-time model for baseline wander of Fig. 4.9 with $\alpha = 2\pi$.

1. Assume that data is binary and uncoded such that $\mathcal{A}(e^{j2\pi\Omega}) = 1$ for all Ω. How large is Ω_c allowed to be for ISI e_k due to baseline wander to be smaller than 1% RMS?
2. MFM is a run-length-limited modulation code with suppressed low-frequency content. Specifically, $\mathcal{A}(e^{j2\pi\Omega}) \simeq 0.2$ for frequencies Ω in the vicinity of 0 (compare (4.10)). How much larger can Ω_c now be chosen before e_k exceeds 1% RMS?

4.2 Consider a system with NRZ modulation and an ideal channel (i.e. a channel with transfer function $F(\Omega) = 1$ for all Ω and no noise). Assume that baseline wander occurs as a result of a AC-coupling that may be modeled as a first-order high-pass filter with normalized cut-off frequency Ω_c.

1. Determine the system response $q(t)$, i.e. the response of a single NRZ pulse to the high-pass filter.
2. Sketch $q(t)$ for $\Omega_c = 0.1$ and $\Omega_c = 0.01$. What is the effective duration of $q(t)$ in both cases?
3. Sketch the eye pattern for $\Omega_c = 0.1$. Pay particular attention to worst-case data patterns, i.e. to patterns that yield much intersymbol interference.
4. Determine the worst-case magnitude of the baseline wander as a function of Ω_c, along with the worst-case eye opening (relative to a value of 1 for an NRZ signal without baseline wander). What is the data pattern that causes this worst-case opening? How likely is this pattern if data a_k is uncorrelated?
5. Use the results of Problem 4.1 to determine the RMS baseline wander as a function of Ω_c. Compare this to the worst-case value. Which of the two values is most relevant?

4.3 Digital recording systems have long used run-length-limited codes with rates $R = 1/2$ and $R = 2/3$. Quantify the corresponding SNR losses with respect to uncoded storage for the situation that the system operates at densities that are high in the sense of Section 4.4.

4.4 Retrace the steps of Example 4.13 to quantify the effect on SNR of going from 2-AM to 8-AM.

4.5 Consider a self-synchronizing scrambler with polynomial $G(\mathcal{D}) = 1 + \mathcal{D} + \mathcal{D}^4$. Assume that the shift register of the scrambler has known contents $a_{k-1}, \ldots, a_{k-4}$ (see Figs 4.19 and 4B.1). How must the source data sequence d_k be chosen for all scrambled digits $a_k, a_{k+1}, \ldots$ to be $+1$? Comment on the likelihood of this particular sequence d_k.

4.6 Show that the time-reversed version of a pseudorandom sequence is also pseudorandom. How do the generator polynomials of both sequences interrelate?

4.7 This problem is concerned with the time-average mean $\overline{a} \triangleq \frac{1}{p}\sum_{k=0}^{p-1} a_k$ of a pseudorandom data sequence a_k of period $p = 2^N - 1$.

1. How many $+1$s and -1s are there in a full period p of the sequence?
   Hint: use the fact that every possible N-bit word occurs exactly once except for the all-one word.
2. Show that $\overline{a} = -1/p$.

This shows that $\overline{a}$ rapidly approaches zero as N increases.

4.8 This problem is concerned with the time-average autocorrelation function

$$R_a(k) \triangleq \frac{1}{p}\sum_{l=0}^{p-1} a_l a_{l+k} \tag{4P.1}$$

of a pseudorandom data sequence a_k of period $p = 2^N - 1$. In deriving this autocorrelation we will make use of the *cycle-and-add* property of pseudorandom sequences [49]. This property says that multiplication of a_k with a time-shifted version of itself yields another time-shifted version, i.e. $a_l a_{l+k} = a_{l+j}$ for all l and all $k \in \{1, \ldots, p-1\}$ where j is fully determined by k.

1. Show that $R_a(0) = 1$.
2. Use the cycle-and-add property along with the result of Problem 4.7 to show that $R_a(k) = -1/p$ for $k = 1, \ldots, p-1$.
3. Prove that $R_a(k)$ is periodic in k with period p.

This shows that $R_a(k)$ rapidly approaches the autocorrelation function of an uncorrelated random data sequence as N increases.

4.9 Use the results of Problem 4.8 to determine the time-average amplitude spectrum

$$S_a(m) \triangleq \frac{1}{p}\sum_{k=0}^{p-1} R_a(k) e^{-j2\pi km/p} \tag{4P.2}$$

of a pseudorandom data sequence a_k of period $p = 2^N - 1$. Comment on the structure of this function and compare it to the power spectral density of an uncorrelated random data sequence.

4.10 Consider a system in which sinc-pulses $c(t) = \sin(\pi t/T)/(\pi t/T)$ are used for signaling and in which the channel has white noise and an ideal low-pass transfer function with cut-off frequency f_c. Sketch the folded signal-to-noise $\mathrm{FSN}(e^{j2\pi\Omega})$ for the signaling rates $1/T = f_c$, $1/T = 2f_c$ and $1/T = 4f_c$. Which two regions of signaling rates can be distinguished? How does SNR change with $1/T$ in each of these regions?

4.11 Consider the GCR code of Section 4.8.3 for the situation that the source data sequence d_k is uncorrelated. Determine the average number of transitions per symbol interval as well as the average run-length $\overline{T}$ of encoded bits a_k. Compare these figures with those of a maxentropic $(0, 2)$ sequence.

4.12 What rate R should a code with the trellis diagram of Fig. 4.32 have in order to be maxentropic? Does the AMI code of Section 4.10 qualify?

4.13 Follow the lines of Example 4.26 to specify two constraints that will together produce a zero at a quarter of the symbol frequency ($\Omega = 0.25$). How practical is such a zero?
Hint: Observe that the real and imaginary parts of $A(0.25)$ are fully determined by the even

and odd digits a_{2m} and a_{2m+1}, respectively, and observe that these parts must be constrained simultaneously in order to obtain a spectral zero.

4.14 Consider a binary DC-free code with a given digital sum variation DSV. Show that runs of identical coded symbols can be at most DSV symbol intervals long. Regard the code as a run-length-limited (d, k) code as in Section 4.8. Determine the smallest possible value of d and the largest possible value of k.

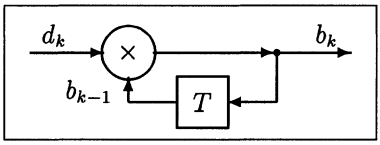


Figure 4P.1: First-order precoder.

4.15 In this problem we analyze some statistical properties of the first-order precoder of Fig. 4P.1. Precoder input d_k and output b_k are both binary and in the alphabet $\{-1, 1\}$. We assume that d_k is stationary and uncorrelated with mean $\overline{d}$ such that $|\overline{d}| < 1$. If the symbols $d_k = +1$ and $d_k = -1$ have probabilities p and $q = 1-p$, respectively, then $\overline{d} = p - q = 2p - 1$ or, equivalently, $p = (1 + \overline{d})/2$. The stationarity of d_k implies that b_k must also be stationary.

1. Show that b_k has zero mean. Hint: express $E[b_k]$ recursively in terms of $E[b_{k-1}]$ and $E[d_k]$, and make use of the fact that b_k is stationary.
2. Show that b_k has autocorrelation function $R_k^b = \overline{d}^{|k|}$ for all k. When is b_k uncorrelated?
   Hint: validate the expression by induction with respect to k. First consider $k = 0$ and next express R_{k+1}^b in terms of R_k^b.
3. Verify that b_k has power spectrum

$$P_b(e^{j2\pi\Omega}) \triangleq \sum_{k=-\infty}^{\infty} R_k e^{-j2\pi\Omega k} = \frac{1 - \overline{d}^2}{1 + \overline{d}^2 - 2\overline{d}\cos 2\pi\Omega}. \tag{4P.3}$$

4.16 Consider the Twinned-Binary and AMI modulation codes of Section 4.10. Assume that the data sequence d_n is uncorrelated.

1. Determine the transmitted power P_s. Compute and sketch the power spectral density $P_s(\Omega)$. Comment on this spectrum.
2. Determine the DSV. Are Twinned-Binary and AMI maxentropic?
3. Determine the maximum run-length k. What is the implication for timing recovery?

4.17 Redo Problem 4.16 for the Modified Duobinary code of Section 4.10.

4.18 Determine the DSV and the maximum run-length of the Quadra-Phase (QP) code described in Section 4.10. Is QP maxentropic? Specify a zero-disparity code with the same word-length $n = 4$, the same rate $R = 0.5$ (i.e. the same number of code words) and a smaller DSV. Does this code ring a bell of familiarity?

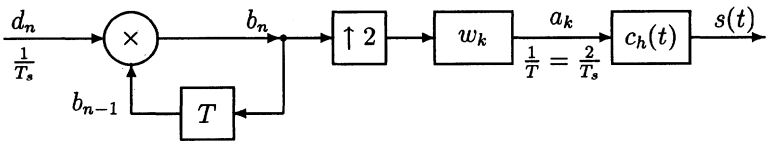


Figure 4P.2: Hedeman-3 encoder.

4.19 Fig. 4P.2 depicts an encoder for the Hedeman-3 (H3) code [122]. A binary data sequence $d_n \in \{-1, 1\}$ is applied to a first-order precoder, upsampled by a factor $L = 2$ and filtered by a filter with impulse response $w_k \triangleq \delta_k - \delta_{k-3}$ where δ_k denotes the Kronecker delta function. The resulting encoded sequence a_k of rate $1/T = 2/T_s$ is transformed into a continuous-time signal $s(t)$ via a hold function $c_c(t)$ of width T. For the sake of completeness we note that omission of the precoder in Fig. 4P.2 results in a write equalization scheme that originates in [54] and was reinvented in [86].

1. Is $s(t)$ binary or ternary? Is H3 linear?
2. Find the compound symbol response $c_c(t)$ of the encoder as defined in Section 4.10.
3. Determine the power spectral density $P_s(\Omega)$ of $s(t)$ for the situation that d_n is uncorrelated.
4. Comment on the merits and demerits of H3.

4.20 Determine the power spectral density $\mathcal{A}(e^{j2\pi\Omega})$ of the Pair-Selected Ternary code of Example 4.34 for the situation that data d_n is uncorrelated. Compare this density with the one of AMI and comment on the differences.

4.21 Show that full-set zero-disparity codes have DSV$= n$ and that DSV$\leq n$ for pruned zero-disparity codes. Give an example of a pruned code with DSV$< n$.

4.22 In this problem we are concerned with the digital sum variation (DSV) of full-set low-disparity codes (see Section 4.11.3). Code words have a fixed length n and are grouped into two tables. Table A collects code words whose disparity d is between 0 and K (both inclusive), while table B contains words with $d \in \{-K, \ldots, 0\}$. The encoder keeps track of the running digital sum at the end of each codeword, which we denote by RDS$_e$. It opts for table B in the next code word when RDS$_e \geq 0$ and for table A when RDS$_e < 0$.

1. Assume that the initial value of RDS$_e$ is zero. Which distinct values can RDS$_e$ assume? Distinguish between binary and ternary codes and assume for binary codes that K is even.
2. For any code word $\underline{c}$ with digits $c_0, \ldots, c_{n-1}$ we define the 'local' running digital sum as $b_i \triangleq \sum_{j=0}^{i} c_i$. The disparity d is the terminal value of this sum, i.e. $d = b_{n-1}$. Determine the minimum $b_{\min} \triangleq \min_{i=0,\ldots,n-1} b_i$ and maximum $b_{\max} \triangleq \max_{i=0,\ldots,n-1} b_i$ among all code words $\underline{c}$ in table A. Distinguish between binary and ternary codes, and assume for binary codes that K is even.
3. Determine the minimum and maximum value that the running digital sum can assume for binary codes with even K. What is the digital sum variation?
4. Repeat the previous question for ternary codes.

4.23 Determine the efficiency of the 4B-3T codes of Example 4.37 relative to a maxentropic DC-free code with the same digital sum variation (DSV) as 4B-3T. Answer the same question for the MS43 code of Example 4.38.

REFERENCES

[1] M.R. Aaron, 'PCM Transmission in the Exchange Plant', *Bell Syst. Tech. J.*, Vol. 41, pp. 99-141, Jan. 1962.

[2] I.L. Adams, 'An Improved Pseudo-Random Digital Sequence Error-Detector', *POEEJ* (Electrical Engineering Journal of the British Post Office), Vol. 73, pp. 186-187, Oct. 1980.

[3] R.L. Adler, D. Coppersmith and M. Hassner, 'Algorithms for Sliding Block Codes. An Application of Symbolic Dynamics to Information Theory', *IEEE Trans. Inform. Theory*, Vol. IT-29, No. 1, pp. 5-22, Jan. 1983.

[4] R. Adler, M. Hassner, and J. Moussouris, 'Method and apparatus for Generating a Noiseless Sliding Block Code for a $(1, 7)$ Channel with Rate $2/3$, U.S. Pat. 4 413 251, 1982.

[5] R.H. Barker, B.P. Patent No. 706,687, issued April 7, 1954.

[6] J.W.M. Bergmans, 'Discrete-Time Models for Digital Magnetic Recording', *Philips J. Res.*, Vol. 41, pp. 531-558, 1986.

[7] J.W.M. Bergmans and S. Mita, 'Two New Equalisation Schemes for High-Density Digital Magnetic Recording Systems with Quad-Phase Modulation Code', *Electron. Lett.*, Vol. 26, No. 1, pp. 13-14, Jan. 4, 1990.

[8] J.W.M. Bergmans, S. Mita, M. Izumita and N. Doi, 'Partial-Response Decoding of Rate 1/2 Modulation Codes for Digital Storage', *IEEE Trans. Commun.*, Vol. COM-39, pp. 1569-1581, Nov. 1991.

[9] J.W.M. Bergmans, 'SNR Merits of Binary Modulation Codes in Equalized Digital Recording Systems', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 10, pp. 157-167, Jan. 1992.

[10] J.W.M. Bergmans, 'Effect of Binary Modulation Codes with Rate $R = 1/N$ on Equivalent Discrete-Time Models for Channels with Intersymbol Interference', *IEEE Trans. Inform. Theory*, Vol. 38, No. 5, pp. 1587-1592, Sept. 1992.

[11] J.W.M. Bergmans, H.D.M. Hollmann and J.O. Voorman, 'Anti-Whistle Codes', submitted to *IEEE Trans. Commun.*, 1996.

[12] J.A. Bixby and R.A. Ketcham, 'Q.P., An Improved Code for High-Density Digital Recording', *IEEE Trans. Magn.*, Vol. MAG-15, pp. 1465-1467, Nov. 1979.

[13] R. Blahut, *Theory and Practice of Error Control Codes*. Menlo Park, CA: Addison-Wesley, 1983.

[14] R. Blahut, *Digital Transmission of Information*. Reading, MA: Addison-Wesley, 1990.

[15] F.K. Bowers, 'Pulse Code Transmission System', U.S. Patent 2,967,947, Oct. 25, 1960.

[16] R.M. Brooks and A. Jessop, 'Line Coding for Optical Fibre Systems', *Int. J. Electron.*, Vol. 55, No. 1, pp. 81-120, July 1983.

[17] R. Brush, 'Design Considerations for the D-2 PAL Composite DVTR', in *Proc. 4th Int. Conf. Video, Audio and Data Recording*, (York, England), Mar. 1988, pp. 141-148.

[18] J.B. Buchner, 'Ternary Line Codes', *Philips Telecommun. Review*, Vol. 34, No. 2, pp. 72-85, June 1976.

[19] G.L. Cariolaro, G.L. Pierobon and G.P. Tronca, 'Analysis of Codes and Spectra Calculations', *Int. J. Electron.*, Vol. 55, No. 1, pp. 35-79, 1983.

[20] R.O. Carter, 'Low-Disparity Binary Coding System', *Electron. Lett.*, Vol. 1, pp. 67-69, May 1965.

[21] K.W. Cattermole, 'Low-Disparity Codes and Coding for PCM', *IEE Conf. on Transmission Aspects of Commun. Networks*, London, Feb. 1964.

[22] K.W. Cattermole, *Principles of Pulse Code Modulation*. London: Iliffe Books, 1969.

[23] K.W. Cattermole, 'Principles of Digital Line Coding', *Int. J. Electron.*, Vol. 55, pp. 3-33, July 1983.

[24] T.M. Chien, 'Upper Bound on the Efficiency of DC-Constrained Codes', *Bell Syst. Tech. J*, Vol. 49, pp. 2267-2287, Nov. 1970.

[25] R.D. Cideciyan, F. Dolivo, R. Hermann, W. Hirt, and W. Schott, 'A PRML System for Digital Magnetic Recording', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 38-56, Jan. 1992.

[26] M. Cohn, G. Jacoby, and A. Bates, III, 'Data Encoding Method and System Employing Two-Thirds Code Rate with Full Word Look-Ahead', U.S. Pat. 4 337 458, 1982.

[27] A. Croisier, 'Introduction to Pseudoternary Transmission Codes', *IBM J. Res. Develop*, Vol. 14, pp. 354-367, July 1970.

[28] N.Q. Duc, 'Line Coding Techniques for Baseband Digital Transmission', *Austral. Telecommun. Res.*, Vol. 9, pp. 3-17, May 1975.

[29] N.Q. Duc and B.M. Smith, 'Line Coding for Digital Data Transmission', *Austral. Telecommun. Res.*, Vol. 11, No. 2, pp. 14-27, 1977.

[30] J.S. Eggenberger and P. Hodges, 'Sequential Encoding and Decoding of Variable Word Length, Fixed Rate Data Rate Codes', U.S. Patent 4,115,768, 1978.

[31] J. Eggenberger and A.M. Patel, 'Method and Apparatus for Implementing Optimum PRML Codes', U.S. Patent 4,707,681, issued Nov. 17, 1987.

[32] I.J. Fair, W.D. Grover, W.A. Krzymien, and R.I. MacDonald, 'Guided Scrambling: A New Line Coding Technique for High Bit Rate Fiber Optic Transmission Systems', *IEEE Trans. Commun.*, Vol. 39, No. 2, pp. 289-297, Feb. 1991.

[33] P.A. Franaszek, 'Sequence-State Coding for Digital Transmission', *Bell Syst. Tech. J.*, Vol. 46, pp. 143-157, Dec. 1967.

[34] P.A. Franaszek, 'Sequence State Methods of Run-Length-Limited Coding', *IBM J. Res. Develop.*, Vol. 14, No. 4, pp. 375-383, July 1970.

[35] P.A. Franaszek, 'Run-Length-Limited Variable Length Coding with Error Propagation Limitation', U.S. Patent 3,689,899, 1972.

[36] P.A. Franaszek, 'On Future-Dependent Block-Coding for Input-Restricted Channels', *IBM J. Res. Develop.*, Vol. 23, pp. 75-81, Jan. 1979.

[37] P.A. Franaszek, 'Synchronous Bounded Delay Coding for Input-Restricted Channels', *IBM J. Res. Develop.*, Vol. 24, pp. 43-48, Jan. 1980.

[38] P.A. Franaszek, 'A General Method for Channel Coding', *IBM J. Res. Develop.*, Vol. 24, pp. 638-641, Sept. 1980.

[39] P.A. Franaszek, 'Construction of Bounded Delay Codes for Discrete Noiseless Channels', *IBM J. Res. Develop.*, Vol. 26, pp. 506-514, July 1982.

[40] J.N. Franklin and J.R. Pierce, 'Spectra and Efficiency of Binary Codes Without DC', *IEEE Trans. Commun.*, Vol. COM-20, No. 6, pp. 1182-1184, Dec. 1972.

[41] L.J. Fredrickson, 'On the Shannon Capacity of DC- and Nyquist-Free Codes', *IEEE Trans. Inform. Theory*, Vol. IT-37, No. 3, pp. 918-923, May 1991.

[42] L.J. Fredrickson, R. Karabed, J.W. Rae, P.H. Siegel, H. Thapar and R.W. Wood, 'Improved Trellis Coding for Partial-Response Channels', *IEEE Trans. Magn.*, Vol. 31, No. 2, pp. 1141-1148, March 1995.

[43] S. Fukuda, Y. Kojima, Y. Shimpuku and K. Odaka, '8/10 Modulation Codes for Digital Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-22, pp. 1194-1197, Sept. 1986.

[44] A. Gabor, 'Digital Signals at High Density on Magnetic Tape', *Electronics*, Vol. 32, pp. 72-75, Oct. 1959.

[45] P. Galko and S. Pasupathy, 'The Mean Power Spectral Density of Markov Chain Driven Signals', *IEEE Trans. Inform. Theory*, Vol. IT-28, pp. 473-481, May 1982.

[46] A. Gallopoulos, C. Heegard and P.H. Siegel, 'The Power Spectrum of Run-Length-Limited Codes', *IEEE Trans. Commun.*, Vol. COM-37, No. 9, pp. 906-917, Sept. 1989.

[47] P.J. van Gerwen, 'On the Generation and Application of Pseudo-Ternary Codes in Pulse Transmission', *Philips Res. Rept.*, Vol. 20, pp. 469-484, Aug. 1965.

[48] P.J. van Gerwen, W.A.M. Snijders and N.A.M. Verhoeckx, 'Dual Digital Transmission System for Multipair Cables Based on 'Crank-Shaft Coding", *Electron. Lett.*, Vol. 20, No. 15, pp. 619-621, Jul. 1984.

[49] S.W. Golomb, *Shift Register Sequences*. San-Fransisco: Holden-Day, 1967.

[50] L.J. Greenstein, 'Spectrum of a Binary Signal Block Coded for DC Suppression', *Bell Syst. Tech. J.*, Vol. BSTJ-53, No. 6, pp. 1103-1126, July-Aug. 1974.

[51] J.M. Griffiths, 'Binary Code Suitable for Line Transmission', *Electronics Letters*, Vol. 5, pp. 79-81, 1969.

[52] R. Haeb and R.T. Lynch, Jr., 'Trellis Codes for Partial-Response Magnetooptical Direct Overwrite Recording', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 182-190, Jan. 1992.

[53] M. Hecht and A. Guida, 'Delay Modulation', *Proc. IEEE*, Vol. 57, pp. 1314-1316, July 1969.

[54] H.D.L. Hollmann, 'The General Solution of Write-Equalization for RLL (d, k) Codes', *IEEE Trans. Inform. Theory*, Vol. IT-37, No. 3, May 1991.

[55] H.D.L. Hollmann, 'On the Construction of Bounded-Delay Encodable Codes for Constrained Systems', *IEEE Trans. Inform. Theory*, Vol. 41, No. 5, pp. 1354-1378, Sept. 1995.

[56] G. Jacoby, 'A New Look-Ahead Code for Increased Data Density', *IEEE Trans. Magn.*, Vol. MAG-13, No. 5, pp. 1202-1204, Sept. 1977.

[57] G. Jacoby and R. Kost, 'Binary Two-Thirds Rate Code with Full Word Look-Ahead', *IEEE Trans. Magn.*, Vol. MAG-20, No. 5, pp. 709-714, Sept. 1984.

[58] F. Jorgensen, *The Complete Handbook of Magnetic Recording*. Blue Ridge Summit, Pennsylvania: TAB Books, July 1980.

[59] P. Kabal and S. Pasupathy, 'Partial-Response Signaling', *IEEE Trans. Commun.*, Vol. COM-23, No. 9, pp. 921-934, Sept. 1975.

[60] J.A.H. Kahlman and K.A Schouhamer Immink, 'Channel Code with Embedded Pilot Tracking Tones for DVR', *IEEE Trans. Consumer Electr.*, Vol. CE-41, No. 1, pp. 180-185, Feb. 1995.

[61] R. Karabed and P.H. Siegel, 'Matched Spectral-Null Codes for Partial-Response Channels', *IEEE Trans. Inform. Theory*, Vol. IT-37, No. 3, pp. 818-855, May 1991.

[62] K.J. Kerpez, 'The Power Spectral Density of Maximum Entropy Charge Constrained Codes', *IEEE Trans. Inform. Theory*, Vol. 35, No. 3, pp. 692-695, May 1989.

[63] K.J. Kerpez, A. Gallopoulos and C. Heegard, *Maximum Entropy Charge-Constrained Run-Length Codes, IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 242-253, Jan. 1992.

[64] A. Knoll, 'Spectrum Analysis of Digital Magnetic Recording Waveforms' *IEEE Trans. on Electronic Computers*, Vol. EC-16, No. 6, pp. 732-743, Dec. 1967.

[65] H. Kobayashi, 'A Survey of Coding Schemes for Transmission and Recording of Digital Data', *IEEE Trans. Commun. Technol.*, Vol. COM-19, No. 6, pp. 1087-1100, Dec. 1971.

[66] E.R. Kretzmer, 'Generalization of a Technique for Binary Data Communication', *IEEE Trans. Commun. Technol.*, Vol. COM-14, pp. 67-68, Feb. 1966.

[67] J.L. Lawson and G.E. Uhlenbeck, *Threshold Signals*. Boston Technical Publishers Inc., 1964.

[68] E.A. Lee and D.G. Messerschmitt, *Digital Communication*. Boston: Kluwer Academic Publishers, 1988.

[69] A. Lempel and M. Cohn, 'Lookahead Coding for Input Restricted Channels', *IEEE Trans. Inform. Theory*, Vol. IT-28, pp. 933-937, Nov. 1982.

[70] A. Lender, 'The Duobinary Technique for High-Speed Data Transmission', *IEEE Trans. Communication and Electronics*, Vol. 82, pp. 214-218, May 1963.

[71] D.A. Lindholm, 'Power Spectra of Channel Codes for Digital Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-14, No. 5, pp. 321-323, Sept. 1978.

[72] R.T. Lynch, Jr., 'Channels and Codes for Magnetooptical Recording', *IEEE J. Selected Areas in Commun.*, Vol. 10, No. 1, pp. 57-72, Jan. 1992.

[73] R.F. Lyon, 'Two-Level Block Encoding for Digital Transmission', *IEEE Trans. Commun.*, Vol. COM-21, No. 12, pp. 1438-1441, Dec. 1973.

[74] R.W. Lucky, J. Salz and E.J. Weldon, Jr., *Principles of Data Communication*. New York: McGraw-Hill, 1968.

[75] B. Marcus, 'Factors and Extensions of Full Shifts', *Monatshefte fur Math.*, Vol. 88, pp. 239-247, 1979.

[76] B.H. Marcus, A.M. Patel and P.H. Siegel, 'Method and Apparatus for Implementing a PRML Code', U.S. Patent 4,786,890, issued Nov. 22, 1988.

[77] B.H. Marcus and R. Roth, 'Bounds on the Number of States in Encoder Graphs for Input-Constrained Channels', *IEEE Trans. Inform. Theory*, Vol. IT-37, No. 3, pp. 742-758, May 1991.

[78] B. Marcus, P.H. Siegel and J.K. Wolf, 'Finite-State Modulation Codes for Data Storage', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 5-37, Jan. 1992.

[79] J.L. Massey, 'Optimum Frame Synchronization', *IEEE Trans. Commun.*, Vol. COM-20, No. 2, pp. 115-119, Apr. 1972.

[80] R.J. McEliece, *Finite Fields for Computer Scientists and Engineers*. Norwell, MA: Kluwer Academic Publishers, 1987.

[81] L.A. Meacham, 'Twinned Binary Transmission', U.S. Patent No. 2,759,047, issued Aug. 14, 1956.

[82] A. Miller, U.S. pat. 3 108 261 (Oct. 22, 1963).

[83] J.W. Miller, US Patent 4,027,335 (May 31, 1977).

[84] Y. Nakagome, K. Amano and C. Ota, 'A Multilevel Code Having Limited Digital Sum', *Electron. Commun. in Japan*, Vol. 51-A, pp. 37-44, Jan. 1968.

[85] K. Norris and D.S. Bloomberg, 'Channel Capacity of Charge-Constrained Run-Length-Limited Codes', *IEEE Trans. Magn.*, Vol. MAG-17, pp. 3452-3455, 1981.

[86] Y. Okamoto, H. Osawa, K. Takigawa and K. Ono, 'Performance Improvement in Digital Magnetic Recording by Write and Read Equalization', *Electronics and Communications in Japan*, Part 2, Vol. 76, No. 3, pp. 52-62, 1993.

[87] A. Patel, 'Zero-Modulation Encoding in Magnetic Recording', *IBM J. Res. Develop.*, Vol. 19, No. 4, pp. 366-378, July 1975.

[88] J.B.H. Peek, 'Communications Aspects of the Compact Disc Digital Audio System', *IEEE Communications Magazine*, pp. 7-15, Feb. 1985.

[89] M.G. Pelchat and J.M. Geist, 'Surprising Properties of Two-Level 'Bandwidth Compaction' Codes', *IEEE Trans. Commun.*, Vol. COM-23, No. 9, pp. 878-883, Sept. 1975.

[90] W. Peterson and E. Weldon, *Error-Correcting Codes*, 2nd edition. Cambridge, MA: M.I.T. Press, 1972.

[91] J.R. Pierce, 'Some Practical Aspects of Digital Transmission', *IEEE Spectrum*, Vol. 5, pp. 63-70, 1968.

[92] G. Pierobon, 'Codes for Zero Spectral Density at Zero Frequency', *IEEE Trans. Inform. Theory*, Vol. IT-30, No. 2, pp. 425-429, Mar. 1984.

[93] J.W. Rae, G.S. Christiansen, P.H. Siegel, R. Karabed, H. Thapar and S. Shih, 'Design and Performance of a VLSI 120 Mb/s Trellis-Coded Partial Response Channel', *IEEE Trans. Magn.*, Vol. MAG-31, No. 2, pp. 1208-1214, March 1995.

[94] J.E. Savage, 'Some Simple Self-Synchronizing Digital Data Scramblers', *Bell Syst. Tech. J.*, Vol. 46, pp. 449-487, Feb. 1967.

[95] R.C. Schneider, 'Write Equalization in High-Density Recording', *IBM J. Res. Develop.*, Vol. 29, No. 6, pp. 563-568, Nov. 1985.

[96] K.A. Schouhamer Immink, 'Spectrum Shaping with Binary DC^2-constrained Codes', *Philips J. Res.*, Vol. 40, pp. 40-53, 1985.

[97] K.A. Schouhamer Immink, 'Signal-to-Noise Ratio of Pilot Tracking Tones Embedded in Binary Coded Signals', *IEEE Trans. Magn.*, Vol. 24, No. 2, pp. 2004-2009, March 1988.

[98] K.A. Schouhamer Immink, 'Coding Techniques for Partial-Response Channels', *IEEE Trans. Commun.*, Vol. COM-36, pp. 1163-1165, Oct. 1988.

[99] K.A. Schouhamer Immink, 'Runlength-Limited Sequences', *Proc. IEEE*, Vol. 78, No. 11, pp. 1745-1759, Nov. 1990.

[100] K.A. Schouhamer Immink, *Coding Techniques for Digital Recorders*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[101] K.A. Schouhamer Immink and H.D.L. Hollmann, 'Prefix-Synchronized Run-Length-Limited Sequences', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 214-222, Jan. 1992.

[102] K.A. Schouhamer Immink, 'Block-Decodable Runlength-Limited Codes via Look-Ahead Technique', *Philips J. Res.*, Vol. 46, pp. 293-3110, 1992.

[103] K.A. Schouhamer Immink, 'EFMPLUS: The Coding Format of the Multimedia Compact Disc', *IEEE Trans. Consumer Electr.*, Vol. 41, No. 3, pp. 491-497, Aug. 1995.

[104] C.E. Shannon, 'A Mathematical Theory of Communication', *Bell Syst. Tech. J.*, Vol. 27, pp. 379-423, 623-656, Oct. 1948.

[105] P.H. Siegel, 'Recording Codes for Digital Magnetic Storage', *IEEE Trans. Magn.*, Vol. MAG-21, No. 5, pp. 1344-1349, Sep. 1985.

[106] P.H. Siegel and J.K. Wolf, 'Modulation and Coding for Information Storage', *IEEE Communications Magazine*, Vol. 29, No. 12, pp. 68-86, Dec. 1991.

[107] M.K. Simon, J.K. Omura and R.A. Scholtz, *Spread Spectrum Communications*. Rockville, MD: Computer Science, 1985.

[108] J.M. Sipress, 'A New Class of Selected Ternary Pulse Transmission Plans for Digital Transmission Lines', *IEEE Trans. Commun. Technol.*, Vol. COM-13, pp. 366-372, Sept. 1965.

[109] J.J. Stiffler, *Theory of Synchronous Communications*. Englewood Cliffs, N.J: Prentice-Hall, 1971.

[110] E.D. Sunde, 'Theoretical Fundamentals of Pulse Transmission - I', *Bell Syst. Tech. J.*, Vol. 33, pp. 721-788, May 1984.

[111] Y. Takasaki, M. Tanaka, N. Maeda, K. Yamashita and K. Nagano, 'Optical Pulse Formats for Fibre Optic Digital Communications', *IEEE Trans. Commun.*, Vol. COM-24, pp. 404-413, Apr. 1976.

[112] D. Tang and L. Bahl, 'Block Codes for a Class of Constrained Noiseless Channels', *Inform. Contr.*, Vol. 17, pp. 436-461, 1970; and pp. 462-474, 1973.

[113] S. Tazaki, F. Takeda, H. Osawa and Y. Yamada, 'Performance Comparison of 8-10 Conversion Codes', Proc. 5th Int. Conf. on Video and Data Recording, Southampton, U.K., pp. 79-84, 1984.

[114] D.H. Veillard, 'Compact Spectrum Recording, A New Binary Process Maximizing the Use of a Recording Channel', *IEEE Trans. Magn.*, Vol. MAG-20, pp. 891-893, Sept. 1984.

[115] M.G. Vry and P.J. van Gerwen, 'The Design of a $1+1$ System for Digital Transmission to the Subscriber', in *Proc. ISSLS 80*, NTG-Fachberichte 73, pp. 36-40.

[116] D.B. Waters, 'Line Codes for Metallic Cable Systems', *Int. J. Electron.*, Vol. 55, No. 1, pp. 159-169, July 1983.

[117] G.F. Watson, 'Prolog to Runlength-Limited Sequences', Proc. IEEE, Vol. 78, No. 11, p. 1744, Nov. 1990.

[118] A. Weathers and J.K. Wolf, 'A New Rate 2/3 Sliding-Block Code for the (1,7) Run-Length Constraint with the Minimal Number of Encoder States', *IEEE Trans. Inform. Theory*, Vol. IT-37, No. 3, pt. II, pp. 908-913, May 1991.

[119] J.K. Wolf and G. Ungerboeck, 'Trellis Coding for Partial-Response Channels', *IEEE Trans. Commun.*, Vol. COM-34, No. 8, pp. 765-773, Aug. 1986.

[120] H.-W. Wong-Lam and H.D.L. Hollmann, 'A General Solution of Write Equalization for Digital Magnetic Tape Recording', *IEEE Trans. Magn.*, Vol. MAG-27, No. 5, pp. 4377-4381, Sept. 1991.

[121] R. Wood, 'Viterbi Reception of Miller-Squared Code on a Tape Channel', *Proc. 4th Int. Conf. Video and Data Recording, IERE Conf. Proc. 54*, Southampton, England, Apr. 1982, pp. 333-343.

[122] R.W. Wood, 'Further Comments on the Characteristics of the Hedeman H-1, H-2 and H-3 Codes', *IEEE Trans. Commun.*, Vol. COM-31, No. 1, pp. 105-110, Jan. 1983.

[123] D. Palmer, P. Ziperovich, R. Wood and T. Howell, 'Identification of Nonlinear Write Effects Using Pseudorandom Sequences', *IEEE Trans. Magn.*, Vol. MAG-23, No. 5, pp. 2377-2379, Sept. 1987.

[124] S. Yoshida and S. Yajima, 'On the Relation Between an Encoding Automation and the Power Spectrum of Its Output Sequence', *Trans. IECE Japan*, Vol. E59, No. 5, pp. 1-7, May 1976.

[125] R. Ziemer and W. Peterson, *Digital Communications and Spread Spectrum.* New York: Macmillan, 1985.

5

LINEAR EQUALIZATION

5.1 INTRODUCTION

Baseband data transmission is often hampered by linear intersymbol interference (ISI) and noise (see Chapters 2 and 3). In the present chapter we deal with the oldest technique to counter these two disturbances, namely linear equalization. The linear equalizer (LE) is just a linear filter that suppresses ISI at the periodic instants at which decisions are taken. The name of the technique originates from the fact that the filter must essentially equalize the transfer characteristics of the channel across a band that ranges from DC up to about the Nyquist frequency. A roll-off beyond this band serves to suppress out-of-band noise. The LE can be analog, digital or a mixture of the two. The digital portion of the LE is called T-spaced when the sampling rate coincides with the symbol rate $1/T$, and fractionally-spaced when it is oversampled with respect to symbol rate. If the channel characteristics are *a priori* known then the LE can be a fixed filter. More often than not, however, channel characteristics are uncertain and the LE is adaptive. In the present chapter we are mainly concerned with the desired transfer characteristics of the LE, and to a lesser extent with the problem of achieving or approximating these desired characteristics with a practical equalizer implementation. That problem is inherent in adaptive equalization as discussed in Chapter 8.

Ideally the transfer characteristics of the equalizer are determined in such a manner as to minimize the probability of decision errors. The ensuing design problem is, however, mathematically untractable and one usually resorts to simpler criteria instead. Of particular interest is the zero-forcing (ZF) criterion, which states that all ISI should be eliminated at the decision instants (Section 5.2). The ZF criterion imposes no constraint on the intermediate instants, and this freedom can be used in several ways. One of these is to minimize noise enhancement (Section 5.3). Unfortunately, even the minimum enhancement becomes infinite when the folded signal-to-noise ratio of the channel has spectral zeros. The ZF LE then becomes impractical. To overcome this problem one must allow some residual ISI in exchange for improved noise suppression. This tradeoff is central to the MMSE LE, which minimizes the compound power of ISI and noise at the decision instants (Section 5.4). As a result, noise enhancement can

no longer be infinite but can still be large. Better margins against noise require more powerful reception techniques, like those of Chapters 6 and 7.

ZF and MMSE equalization can be used with arbitrary modulation codes but they do not necessarily exploit all code properties to full advantage. In particular for run-length-limited codes, the minimum run-length constraint provides added freedom for equalizer dimensioning, and this freedom can be used to lower noise enhancement (Section 5.5). Minimization of noise enhancement serves as a key objective for equalizer design throughout this chapter, but other objectives can be brought to bear as well. By way of illustration we discuss in Section 5.6 how a non-adaptive LE can be designed to have an optimum resistance to channel parameter variations and to timing errors as induced by imperfect timing recovery. No matter what design approach is used, linear equalization is usually performed at the receiving end of the system, and our treatment in Sections 5.2 to 5.6 reflects this practice. A part of the equalization can, however, be performed at the transmitting end of the system by means of a transmit filter. The design of such filters is briefly addressed in Section 5.7. In cases where both a transmit filter and a receive-side LE are present the question arises how filtering should be distributed across transmitter and receiver in order to maximize the noise immunity of the system. This question is also addressed in Section 5.7. A final section provides considerations that pertain to implementation of equalizer and transmit filter.

The first half of the chapter builds upon material of Sections 3.2 and 3.10, and the reader is adviced to review these sections before passing on.

5.2 ZERO-FORCING LINEAR EQUALIZER

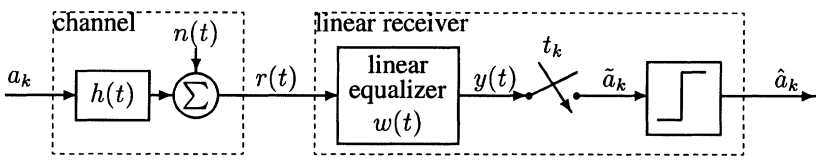


Figure 5.1: Zero-forcing linear equalizer operating on the output $r(t)$ of a noisy dispersive channel. The output $y(t)$ of the LE is sampled at the instants $t_k = kT$ to obtain estimates $\tilde{a}_k$ about a_k. These are quantized by a slicer into symbol decisions $\hat{a}_k$.

The channel model in Fig. 5.1 is the hybrid one of Section 3.2.4, i.e. the symbol response $h(t)$ accounts for the compound effect of the linear pulse modulator in the transmitter and the actual continuous-time channel. The hybrid channel operates on the encoded data sequence a_k of rate $1/T$ and has noise $n(t)$ that is additive and stationary. The receiver contains a linear equalizer with impulse response $w(t)$, whose output is sampled at instants $t_k = kT$. A symbol-by-symbol threshold detector operates on the sampler output $\tilde{a}_k$ to take decisions $\hat{a}_k$ with respect to a_k. The zero-forcing (ZF) criterion states that transmission be *distortionless*, i.e. that all ISI be forced to zero at the decision instants t_k. The detector input $\tilde{a}_k$ is, then, just a noisy version of a_k, i.e.

$\tilde{a}_k = a_k + u_k$ where $u_k = (n * w)(t_k)$ is the noise component of $\tilde{a}_k$. In order to achieve this objective the system response $q(t) \triangleq (h * w)(t)$ up until the equalizer must be a *Nyquist-1 pulse*. Such a pulse is characterized in that its sampled version is a Kronecker delta function, i.e. $q(kT) = \delta_k$ for all k. Since the spectral effect of sampling is a folding operation (see Chapter 1), an equivalent characterization is

$$\frac{1}{T} \sum_{n=-\infty}^{\infty} Q(\Omega + n) = 1 \text{ for all } \Omega, \tag{5.1}$$

where $Q(\Omega)$ is the Fourier integral of $q(t)$. We recall that Ω is a normalized measure of frequency, with $\Omega = 1$ corresponding to the symbol rate $1/T$. Equation (5.1) is Nyquist's celebrated first criterion for distortionless transmission [24]. Since $q(t)$ can have arbitrary values at all instants $t \neq t_k$, there are infinitely many Nyquist characteristics $Q(\Omega)$. Of particular interest are characteristics with restricted excess bandwidth.

Example 5.1 The *raised-cosine* family of Nyquist characteristics is defined by

$$Q(\Omega) = \begin{cases} T & \text{for } 0 \leq |\Omega| \leq \frac{1-\beta}{2}, \\ \frac{T}{2}[1 - \sin \frac{\pi}{\beta}(|\Omega| - \frac{1}{2})] & \text{for } \frac{1-\beta}{2} < |\Omega| < \frac{1+\beta}{2}, \text{ and} \\ 0 & \text{for } |\Omega| \geq \frac{1+\beta}{2}. \end{cases} \tag{5.2}$$

These characterics owe their name to the fact that they exhibit a cosine-shaped roll-off that is raised above the abscissa (Fig. 5.2). The *roll-off factor* $\beta \in [0, 1]$ determines the steepness of the roll-off. The excess bandwidth varies from 0 to 100 percents as β increases from 0 to 1. A Nyquist characteristic of more limited interest is

$$Q(\Omega) = \begin{cases} T(1 - |\Omega|) & \text{for } 0 \leq |\Omega| \leq 1, \text{ and} \\ 0 & \text{for } |\Omega| \geq 1. \end{cases} \tag{5.3}$$

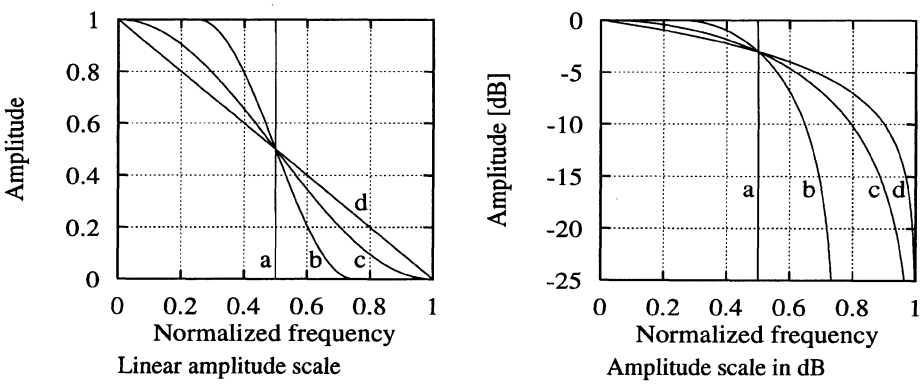


Figure 5.2: Raised-cosine characteristics with roll-off factor $\beta = 0$ (a), $\beta = 0.5$ (b) and $\beta = 1$ (c). Also depicted is the characteristic with linear roll-off according to (5.3) (d).

This function has linear roll-off and its high-frequency content exceeds that of any raised-cosine characteristic (Fig. 5.2.d). The Nyquist-1 pulses corresponding to (5.2) and (5.3) obey

$$q(t) = \frac{\sin \frac{\pi t}{T}}{\frac{\pi t}{T}} \frac{\cos \frac{\beta \pi t}{T}}{1 - (\frac{2\beta t}{T})^2} \tag{5.4}$$

and

$$q(t) = 2\frac{1 - \cos 2\pi t/T}{(2\pi t/T)^2}, \tag{5.5}$$

respectively (Fig. 5.3). The tails of the raised-cosine pulse vanish slowly for small β and much more rapidly as β approaches unity. For $\beta = 1$ there are additional zero-crossings at $|t/T| = k + 0.5$ for $k = 1, 2, \ldots$. Equivalently, $q(t - T/2)$ is a Duobinary pulse, except for a scaling factor $q(-0.5) = q(0.5) = 0.5$ (see Chapter 6 for an explanation of this terminology).

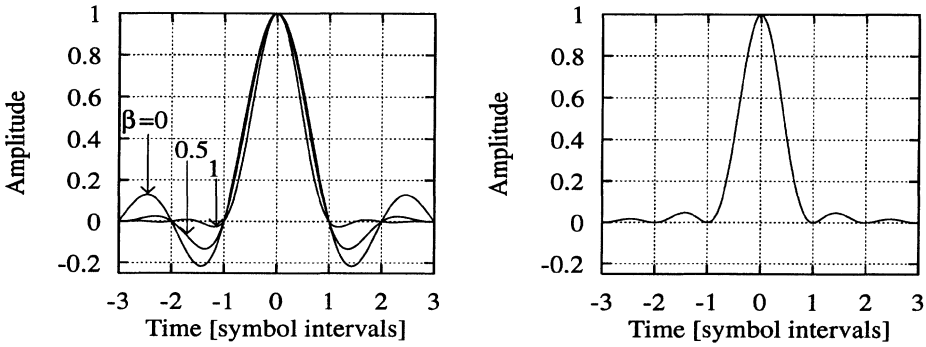


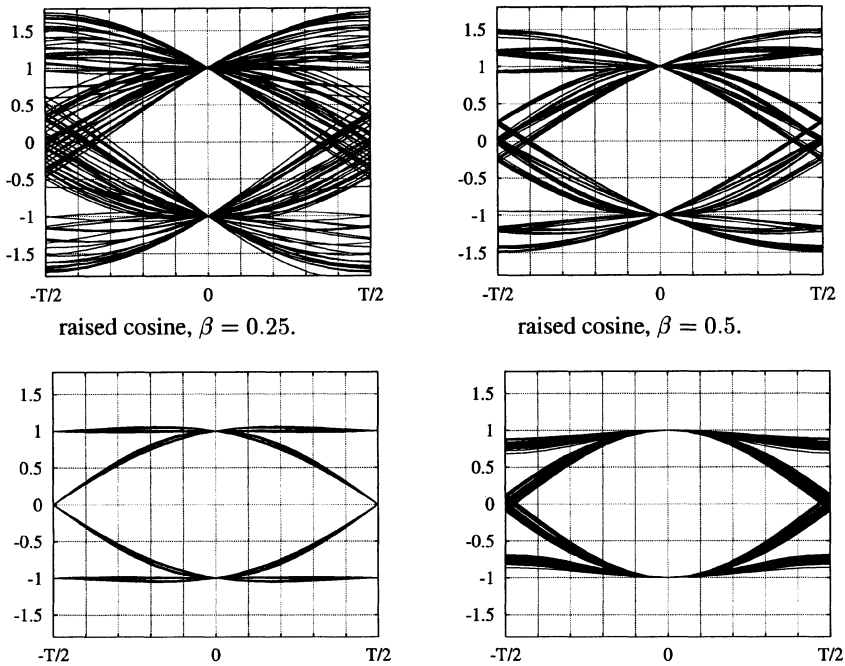
Figure 5.3: Nyquist-1 pulses: raised-cosine (left); pulse of (5.5) (right).

The Nyquist-1 pulse according to (5.5) is part of a family with tolerance to timing jitter in that the derivative of $q(t)$ is zero at the sampling instants $t = kT$ [34]. In fact, it has the smallest bandwidth of any Nyquist-1 pulse with this property [18]. Because of the zero derivative, small displacements of the time axis, as induced by small sampling-phase errors, do not effect the eye height. This is clearly visible in the eye pattern of Fig. 5.4.d, which pertains to binary uncorrelated data. Corresponding raised-cosine eye patterns are all more sensitive small to timing errors (Figs. 5.4.a-c). Their eye width declines gradually with β and vanishes for $\beta = 0$ (see Problem 5.1). One might be tempted to infer that a large roll-off parameter is always preferable. We will see that this is generally untrue for reasons of noise enhancement.

Enhancement of noise and channel instabilities To transform $H(\Omega)$ into a given Nyquist characteristic $Q(\Omega)$, the LE must have transfer function

$$W(\Omega) = \frac{Q(\Omega)}{H(\Omega)} \text{ for all } \Omega. \tag{5.6}$$

Thus $W(\Omega)$ essentially *equalizes* the channel for frequencies where $Q(\Omega)$ has significant content. This approach eliminates all ISI, but at a cost:



raised cosine, $\beta = 0.25$.

raised cosine, $\beta = 0.5$.

raised cosine, $\beta = 1$.

Nyquist-1 pulse according to (5.5).

Figure 5.4: Noiseless eye patterns for various Nyquist-1 pulses with uncorrelated binary data.

1. Any additive noise $n(t)$ in the equalizer input signal is inevitably enhanced. This is so because the LE de-emphasizes 'reliable' frequencies, i.e. those for which $|H(\Omega)|$ is large, whereas it enhances components that may in fact sink away in background noise. This problem becomes especially serious in the presence of spectral zeros, as encountered in digital magnetic recording systems with a differentiating playback head (see Section 2.3.5) and transmission systems with AC-coupling.

2. Channel instabilities at frequencies where $|H(\Omega)|$ is small are strongly amplified.

Example 5.2 At high information densities, digital magnetic recording channels may have a large transfer loss in the vicinity of the Nyquist frequency $\Omega = 0.5$. Small fluctuations in head-to-medium spacing cause this loss to vary by an amount that is significant on a relative scale but hardly noticeable in absolute terms. The equalizer amplifies such variations to an absolute level at which they can become clearly visible in the eye pattern and may well induce significant performance degradations if the equalizer is not adaptive.

In Chapters 6 and 7 we will find that more advanced reception techniques can be less sensitive to the above two problems.

The transfer function $W(\Omega)$ of (5.6) corresponds, in general, to a non-causal filter (why?). A filter of this type can be approximated in practice by adding a sufficient delay.

This causes the sampled output of the equalizer to be an estimate of some past digit a_{k-k_0} rather than of the current digit a_k. The required delay k_0 is often sufficiently small to be perfectly acceptable.

Performance With (5.6), noise $(n * w)(t)$ at the equalizer output is found to have variance

$$\begin{aligned}\varepsilon_{\text{ZF LE}} &= \frac{1}{T}\int_{-\infty}^{\infty}\mathcal{N}(\Omega)|W(\Omega)|^2 d\Omega = \frac{1}{T}\int_{-\infty}^{\infty}\frac{\mathcal{N}(\Omega)|Q(\Omega)|^2}{|H(\Omega)|^2}d\Omega \\ &= \frac{1}{T^2}\int_{-\infty}^{\infty}\frac{|Q(\Omega)|^2}{\text{SN}(\Omega)}d\Omega, \end{aligned} \tag{5.7}$$

where $\mathcal{N}(\Omega)$ is the power spectral density of $n(t)$ and $\text{SN}(\Omega) = |H(\Omega)|^2/[T\mathcal{N}(\Omega)]$ is the spectral signal-to-noise ratio of the channel as introduced in Section 3.2.4. The sampling operation in Fig. 5.1 does not affect noise variance, i.e. the noise component $u_k = (n * w)(kT)$ of the detector input $\tilde{a}_k$ also has variance $\varepsilon_{\text{ZF LE}}$. This noise is superimposed on the nominal data levels $a_k = \pm 1$ with amplitude $A = 1$. The predetection signal-to-noise ratio $\rho_{\text{ZF LE}} \triangleq A^2/\varepsilon_{\text{ZF LE}}$, therefore, amounts to

$$\rho_{\text{ZF LE}} = \frac{1}{\varepsilon_{\text{ZF LE}}} = \frac{T^2}{\int_{-\infty}^{\infty}\frac{|Q(\Omega)|^2}{\text{SN}(\Omega)}d\Omega}. \tag{5.8}$$

This ratio may be compared to the optimum predetection signal-to-noise ratio $\rho_{\text{MFB}} = \text{SNR}$ as given by the matched-filter bound (see equation (3.27) in Section 3.9). The loss in predetection signal-to-noise ratio with respect to the MFB amounts to

$$\mathcal{L}_{\text{ZF LE}} = \frac{\text{SNR}}{\rho_{\text{ZF LE}}} = \text{SNR}\,\varepsilon_{\text{ZF LE}}. \tag{5.9}$$

Since no ISI is present in the sampled output of the equalizer, this loss is entirely due to noise enhancement and depends quite heavily on the selected target characteristic $Q(\Omega)$. In Section 5.3 we will derive the characteristic that minimizes $\mathcal{L}_{\text{ZF LE}}$.

Example 5.3 In Section 2.4.3 we have modeled the behavior of optical recording channels via a Gaussian spot profile of $1/e$-width t_0. The corresponding channel transfer function (2.9) has a low-pass nature. Its bandwidth decreases rapidly as the normalized information density $S = t_0/T$ increases (see Fig. 2.35), and noise enhancement increases accordingly. The latter effect is illustrated in Fig. 5.5 for white noise $n(t)$ and target characteristics $Q(\Omega)$ as in Example 5.1. For characteristics in the raised-cosine family, the optimum roll-off factor β decreases as S increases. Ideally $Q(\Omega)$ should more or less match the channel in bandwidth. For densities of practical interest (say $S \in [0.5, 2]$), $\beta = 0.25$ and $\beta = 0.5$ are suitable roll-off factors. Values in this order are often encountered in the literature. A roll-off factor $\beta = 1$, while preferable from a jitter standpoint, causes unacceptable noise enhancement. This enhancement occurs mainly at frequencies just below the bit rate, where $|Q(\Omega)|$ is already small but the channel is so far 'down' that equalizer transfer magnitudes $|W| = |Q|/|H|$ nevertheless become

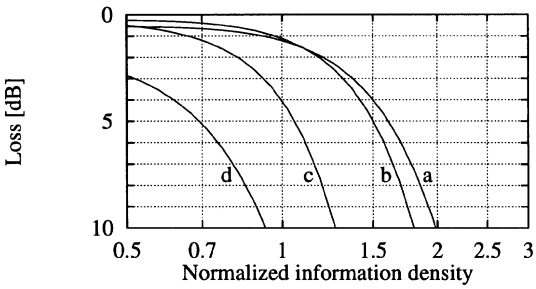
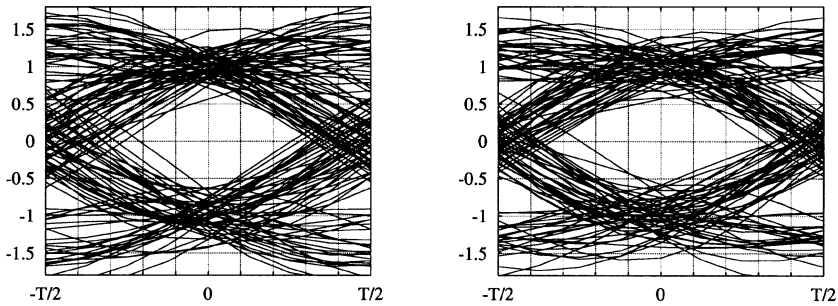


Figure 5.5: Loss $\mathcal{L}$ versus normalized information density S for zero-forcing linear equalizers with various target characteristics: a. raised cosine, $\beta = 0.25$; b: idem, $\beta = 0.5$, c: idem, $\beta = 1$; d. characteristic of (5.3).



a. $\beta = 0.25$. b. $\beta = 0.5$.

Figure 5.6: Eye patterns for a zero-forcing linear equalizer with raised-cosine target characteristics operating on an idealized optical recording channel with $S = 1.5$ and SNR $= 20$ dB.

very large. Resistance to timing jitter as provided by $Q(\Omega)$ according to (5.3) comes at an even higher expense to noise immunity. Only when excess bandwidth is plentiful (i.e. for the smallest densities in Fig. 5.5) is there effectively some freedom to trade margin against jitter off against noise margin. The optimum tradeoff is addressed in Section 5.6.

The eye patterns of Fig. 5.6 pertain to a density $S = 1.5$ and an SNR of 20 dB. They are identical to the $\beta = 0.25$ and $\beta = 0.5$ patterns of Fig. 5.4 except for noise. The patterns for $\beta = 1$ and for the characteristic of (5.5) are fully silted up by noise and are therefore not shown.

5.3 ZF LE WITH MINIMUM NOISE ENHANCEMENT

From Section 3.7 we recall that the sampled output y_k of a matched filter is a sufficient statistic for optimum detection of a_k. We can apply y_k to a discrete-time equalizer that suppresses all ISI. This equalizer is unique and can be combined with the matched filter into an equivalent continuous-time ZF LE. To demonstrate this, we show in Fig. 5.7 the

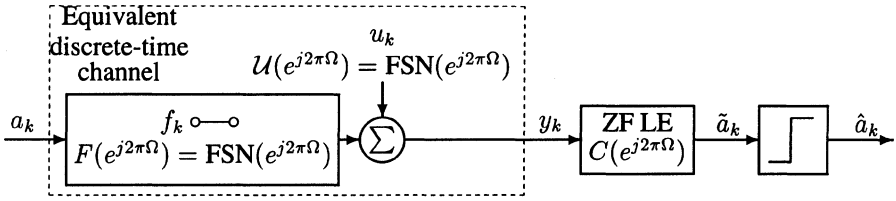


Figure 5.7: T-spaced equalizer operating on the sampled output y_k of a matched filter.

equivalent discrete-time model that was derived for y_k in Section 3.10.1, along with a discrete-time ZF LE with transfer function $C(e^{j2\pi\Omega})$.

To suppress all ISI, the channel of Fig. 5.7 should be perfectly equalized, i.e. $C(e^{j2\pi\Omega})$ should equal

$$C(e^{j2\pi\Omega}) = \frac{1}{\text{FSN}(e^{j2\pi\Omega})} \text{ for all } \Omega \tag{5.10}$$

where $\text{FSN}(e^{j2\pi\Omega})$ denotes the folded signal-to-noise ratio of the channel (see Sections 3.2.4 and 3.10.1). This transfer function is indeed unique. Unlike the continuous-time situation just studied, there is no need to specify a target characteristic $Q(\Omega)$. Fig. 5.8.a depicts the ZF LE of Fig. 5.7 along with the matched filter and sampler that precede it. Its counterpart of Fig. 5.8.b uses a continuous-time transversal filter with impulse response $c(t) = \sum_{k=-\infty}^{\infty} c_k \delta(t - kT)$ where c_k is the impulse response of the discrete-time equalizer of Fig. 5.8.a and $\delta(t)$ represents Dirac's delta function. It is not difficult to see that both receivers have the same detector input $\tilde{a}_k$ (see also Problem 1.7). The matched filter and transversal filter may be combined into a single continuous-time filter with transfer function

$$W(\Omega)C(e^{j2\pi\Omega}) = \frac{H^*(\Omega)}{\mathcal{N}(\Omega)} \frac{1}{\text{FSN}(e^{j2\pi\Omega})}. \tag{5.11}$$

This combination reduces dynamic range problems that may arise due to the fact that $|C(e^{j2\pi\Omega})|$ tends to be large at frequencies where $|W(\Omega)|$ is small and vice versa. The system transfer function corresponding to (5.11) may be written

$$Q(\Omega) = H(\Omega)W(\Omega)C(e^{j2\pi\Omega}) = \frac{|H(\Omega)|^2}{\mathcal{N}(\Omega)} \frac{1}{\text{FSN}(e^{j2\pi\Omega})} = T\frac{\text{SN}(\Omega)}{\text{FSN}(e^{j2\pi\Omega})} \tag{5.12}$$

where $\text{SN}(\Omega)$ denotes the spectral signal-to-noise ratio as introduced in Section 3.2.4. This is the unique Nyquist-1 characteristic that minimizes noise enhancement.

Exercise 5.1 Show that the characteristic of (5.12) indeed satisfies Nyquist's first criterion for distortionless transmission.

Performance From (5.10) we see that the equalizer inevitably enhances noise in introducing gain at frequencies where the folded signal-to-noise ratio is small. The noise

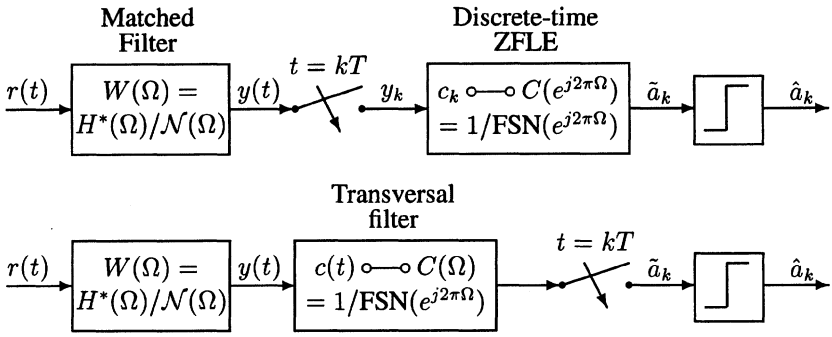


Figure 5.8: a. A matched filter and sampler transform the received signal $r(t)$ into a sufficient statistic y_k. The ZF LE of Fig. 5.7 operates on y_k to produce decision variables $\tilde{a}_k$. b. All filtering can be done in the continuous-time domain with the help of a continuous-time T-spaced transversal filter with coefficients c_k.

variance $\hat{\varepsilon}_{\mathrm{ZF\,LE}}$ of $\tilde{a}_k$ amounts to

$$\hat{\varepsilon}_{\mathrm{ZF\,LE}} = \int_{-0.5}^{0.5} \mathcal{U}(e^{j2\pi\Omega})|C(e^{j2\pi\Omega})|^2 d\Omega = \int_{-0.5}^{0.5} \frac{1}{\mathrm{FSN}(e^{j2\pi\Omega})} d\Omega. \tag{5.13}$$

The corresponding predetection signal-to-noise ratio $\hat{\rho}_{\mathrm{ZF\,LE}} = 1/\hat{\varepsilon}_{\mathrm{ZF\,LE}}$ may be recognized as the *harmonic average* of $\mathrm{FSN}(e^{j2\pi\Omega})$ that we have denoted $\overline{\mathrm{FSN}}^H$ in Section 1.5. This ratio may be compared to the optimum ratio $\rho_{\mathrm{MFB}} = \mathrm{SNR}$ for the matched-filter bound (MFB). We recall that SNR may be viewed as the *arithmetic average* of $\mathrm{FSN}(e^{j2\pi\Omega})$, i.e. $\mathrm{SNR} = \int_{-0.5}^{0.5} \mathrm{FSN}(e^{j2\pi\Omega}) d\Omega$ (see (3.7)). In the nomenclature of Section 1.5 we may write $\mathrm{SNR} = \overline{\mathrm{FSN}}^A$. Compared to the matched-filter bound, the optimum ZF LE, therefore, suffers a loss in predetection signal-to-noise ratio that may be written

$$\hat{\mathcal{L}}_{\mathrm{ZF\,LE}} = \frac{\rho_{\mathrm{MFB}}}{\hat{\rho}_{\mathrm{ZF\,LE}}} = \frac{\mathrm{SNR}}{\overline{\mathrm{FSN}}^H} = \frac{\overline{\mathrm{FSN}}^A}{\overline{\mathrm{FSN}}^H} \geq 1. \tag{5.14}$$

The inequality is a standard property of arithmetic and harmonic averages (see (1.86)). Again, since no ISI is present in $\tilde{a}_k$, the loss of (5.14) is entirely due to *noise enhancement* by the equalizer. There are two extreme cases:

1. Let $\mathrm{FSN}(e^{j2\pi\Omega})$ be independent of Ω, i.e. $\mathrm{FSN}(e^{j2\pi\Omega}) = \mathrm{SNR}$ for all Ω. Then $\hat{\mathcal{L}}_{\mathrm{ZF\,LE}} = 0$ dB, i.e. the ZF LE achieves the MFB. No other receiver can possibly do better in this case.

2. Let $\mathrm{FSN}(e^{j2\pi\Omega})$ be zero at some frequency Ω_0. Then the integral in (5.13) diverges, and $\hat{\mathcal{L}}_{\mathrm{ZF\,LE}} = \infty$.

In practice $\hat{\mathcal{L}}_{\mathrm{ZF\,LE}}$ may be anywhere in between these two extremes. The more $\mathrm{FSN}(e^{j2\pi\Omega})$ varies with Ω, the larger $\hat{\mathcal{L}}_{\mathrm{ZF\,LE}}$ will be. Thus, for channels with severe

amplitude distortion and approximately white noise, ZF linear equalization inevitably results in serious performance losses with respect to the MFB. Spectral zeros cause infinite noise enhancement. We will see in the next section that this can be avoided by deviating slightly from Nyquist's first criterion for distortionless transmission.

Example 5.4 For the Lorentzian recording channel with differentiating playback head, $\mathrm{FSN}(e^{j2\pi\Omega})$ is zero at DC ($\Omega = 0$), and the ZF LE has infinite noise enhancement.

Example 5.5 For the idealized optical recording channel of Example 5.3, $\mathrm{FSN}(e^{j2\pi\Omega})$ is a low-pass characteristic whose bandwidth decreases steadily as the normalized information density S increases (Fig. 4C.1). The harmonic average of FSN is largely determined by the minimum of FSN (see Section 1.5), which decreases rapidly with S. Losses $\hat{\mathcal{L}}_{\mathrm{ZF\,LE}}$ increase accordingly (Fig. 5.9). For densities of practical interest (say $S \in [0.5, 2]$), the $\beta = 0.25$ and $\beta = 0.5$ losses of Fig. 5.5 lag only tenths of a dB behind $\hat{\mathcal{L}}_{\mathrm{ZF\,LE}}$.

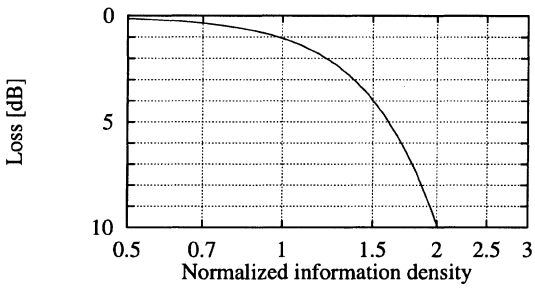


Figure 5.9: Loss $\hat{\mathcal{L}}_{\mathrm{ZF\,LE}}$ versus normalized information density S for ZF LE with minimum noise enhancement operating on the idealized recording channel of Example 5.3.

5.4 MINIMUM MEAN-SQUARE ERROR LINEAR EQUALIZER

To avoid infinite noise enhancement we must drop the requirement that all ISI be eliminated. Instead we require that the compound power of residual ISI and noise at the detector input be minimized. This is the minimum mean-square error (MMSE) criterion. We first formulate this criterion mathematically and then derive the MMSE equalizer. The system model is shown in Fig. 5.10.a. A prefilter with transfer function $W(\Omega)$ serves to suppress noise. Its output is sampled at the instants $t = kT$. The sampled sequence y_k is applied to a T-spaced discrete-time equalizer with transfer function $C(e^{j2\pi\Omega})$ whose output is a sequence of decision variables $\tilde{a}_k$. These are quantized into symbol decisions $\hat{a}_k$ by means of a slicer. Just as in Section 5.3, the prefilter is ideally a matched filter so as to ensure that y_k is a sufficient statistic for optimum detection. Again the prefilter and equalizer may be combined into a single continuous-time filter with transfer function $W(\Omega)C(e^{j2\pi\Omega})$, as explained in Section 5.3. We may describe y_k

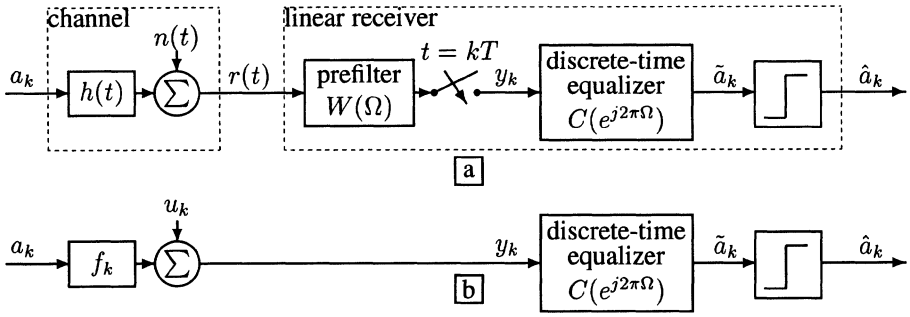


Figure 5.10: a. The output $r(t)$ of a continuous-time noisy dispersive channel is applied to a prefilter, baud-rate sampler, T-spaced equalizer and slicer. b. Equivalent discrete-time model for the input sequence y_k of the equalizer.

in terms of an equivalent discrete-time model as in Fig. 5.10.b. Here $y_k = (a * f)_k + u_k$ where $f_k \triangleq (h * w)(kT)$ and $u_k \triangleq (n * w)(kT)$. The Fourier transform $F(e^{j2\pi\Omega})$ of f_k and the power spectral density $\mathcal{U}(e^{j2\pi\Omega})$ of u_k are fully determined by the continuous-time system parameters (see Chapter 1 for a general review of this dependence). In the particular event that the prefilter is a matched filter, i.e. if $W(\Omega) = H^*(\Omega)/\mathcal{N}(\Omega)$, we recall from Section 3.10.1 that

$$F(e^{j2\pi\Omega}) = \mathcal{U}(e^{j2\pi\Omega}) = \mathrm{FSN}(e^{j2\pi\Omega}). \tag{5.15}$$

The discrete-time equalizer in Fig. 5.10 transforms y_k into a sequence $\tilde{a}_k$ that is desired to approximate a_k as well as possible. The measure of resemblance is the *mean-square error* ε between $\tilde{a}_k$ and a_k. This is the power of the error sequence $e_k \triangleq \tilde{a}_k - a_k$, i.e.

$$\varepsilon \triangleq E[e_k^2] = E[(\tilde{a}_k - a_k)^2] \tag{5.16}$$

where $E[.]$ denotes expectation across the data and noise ensembles. For the ZF LE, e_k consists purely of noise, and ε is just a filtered noise variance (cf. (5.13)). For the MMSE LE, by comparison, e_k contains residual ISI on top of noise, and the combined power of both disturbances is minimized. In this way noise enhancement can remain finite even for channel transfer functions $F(e^{j2\pi\Omega})$ with spectral zeros. Minimization of ε is easily accomplished in the frequency domain. Before doing this we first provide an intuitive explanation. Denote the PSDs of a_k and u_k by $\mathcal{A}(e^{j2\pi\Omega})$ and $\mathcal{U}(e^{j2\pi\Omega})$, respectively. Consider a given frequency Ω. At this frequency the data component of the equalizer input y_k has PSD $\mathcal{A}(e^{j2\pi\Omega})|F(e^{j2\pi\Omega})|^2$ and the noise component has PSD $\mathcal{U}(e^{j2\pi\Omega})$. There are two limiting possibilities:

1. The equalizer input consists mainly of noise ($\mathcal{A}|F|^2 \ll \mathcal{U}$). Then the best strategy is to block noise, i.e. to take $C(e^{j2\pi\Omega}) = 0$.

2. The data component overwhelms the noise ($\mathcal{A}|F|^2 \gg \mathcal{U}$). Then there is no risk of excessive noise enhancement, and we can freely choose $C(e^{j2\pi\Omega}) = 1/F(e^{j2\pi\Omega})$ so as to minimize ISI.

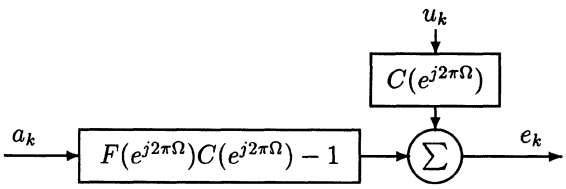


Figure 5.11: The error sequence e_k consists of two independent stochastic components: a filtered data component and a filtered noise component. The respective filters have transfer functions $F(e^{j2\pi\Omega})C(e^{j2\pi\Omega}) - 1$ and $C(e^{j2\pi\Omega})$.

The MMSE equalizer indeed has this limiting behavior. With the help of Fig. 5.10 we can model the error sequence $e_k = \tilde{a}_k - a_k$ as in Fig. 5.11. The data and noise components of e_k are formed from a_k and u_k by filters with transfer function $F(e^{j2\pi\Omega})C(e^{j2\pi\Omega}) - 1$ and $C(e^{j2\pi\Omega})$, respectively. Since data and noise are statistically independent, the power spectral density (PSD) $\mathcal{E}(e^{j2\pi\Omega})$ of e_k is the sum of the PSDs of these filtered random components, i.e.

$$\mathcal{E} = \mathcal{A}|FC - 1|^2 + \mathcal{U}|C|^2 \tag{5.17}$$

where we have suppressed the dependencies on Ω for notational convenience. The power ε of e_k is just the integral of $\mathcal{E}$ across the fundamental interval, i.e.

$$\varepsilon = \int_{-0.5}^{0.5} [\mathcal{A}|FC - 1|^2 + \mathcal{U}|C|^2] d\Omega. \tag{5.18}$$

To minimize ε we use a technique that is known as *calculus of variations*. This technique is applicable because ε is essentially a quadratic function of C, with the characteristic unique bowl-shaped minimum. Thus small variations of C around the optimum $\hat{C}$ should have a negligible influence on ε. Mathematically, this condition can expressed by writing C as $\hat{C} + \mu V$, where V is an arbitrary variation. The requirement that infinitely small variations should not influence ε reads

$$\frac{\partial \varepsilon}{\partial \mu}|_{\mu=0} = 0 \text{ for all } V(e^{j2\pi\Omega}). \tag{5.19}$$

This requirement suffices to identify $\hat{C}$. To show this, we substitute $\hat{C} + \mu V$ for C in (5.18) and evaluate the derivative of (5.19). The result is

$$\frac{\partial \varepsilon}{\partial \mu}|_{\mu=0} = 2\text{Re} \int_{-0.5}^{0.5} V^*[\mathcal{A}F^*\{F\hat{C} - 1\} + \mathcal{U}\hat{C}] d\Omega. \tag{5.20}$$

In order for this expression to be zero for all $V(e^{j2\pi\Omega})$, the quantity in square brackets should vanish for all Ω. The implication is that the MMSE LE has transfer function

$$\hat{C} = \frac{F^*}{\mathcal{U}} \frac{1}{\frac{|F|^2}{\mathcal{U}} + \frac{1}{\mathcal{A}}}. \tag{5.21}$$

Upon using this expression in (5.18) we find after some elementary manipulations that the minimum mean-square error amounts to

$$\varepsilon_{\text{MMSE LE}} = \int_{-0.5}^{0.5} \frac{1}{\dfrac{|F|^2}{\mathcal{U}} + \dfrac{1}{\mathcal{A}}} d\Omega. \tag{5.22}$$

Exercise 5.2 Verify that $\hat{C}$ has the behavior that was conjectured above for the limiting cases $\mathcal{A}|F|^2 \ll \mathcal{U}$ and $\mathcal{A}|F|^2 \gg \mathcal{U}$.

Equalized system response The equalized system has transfer characteristic

$$F\hat{C} = \frac{\dfrac{|F|^2}{\mathcal{U}}}{\dfrac{|F|^2}{\mathcal{U}} + \dfrac{1}{\mathcal{A}}} = \frac{\text{FSN}}{\text{FSN} + \mathcal{A}^{-1}}. \tag{5.23}$$

The second equality applies only if the prefilter in Fig. 5.10 is a matched filter. If the factor $1/\mathcal{A}$ were absent then a Nyquist-1 characteristic would emerge ($F\hat{C} = 1$ for all Ω). The presence of the factor $1/\mathcal{A}$ causes a deviation from Nyquist-1.

Example 5.6 Assume that F is zero at DC. Then the equalized system characteristic $F\hat{C}$ also has a zero at DC and cannot be a Nyquist-1 function. In principle, this zero causes baseline wander as described in Example 4.6. Depending on the severeness of this wander, adaptive DC-restoration may be needed in the receiver to deal with it (see Chapter 8). Alternatively, a DC-free modulation code may be used as in Example 4.8.

Continuous-time MMSE LE Ideally the prefilter in Fig. 5.10 is a matched filter. In that case we may apply (5.15) to show that the discrete-time MMSE equalizer has transfer function

$$\hat{C} = \frac{1}{\text{FSN} + \mathcal{A}^{-1}} \tag{5.24}$$

and achieves a minimum mean-square error

$$\hat{\varepsilon}_{\text{MMSE LE}} = \int_{-0.5}^{0.5} \frac{d\Omega}{\text{FSN} + \mathcal{A}^{-1}}. \tag{5.25}$$

A comparison of (5.24) and (5.25) reveals that $c_0 = \hat{\varepsilon}_{\text{MMSE LE}}$, i.e. $\hat{\varepsilon}_{\text{MMSE LE}}$ is directly reflected in the central equalizer tap. The matched filter and discrete-time equalizer can be combined into a single continuous-time equalizer with transfer function

$$W(\Omega)\hat{C}(e^{j2\pi\Omega}) = \frac{H^*(\Omega)}{\mathcal{N}(\Omega)} \frac{1}{\text{FSN}(e^{j2\pi\Omega}) + \mathcal{A}^{-1}(e^{j2\pi\Omega})}. \tag{5.26}$$

The predetection signal-to-noise ratio $\hat{\rho}_{\text{MMSE LE}} = 1/\hat{\varepsilon}_{\text{MMSE LE}}$ may be recognized as the *harmonic mean* of $\text{FSN} + \mathcal{A}^{-1}$. In the notation of Section 1.5 we may write

$$\hat{\rho}_{\text{MMSE LE}} = \overline{\text{FSN} + \mathcal{A}^{-1}}^{H} \tag{5.27}$$

Since $\mathcal{A}^{-1}$ is a non-negative function of frequency we have $\overline{\mathrm{FSN} + \mathcal{A}^{-1}}^{H} \geq \overline{\mathrm{FSN}}^{H} = \hat{\rho}_{\mathrm{ZF\,LE}}$. This shows that the MMSE LE is superior to the ZF LE in terms of predetection signal-to-noise ratio. In effect, the factor $\mathcal{A}^{-1}$ in (5.27) acts as an extra portion of signal-to-noise ratio that comes on top of the folded SNR of the channel. This portion may be used to 'mask' the poorest folded SNR's. To this end, the modulation code should produce a data spectrum that resembles $\mathrm{FSN}(e^{j2\pi\Omega})$.

Just as for the ZF LE we may compare the performance of (5.27) with the matched-filter bound:

$$\hat{\mathcal{L}}_{\mathrm{MMSE\,LE}} = \frac{\rho_{\mathrm{MFB}}}{\hat{\rho}_{\mathrm{MMSE\,LE}}} = \mathrm{SNR}\,\hat{\varepsilon}_{\mathrm{MMSE\,LE}} = \frac{\overline{\mathrm{FSN}}^{A}}{\overline{\mathrm{FSN} + \mathcal{A}^{-1}}^{H}}. \tag{5.28}$$

Bias The observant reader will note that $\hat{\mathcal{L}}_{\mathrm{MMSE\,LE}}$ can, in theory, become smaller than unity, notably at very poor SNRs. This would mean that the MMSE LE improves upon the matched-filter bound, which is clearly impossible. The principal reason for this anomaly is that we have tacitly assumed in (5.27) that the desired signal component at the equalizer output has amplitude $A = 1$, just as for the ZF LE. In reality, however, A is somewhat smaller than unity, i.e. the equalizer output is *biased.* Actual equalizer performance is, therefore, somewhat worse than predicted by (5.27) and (5.28). The bias is largest at poor SNRs and may be analyzed in a general setting for the situation that data a_k is uncorrelated (see Appendix 5A). The corresponding predetection signal-to-noise ratio is found to be

$$\hat{\rho}_{\mathrm{MMSE\,LE,unbiased}} = \overline{\mathrm{FSN} + 1}^{H} - 1 \tag{5.29}$$

and indeed cannot exceed ρ_{MFB}. At SNRs of practical interest the impact of bias is small enough for (5.27) and (5.28) to be perfectly applicable.

Bias can, in principle, be corrected by including a gain factor $1/A$ just before the detector, or, equivalently, by scaling all detection thresholds by a factor A. This can be accomplished adaptively based on the techniques of Sections 8.3.4 and 8.8.1, respectively. For binary data $a_k \in \{-1, 1\}$, decisions are taken with respect to a single threshold at DC, and bias correction clearly has no effect on bit-error rates. For multilevel data, by comparison, bias correction will, in general, improve bit-error rates somewhat, especially at poor SNRs. Further considerations in this context may be found in [8].

Effect of modulation code on performance The predetection signal-to-noise ratio of (5.27) is a harmonic average and is as such mainly determined by the minimum of $\mathrm{FSN} + 1/\mathcal{A}$. If no modulation code is used then nominally $\mathcal{A} = 1$ for all Ω. In that case $\hat{\rho}_{\mathrm{MMSE\,LE}}$ is mainly determined by the poorest values of FSN and cannot become smaller than unity. Practical bit-error-rate levels, of course, require predetection signal-to-noise ratios well beyond this lower limit.

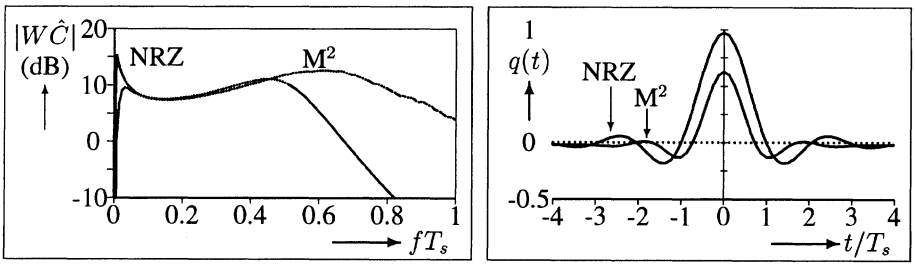
Assume that $\mathrm{FSN}(e^{j2\pi\Omega})$ has a deep depression in the vicinity of some frequency Ω_0. To mask the effect of this depression, one might use a modulation code for which

$\mathcal{A}^{-1}(e^{j2\pi\Omega}) \gg 1$ in this same vicinity. This requires that the code suppresses data frequencies near Ω_0. One might expect that this would improve $\overline{\text{FSN} + \mathcal{A}^{-1}}^H$ relative to uncoded transmission, for which $\mathcal{A}(e^{j2\pi\Omega}) = 1$. However, a nonflat data PSD can only be 'bought' at the cost of redundancy, and this inevitably causes $\text{FSN}(e^{j2\pi\Omega})$ to degrade (see Section 4.4.2 and Appendix 4C.3). In many instances the degradation will outpace any improvement that is brought about in $\mathcal{A}(e^{j2\pi\Omega})$, and the net effect of coding will be a degradation of $\hat{\rho}_{\text{MMSE LE}}$ [6].

Example 5.7 In Appendix 3A.1 we have evaluated $\text{FSN}(e^{j2\pi\Omega})$ for a Lorentzian magnetic recording channel with differentiating playback head and white noise. This function has a zero at DC, so that ZF equalization results in infinite noise enhancement. Two alternatives are:

1. The equalizer may be optimized in the mean-square sense without any shaping of the data spectrum (i.e. with $\mathcal{A}(e^{j2\pi\Omega}) = 1$). Fig. 5.12.a depicts the transfer magnitude of the MMSE equalizer for a normalized information density $D = 2$ and SNR $= 20$ dB. Rather than a full-fledged integration, the equalizer performs a pseudo-integration that stops below normalized frequencies of around 0.02. This keeps noise enhancement finite. In return, the equalized system response $q(t)$ is no longer strictly a Nyquist-1 pulse (Fig. 5.12.b). Instead it is somewhat tilted, and this leads to residual ISI in the form of baseline wander. The net achievement of the equalizer is a loss with respect to the matched-filter bound of some 5 dB.

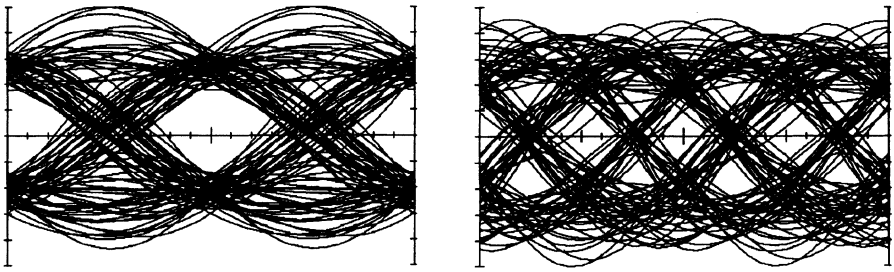
2. A modulation code with small low-frequency content may be used. This allows the equalizer to suppress low frequencies rather than enhance them, as illustrated in Fig. 5.12.a for the M^2 code of Example 4.39. M^2 is DC-free and has a band-pass power spectrum that resembles $\text{FSN}(e^{j2\pi\Omega})$. To permit a direct comparison between NRZ and M^2, frequencies in Fig. 5.12.a are normalized relative to the user bit rate $1/T_s$, and time in Fig. 5.12.b is normalized in units T_s. Since M^2 has rate $R = 1/2$, its symbol rate $1/T$ is twice as high as that of NRZ. This induces a significant loss of received data energy (see Section 4.4). As a consequence, the MSE is some 4 dB poorer than for NRZ, even though at a first glance M^2 would seem to have a favorable spectrum. This degradation is clearly visible in the eye patterns (Fig. 5.13). The M^2 eye does not exhibit significant baseline wander thanks to the fact that M^2 is DC-free. The strong correlation of M^2 induces an equalized system response $q(t)$ that deviates strongly from a Nyquist-1 pulse



a. Equalizer transfer magnitudes $|W\hat{C}|$. b. Equalized system responses $q(t)$.

Figure 5.12: Characteristics of two systems with MMSE equalizer and a Lorentzian channel with differentiating playback head, normalized information density D=2 and white noise at SNR = 20 dB.

with $T/2$-spaced zero crossings (Fig. 5.12.b). In Section 5.5 we will examine the nature of this response in more detail.



NRZ $\qquad$ M^2

Figure 5.13: Eye patterns for the equalizers of Fig. 5.12.

Equalizer tolerances In order to achieve the optimum performance of (5.25) the transfer function of the equalizer must satisfy (5.26) at every frequency Ω. This is, in general, not possible with an equalizer of finite complexity and it is of interest to know what deviations are acceptable. Qualitative insight into this matter may be gained by noting that small deviations will hardly affect the noise component of the equalizer output. Furthermore, if the data a_k has little spectral content at a given frequency Ω then deviations at this frequency will hardly be visible in the equalizer output. Conversely, if $\mathcal{A}$ is large at frequency Ω then even small deviations can have a significant impact. This qualitative argumentation suggests that required accuracy of the equalizer is proportional to $\mathcal{A}$. Relative to uncoded transmission ($\mathcal{A} = 1$), tolerances become tighter wherever $\mathcal{A} > 1$ and looser wherever $\mathcal{A} < 1$. A quantitative analysis of these effects may be based on (5.18). Illustrations may be found in Section 4.3. That section pertains to variations of the channel parameters, but is equally applicable here because it is the cascade of channel and equalizer that defines the system response. Variations of channel and equalizer, therefore, affect residual ISI equally. In Section 5.6 we will optimize the tolerance of the equalizer to a prescribed set of variations.

5.5 LINEAR EQUALIZATION FOR RUN-LENGTH-LIMITED CODES

Run-length-limited (RLL) codes have a parameter d that stipulates the minimum run-length (see Section 4.8). For a given value of d, runs of consecutive identical symbols a_k must be at least $d + 1$ symbol intervals long. An important practical case is $d = 1$. Here strings like '$- - - + + - - + + + - - -$' and '$+ + - - - - + + - - + + +$' are allowed, while strings like '$- - + - - -$' and '$+ + + + - + + - + +$

+' are forbidden (the symbols '−' and '+' are shorthand notation for $a_k = -1$ and $a_k = +1$, respectively). Examples of $d = 1$ codes are MFM, $(1,7)$ and M^2. The fact that many data sequences are forbidden allows the equalizer to deviate from Nyquist's first criterion for distortionless transmission and this, in turn, may lead to lower noise enhancement than for an equalizer that suppresses all ISI. For simplicity we describe the basic approach for an equivalent discrete-time system. Translation of the results to the original continuous-time domain is left as an exercise to the reader.

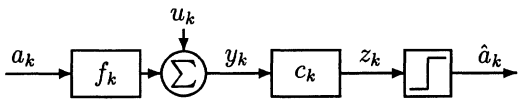


Figure 5.14: Discrete-time system with binary run-length-limited data sequence a_k, noisy dispersive channel and linear equalizer.

In Fig. 5.14, a binary data sequence a_k with minimum run-length parameter $d = 1$ is transmitted across a discrete-time channel with impulse response f_k and additive noise u_k. The channel output $y_k = (a * f)_k + u_k$ is applied to a discrete-time linear equalizer with impulse response c_k whose output may be written $z_k = (a * q)_k + (u * c)_k$ where $q_k \triangleq (f * c)_k$ is the equalized system response. Let us consider the situation that q_k is a symmetric function of k according to

$$q_k = \begin{cases} 1 & \text{for } k = 0, \\ \alpha_1 & \text{for } k = \pm 1, \\ \alpha_2 & \text{for } k = \pm 2, \text{ and} \\ 0 & \text{otherwise.} \end{cases} \tag{5.30}$$

Then the data component $(a * q)_k = a_k + \alpha_1(a_{k-1} + a_{k+1}) + \alpha_2(a_{k-2} + a_{k+2})$ of z_k is fully determined by 5 consecutive bits $a_{k-2}, \ldots, a_{k+2}$. The $d = 1$ constraint precludes various bit-combinations and the remaining combinations are summarized in Table 5.1 along with the corresponding data levels $(a * q)_k$.

$a_{k-2}, \ldots, a_{k+2}$	$(a * q)_k$	$a_{k-2}, \ldots, a_{k+2}$	$(a * q)_k$
$+ + + + +$	$1 + 2\alpha_1 + 2\alpha_2$	$- - - - -$	$-1 - 2\alpha_1 - 2\alpha_2$
$+ + + + -$	$1 + 2\alpha_1$	$- - - - +$	$-1 - 2\alpha_1$
$- + + + +$	$1 + 2\alpha_1$	$+ - - - -$	$-1 - 2\alpha_1$
$+ + + - -$	1	$- - - + +$	-1
$- - + + +$	1	$+ + - - -$	-1
$- + + - -$	$1 - 2\alpha_2$	$+ - - + +$	$-1 + 2\alpha_2$
$- - + + -$	$1 - 2\alpha_2$	$+ + - - +$	$-1 + 2\alpha_2$
$- + + + -$	$1 + 2\alpha_1 - 2\alpha_2$	$+ - - - +$	$-1 - 2\alpha_1 + 2\alpha_2$

Table 5.1: Allowable data patterns and corresponding data levels at the output of the equalizer of Fig. 5.14.

It is not difficult to see that $|(a * q)_k| \geq 1$ for all bit-combinations in Table 5.1 if and only if

$$\alpha_1 \geq -\alpha_2 \geq 0. \tag{5.31}$$

By comparison, for a system without ISI (i.e. for $\alpha_1 = \alpha_2 = 0$) we would have $|(a * q)_k| = |a_k| = 1$. Thus condition (5.31) ensures that data levels at the input of the detector will be at least as large as for a channel without ISI. In effect, the condition ascertains that only *constructive* ISI can arise, i.e. ISI that improves the eye opening relative to a system without ISI. Destructive ISI, on the other hand, is precluded by the $d = 1$ constraint.

Exercise 5.3 Consider bit patterns that are precluded by the $d = 1$ constraint and assume that condition (5.31) is met. Show that several of these patterns can yield data levels $(a * q)_k$ of magnitude $|(a * q)_k| < 1$.

It follows that we are free to deviate from Nyquist's first criterion for distortionless transmission as long as we remain within the confines of condition (5.31). This freedom can be used to arrive at a lower noise enhancement than for a Nyquist-1 system. It is worth noting that the condition precludes system responses q_k that are DC-free. For this reason there is still insufficient freedom to avoid infinite noise enhancement for a channel that blocks DC. The scheme of Fig. 5.14 may be extended with a feedback detector to overcome this problem (see Section 6.6). Alternatively, one may allow some of the components q_k for $|k| > 2$ to be nonzero as in MMSE equalization. This will yield residual ISI but will also allow q_k to be DC-free so as to avoid infinite noise enhancement. The equalized system response for M^2 of Fig. 5.12.b is a case in point.

Exercise 5.4 Verify that this response meets condition (5.31).

The approach under consideration is closely related to partial-response techniques (see Chapter 6), and sometimes works quite well (see, for example, [37], where $\alpha_1 = 0.5$ and $\alpha_2 = 0$; see also Example 7.7 and Problem 7.10). Heuristically speaking, one should choose α_1 and α_2 in such a manner as for the equalizer to approach a matched filter as well as possible. A design procedure is easily developed for the situation that α_2 is constrained to be zero. Then the only degree of freedom left is α_1, and one readily verifies that the noise variance σ^2 at the output of the equalizer may be expressed as

$$\sigma^2 = \int_{-0.5}^{0.5} \frac{\mathcal{U}(e^{j2\pi\Omega})}{|F(e^{j2\pi\Omega})|^2}(1 + 2\alpha_1 \cos 2\pi\Omega)^2 d\Omega. \tag{5.32}$$

The value of α_1 that minimizes σ^2 may be identified by differentiating σ^2 with respect to α_1 and is positive, coarsely speaking, whenever $|F|^2/\mathcal{U}$ is a low-pass characteristic (Problem 5.10). Should the optimizing value of α_1 be negative then condition (5.31) is not met, and resort must be taken to regular full-response equalization (for which $\alpha_1 = 0$). A similar development is possible for non-zero values of α_2.

5.6 RESISTANCE TO CHANNEL PARAMETER VARIATIONS

Knowledge of the channel impulse response $h(t)$ can, in principle, be obtained by means of adaptive techniques (see Chapter 8). Adaptation may, however, be too slow to track rapid variations of $h(t)$, as induced, for example, by fluctuations of tape-to-head contact in digital magnetic recording. Even slow variations cannot be tracked by non-adaptive equalizers, which are sometimes preferred for reasons of simplicity. In all of these cases, $h(t)$ will traverse some set $\mathcal{H}$ of possible responses that will often be approximately known *a priori*. We illustrate this for a particularly significant type of variations, viz. jitter.

Example 5.8 In Fig. 5.15.a, sampling nominally occurs at the instants kT, but the actual instants $(k+\Delta)T$ are displaced by a delay ΔT that accounts for jitter. In Fig. 5.15.b, sampling occurs at the nominal instants and Δ is thought of as an unknown delay that arises in the channel, i.e. as a channel parameter variation. Both ways of viewing jitter are equivalent as long as Δ is fixed or varies slowly with respect to the symbol rate. In practice it may be approximately known which values of Δ are likely to occur, and with what probability. Mathematically, this prior knowledge takes the form of a *jitter probability density function* $p(\Delta)$. In terms of $p(\Delta)$, $\mathcal{H}$ is the collection of all functions $h(t-\Delta T)$ with nonzero probability, i.e. with $p(\Delta) > 0$. An equalizer that has been designed for optimum performance at $\Delta = 0$ may perform poorly for $\Delta \neq 0$ when the eye width is small.

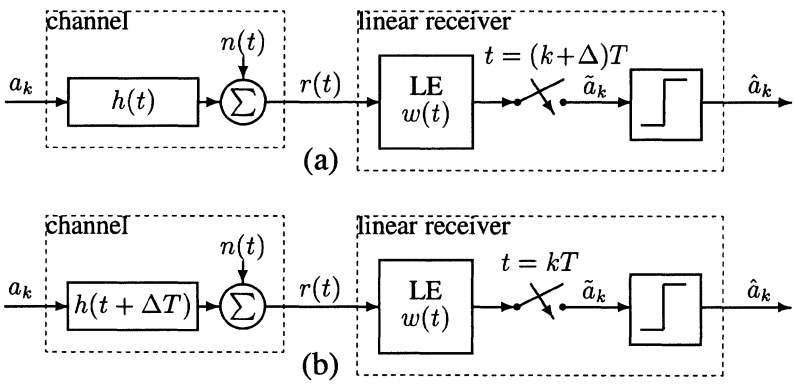


Figure 5.15: a. Timing jitter arises when the output of a linear equalizer is not sampled at the nominal instants $t = kT$ but rather at $(k+\Delta)T$ for some normalized timing-error Δ. b. Jitter can equivalently be thought of as arising in the channel. The symbol response of the channel is shifted over ΔT seconds, and sampling again occurs at the nominal instants $t = kT$.

The notions of this example are easily generalized to variations of arbitrary channel parameters. Let Δ represent the parameter that may vary (such as bandwidth, gain, sampling phase), and denote by $h^{\Delta}(t)$ the channel impulse response that corresponds to a given value of Δ. Let $\mathcal{H}$ be the set of all possible responses $h^{\Delta}(t)$, and denote by $p(\Delta)$

the probability density mass of $h^{\Delta}(t)$. Denote by S the set of all permissible values of Δ, so that

$$\int_{\Delta \in S} p(\Delta) d\Delta = 1. \tag{5.33}$$

In terms of $p(\Delta)$, we can define an *average channel impulse response* $\overline{h}(t)$ according to

$$\overline{h}(t) \triangleq \int_{\Delta \in S} h^{\Delta}(t) p(\Delta) d\Delta. \tag{5.34}$$

Since $h^{\Delta}(t)$ is now a stochastic variable rather than deterministic, equalizer performance should be weighted across the channel ensemble $\mathcal{H}$ against $p(\Delta)$. It is straightforward though somewhat tedious to optimize the LE with respect to a mean-square performance measure of this type [5]. As a rule, resistance to parameter variations comes cheaply in terms of added noise enhancement and/or residual ISI when excess bandwidth is plentiful. Conversely, if the channel has negligible excess bandwidth then tolerance to parameter variations can only be 'bought' at a considerable expense to these two disturbances, and it may be wiser to resort to adaptive reception techniques instead.

5.7 TRANSMIT FILTERS

In some cases it is possible and desirable to perform part of the equalization by means of a transmit filter (TF). The action of this filter supplements that of the linear pulse modulator in the transmitter. The precise subdivision of tasks between modulator and TF is irrelevant to system performance, and for purposes of analysis we shall model the compound effect of both by means of a single modulator with symbol response $c_c(t)$ (Fig. 5.16). The modulator operates on the data sequence a_k and its output $s(t)$ is transmitted or recorded. The continuous-time channel has impulse response $f(t)$ and additive noise $n(t)$, and its output $r(t)$ is applied to an LE whose impulse response $w(t)$ we will initially assume to be known. A sampler that closes at instants $t = kT$ extracts decision variables $\tilde{a}_k$, which are quantized into symbol decisions $\hat{a}_k$ by means of a slicer.

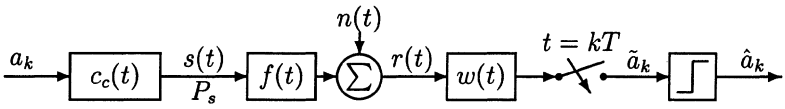


Figure 5.16: Data transmission system with transmit filter, noisy dispersive channel and linear receiver.

The purpose of the TF is very similar to that of the equalizer, and analysis and design approaches reflect this similarity. A practical difference pertains to adaptation. If the TF is to be adaptive, then a feedback path is required so that the channel characteristics can be learned at the transmitting end of the system. Such a path is not often available and for this reason TFs are usually fixed. A fundamental difference between TF and LE

is that the transmitted signal $s(t)$ is generally required to satisfy one or more constraints. These constraints have various origins:

1. Physical transmitters cannot produce infinitely large signals $s(t)$, nor can physical channels accept such signals. For this reason there is generally a constraint on the peak amplitude or power of $s(t)$.

2. In cable transmission, high-frequency components of $s(t)$ induce crosstalk in neighboring wire pairs. To limit this crosstalk, the power spectral density of $s(t)$ is sometimes constrained to be below a prescribed low-pass profile.

3. Digital recording channels often accept only binary signals $s(t)$.

The latter constraint is clearly the severest one and leaves little freedom to choose $c_c(t)$ beyond NRZ-type signaling, linear binary modulation codes such as Bi-phase and Quadra-phase, and binary write equalization schemes as in Section 4.10. For systems in which $s(t)$ can assume a continuum of amplitude levels, one commonly adopts the power constraint on $s(t)$ by virtue of its tractability. In our discussion of transmit filters we shall adhere to this convention. We first consider the optimization of the TF for a given channel and LE, and subsequently consider the joint optimization of TF and LE.

Structure of the optimum transmit filter Noise statistics at the output of the LE do not depend on $c_c(t)$ and TF optimization is, therefore, primarily a matter of maximizing the desired signal component at the LE output while minimizing ISI and satisfying the power constraint on $s(t)$. Depending on whether the ZF or MMSE criterion is adopted, ISI can either be forced to zero or be minimized in power. We first consider the ZF criterion. Here the system response $q(t) = (c_c * f * w)(t)$ will be a scaled Nyquist-1 function whose amplitude is to be maximized for a prescribed power P_s of $s(t)$. We can equivalently minimize P_s across all system responses $q(t)$ that are Nyquist-1 functions with unit amplitude. This minimization of a filtered data variance is fully analogous to the minimization of a filtered noise variance that underlies the ZF LE with minimum noise enhancement. Not surprisingly, then, the optimum TF also consists of a matched filter (MF) that is followed by a T-spaced transversal filter. The MF is now matched to channel and LE, and serves to optimize transmit power rather than signal-to-noise ratio so that it does not depend on the noise PSD. Its transfer function $F^*(\Omega)W^*(\Omega)$ reflects these facts. This same MF forms part of the MMSE TF (Appendix 5B).

ZF and MMSE transmit filters Put $C_c(\Omega) = F^*(\Omega)W^*(\Omega)C_t(e^{j2\pi\Omega})$ where C_t is the transfer function of the transversal part of $C_c(\Omega)$ and remains to be determined. The transfer function of the system up until the sampler is

$$Q(e^{j2\pi\Omega}) = \frac{1}{T}\sum_{n=-\infty}^{\infty} C_c(\Omega+n)F(\Omega+n)W(\Omega+n) = C_t(e^{j2\pi\Omega})X(e^{j2\pi\Omega}) \quad (5.35)$$

where $X(e^{j2\pi\Omega}) \triangleq T^{-1}\sum_{n=-\infty}^{\infty}|F(\Omega+n)W(\Omega+n)|^2$. For the ZF TF all ISI is to be suppressed. This requires that $Q(e^{j2\pi\Omega}) = 1$ for all Ω, i.e. that $C_t(e^{j2\pi\Omega}) =$

$1/X(e^{j2\pi\Omega})$. The corresponding transmit power is

$$P_s = \frac{1}{T^2}\int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega})|C_c(\Omega)|^2 d\Omega = \frac{1}{T}\int_{-0.5}^{0.5} \frac{\mathcal{A}(e^{j2\pi\Omega})}{X(e^{j2\pi\Omega})} d\Omega. \tag{5.36}$$

This is the smallest possible transmit power among all Nyquist-1 system characteristics. It becomes infinite when $X(e^{j2\pi\Omega})$ is zero at one or more frequencies Ω. This is analogous to the infinite noise enhancement of the ZF LE in the face of folded signal-to-noise ratios with spectral zeros.

For the MMSE TF, the optimizing transfer function $\hat{C}_t(e^{j2\pi\Omega})$ may be written $\hat{C}_t(e^{j2\pi\Omega}) = [1 - \hat{Q}(e^{j2\pi\Omega})]/\lambda$ where λ is a Lagrange multiplier that is selected in order to satisfy the constraint on P_s (see Appendix 5B). A second equation that inter-relates Q and C_t is (5.35). The joint solution of both equations is

$$\hat{C}_t(e^{j2\pi\Omega}) = \frac{1}{X(e^{j2\pi\Omega}) + \lambda} \text{ and } \hat{Q}(e^{j2\pi\Omega}) = \frac{X(e^{j2\pi\Omega})}{X(e^{j2\pi\Omega}) + \lambda}. \tag{5.37}$$

The corresponding transmit power is found to be

$$P_s = \frac{1}{T}\int_{-0.5}^{0.5} \frac{\mathcal{A}(e^{j2\pi\Omega})X(e^{j2\pi\Omega})}{[X(e^{j2\pi\Omega}) + \lambda]^2} d\Omega. \tag{5.38}$$

In the usual event that λ is strictly positive, this power cannot be infinite even if $X(e^{j2\pi\Omega})$ has spectral zeros.

Joint optimization of transmit filter and equalizer In Sections 5.3 and 5.4 we have derived optimum ZF and MMSE LEs for a given TF/channel combination. We have now also identified optimum ZF and MMSE TFs for a given channel/LE combination. These results can be combined in order to jointly optimize TF and LE. The analysis is somewhat tricky and is omitted here for brevity (see [32], [33] and [21] for details). The optimizing transfer functions of TF and LE are found to be nonzero across a range $\mathcal{R}$ of frequencies whose total width is exactly equal to $1/T$. More specifically, for every $\Omega \in [-0.5, 0.5]$, $\mathcal{R}$ contains exactly one frequency $\Omega + n$ for some $n \in \mathbb{Z}$, depending on which value of n maximizes the signal-to-noise ratio $|F(\Omega + n)|^2/\mathcal{N}(\Omega + n)$. This reflects the fact that frequencies $\Omega + n$, $n \in \mathbb{Z}$, are all equivalent from a data transmission standpoint. Of these frequencies, the optimum TF/LE combination uses the very one with best signal-to-noise characteristics and discards all the other ones. An anomaly of the solution is that $\mathcal{R}$ need not be continuous. The optimizing transfer functions can, therefore, exhibit large discontinuities and this is evidently not very practical (see [21, Section 5.3] for an illustrative example). In the event that data and noise are both uncorrelated, the optimum TF and LE have amplitude characteristics that are identical except for a scaling factor. Both are proportional to $|F(\Omega)|^{-1/2}$ throughout $\mathcal{R}$. The phase characteristics of TF and LE are arbitrary but must jointly undo any phase distortion of the channel. The cascade of TF, channel and LE, therefore, has linear phase.

A guideline that may be extracted from these considerations is that any filtering is best split equally between transmitter and receiver. This is an old adage (see [4] for an early derivation) that is often followed in wideband data transmission systems. One should bear in mind, however, that this filtering allocation is only optimum if data and noise are both uncorrelated.

Example 5.9 Let us consider a channel without any distortion, i.e. $F(\Omega) = 1$ for all Ω. Noise $n(t)$ is white and has power spectral density N_0. Data a_k is uncorrelated and has variance $\sigma_a^2 = 1$. The overall transfer function of the system is constrained to be a raised-cosine function $Q(\Omega)$ with roll-off parameter β. A fraction $\alpha \in (0,1)$ of this filtering is allocated to the TF, i.e. $C_c(\Omega) = \lambda Q^{\alpha}(\Omega)$ where λ is a constant that is chosen so as to normalize the transmit power P_s to unity. The remaining fraction $1-\alpha$ of the filtering is allocated to the LE, i.e. $W(\Omega) = \lambda^{-1}Q^{1-\alpha}(\Omega)$. For the system thus defined, we can compute the noise variance σ_u^2 at the output of the LE for any $\alpha \in (0,1)$. Fig. 5.17 shows how σ^2 depends on α for various roll-off factors β. The 0 dB level in Fig. 5.17 corresponds to the global minimum of σ_u^2, which occurs for $\alpha = 0.5$.

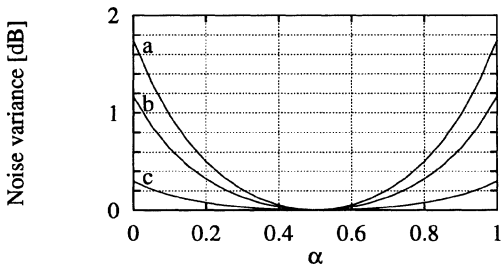


Figure 5.17: Noise variance σ_u^2 at the output of the linear equalizer versus fraction α of all filtering that is allocated to the transmitter. The channel is distortionless and has white noise. The overall system has a raised-cosine characteristic with roll-off factors $\beta = 1$ (a.), $\beta = 0.5$ (b.) and $\beta = 0.1$ (c.).

For any value of β, σ_u^2 is minimized by splitting filtering equally between transmitter and receiver. Here both transmit and receive filter are said to be of the *root raised-cosine* type. The penalty for an unequal split increases with β but even for $\beta = 1$ never exceeds 2 dB.

5.8 IMPLEMENTATION ISSUES

The equalizers and transmit filters that we have studied in preceding sections were all described in terms of impulse responses and transfer functions. We have not yet considered the problem of approximating these characteristics by means of implementable filter structures, be they analog, digital or a mixture of the two. This problem is sufficiently important to merit a brief discussion. A more thorough treatment may be found in the literature (see, for example, [16], [30], [22], [3], [7], [36], [2] and references therein).

Causality and system delay For simplicity of analysis, we have so far assumed that the cascade of transmitter, channel and receiver introduces no delay. In reality all components of the transmission system must be causal, and a strictly positive system delay is unavoidable. Such a delay is usually acceptable and can, in fact, be regarded as a design freedom. This freedom can be exploited to minimize complexity of the employed filters and optimize their performance. A careful choice of the system delay tends to pay off handsomely in both respects.

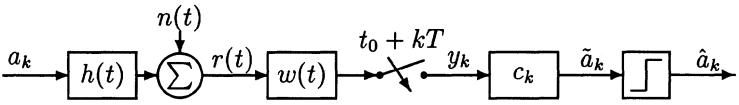


Figure 5.18: Data transmission system with T-spaced discrete-time equalizer.

Example 5.10 In Fig. 5.18, the received signal $r(t)$ is applied to an analog filter with impulse response $w(t)$ that serves to suppress out-of-band noise. The filter output is sampled at instants $t_0 + kT$ where t_0 is a design freedom that allows us to control the system delay. A T-spaced discrete-time linear equalizer with impulse response c_k operates on the sampler output to produce a sequence of decision variables $\tilde{a}_k$, which are sliced into symbol decisions $\hat{a}_k$. The system up until the sampler has a discrete-time impulse response $f_k = (h*w)(t_0+kT)$ that depends on t_0. In [5] it is shown that there is at least one value of t_0 in any interval of width T for which f_k has spectral zeros. For such a value of t_0, the discrete-time LE will suffer infinite noise enhancement when it is dimensioned according to the ZF criterion. Even when the MMSE criterion is used, its impulse response will be long and performance may still be poor. This problem occurs for virtually any function $(h*w)(t)$, including a Nyquist-1 pulse. For such a pulse, performance can be excellent even without discrete-time equalizer if sampling occurs at a proper phase.

This example is somewhat academic but clearly illustrates how dramatic the impact of t_0 on filter complexity and performance can be. A practical implication is that the performance of *adaptive* T-spaced equalizers tends to depend heavily on the sampling phase. Ideally the timing-recovery scheme would never dwell near a 'bad' phase. If the scheme does not have this property then resort may be taken to oversampling. The equalizer is then *fractionally spaced*, i.e. the tap spacing is a fraction of the symbol interval T, such as $T/2$. As a result, the equalizer can accurately compensate for any continuous-time delay, and performance depends much less heavily on sampling phase.

Structure of transmitter and equalizer Analog filters have a non-periodic transfer function that is well-suited to suppress out-of-band noise. Digital filters, by comparison, are ill-suited for this task because of their periodic transfer function. The output $s(t)$ of the transmitter is normally required to have a restricted bandwidth. This necessitates an analog back-end. Similarly, the received signal $r(t)$ generally contains wideband noise and suppression of out-of-band noise requires an analog front-end. In both cases, digital filters may or may not be present, and may or may not be oversampled with respect to the symbol rate.

Example 5.11 For the NRZ modulation code, digital filtering is absent and analog filtering is disguised as a hold operation of width T. This operation yields a sinc-shaped frequency response and as such provides suppression of out-of-band energy.

Example 5.12 A practical approximation of the root-raised-cosine system of Example 5.9 is shown in Fig. 5.19 (see also [2]).

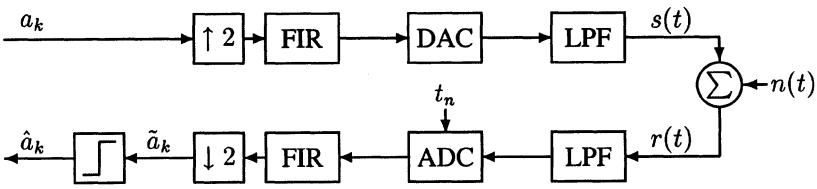


Figure 5.19: Transmission system with root-raised-cosine transmit filter and equalizer. Both are approximated by a combination of analog filter and a $T/2$-spaced digital FIR filter.

The transmitter output $s(t)$ is formed via a digital FIR filter, a digital-to-analog converter (DAC) and an analog filter. For a typical roll-off factor $\beta = 0.5$, $s(t)$ cannot have any content above 75% of the symbol rate $1/T$. The required suppression of out-of-band energy is provided in part by the hold operation that the DAC performs, and for another part by the analog LPF. That LPF is taken from a catalog and could be, for example, a seventh-order Butterworth filter with a 3 dB cut-off frequency in the vicinity of $0.5/T$. The compound effect of DAC hold function and LPF deviates from the desired root-raised-cosine characteristic, and the FIR filter is meant to compensate for this discrepancy. To this end, the filter must be able to independently manipulate frequencies up to around $0.75/T$. This is possible with an oversampling factor of 2. The FIR filter, then, has a tap spacing of $T/2$ seconds. It is designed in order for $s(t)$ to approximate a delayed root-raised-cosine output as well as possible in the mean-square sense. The delay τ is a design freedom, and is aimed at minimizing the filter length that is needed to achieve a prescribed mean-square error. For any candidate value of τ, design of the FIR filter boils down to the solution of a set of linear equations [2].

At the receiving end of the system, an analog LPF suppresses out-of-band noise. This filter is akin to the transmit-side LPF and is also taken from a catalog. The LPF transfer function deviates from the desired root-raised-cosine characteristic and this discrepancy is repaired by means of a subsequent digital FIR filter. That filter also has an oversampling factor of 2 so that it can handle discrepancies ranging from DC up to approximately $1/T$. The LPF output is, therefore, sampled at instants $t_0 + nT/2$ where t_0 defines the system delay. Selection of t_0 is aimed at minimizing the order of the FIR filter. Symbol decisions $\hat{a}_k$ are only produced once per symbol interval so that the FIR output must be down-sampled by a factor of 2. Design of the FIR filter is targeted, for example, at minimizing the mean-square error between the actual decision variables $\tilde{a}_k$ and the ones that would emerge in the ideal root-raised-cosine system. For any value of t_0, this design problem boils down to the solution of a set of linear equations (see Appendix 6B).

The transmit-side and receive-side FIR filters, while similar in nature, can have rather different complexities. The reason is that the transmit-side FIR filter operates on a data sequence a_k with only a few amplitude levels, i.e. its output can be formed without any digital multiplications. This is, in general, not possible for the receive-side FIR filter.

In the above example, filter design effort is focused on the discrete-time domain. This focus is common and stems from the fact that analog filters invariably have one or more poles, whose optimum placement is a difficult problem that is mired with local optima. Discrete-time FIR filters, by comparison, are characterized by a set of zeros and the associated performance surface tends to have no local optima. The global optimum is, moreover, easily identified by means of analytical techniques like those of Sections 5.4 and 5.7, or by adaptation techniques as in Chapter 8. One may infer that implementation of data-transmission filters in mixed analog/digital form is more a matter of convenience than one of necessity. Indeed these filters can be fully analog and this option can be quite attractive when dissipation, chip area and/or bit rates are at a premium. While no general and/or optimum solution to the design of analog data-transmission filters appears to be known, various papers do offer approximate solutions for specific categories of channels. Introductions to this subject area may be found in [16], [15], [30], [22], [25] and [3]. For a tutorial review on IC implementation of analog filters we recommend [31]. Studies pertaining to the design of digital data-transmission filters may be found in [7], [36] and [2].

Adaptive filters Fixed data-transmission filters are applicable whenever the channel characteristics are fixed and *a priori* known. While these conditions are met by assumption throughout this chapter, they are not usually met in practice. To handle variations and uncertainties of the channel characteristics, the data receiver may be equipped with a certain degree of adaptivity (see Chapter 8). This is normally accomplished with an adjustable filter structure that is controlled by an adaptation algorithm. The adjustable parameters usually define one or more zeros of the filter, while the other zeros and all poles are fixed. In Chapter 8 we will derive the optimum ZF and MMSE parameter settings for structures of this type, and we develop adaptation algorithms that converge towards these optima. A problem that is not addressed in Chapter 8 is how the fixed zeros and poles should be chosen. A first-order approach towards this problem consists of two steps:

1. Define the nominal locations of the adjustable zeros, i.e. the desired locations for the nominal channel conditions. In essence, this choice serves to match the achievable range of filter transfer functions to the possible range of channel parameter variations. (If a good match cannot be achieved, then nature and/or degree of adaptivity are inadequate, and either or both must be amended.)

2. Assume that channel conditions and zero locations are both nominal. The remaining zeros and the poles are designed relative to this assumption. The resulting design problem is equivalent to the design of fixed data-transmission filters as discussed above, and methods of solution are similar.

Further considerations in this context may be found in Chapter 8.

5.9 FURTHER READING

The origins of linear equalization lie in the annals of telegraphy. Long-range telegraphy networks were in use in both Europe and the USA by the middle of the 19th century, and the first permanent transatlantic connection was completed as early as 1866. Bandwidths of long connections were so small that signaling rates were measured in symbols per minute. It was soon discovered that long cables could be 'loaded' with inductance so as to improve their frequency response. This is a form of equalization *avant la lettre*. The notion of equalization stems back to the work of Küpfmüller and Nyquist [19], [23] and probably earlier. In 1928 Nyquist formulated his classical first criterion for distortionless transmission [24]. This criterion defines a class of system characteristics that all result in zero ISI. Within this class there is a single characteristic that yields minimum noise enhancement and this characteristic was not identified until 1965. Tuft's seminal paper on this subject also identifies the minimum mean-square error linear equalizer [32]. The fractionally-spaced equalizer was first described by Guidoux [13] and was analyzed in detail by Ungerboeck [35], Gitlin and Weinstein [12] and others.

Extensions of the linear equalizer to more general channel models than the one considered here are developed in [29], [17], [10], [28], [14], [9] and references therein. Specifically, additive disturbances are sometimes cyclostationary rather than stationary (see Chapter 2) and can then be captured in an extended channel model with multiple inputs and a single output. Exploitation of cyclostationarity can be attractive when data and disturbance components of the channel output both have significant excess bandwidth, but the advantage quickly erodes when this is not true. In fact, if the disturbances have negligible excess bandwidth then they effectively become stationary (see Appendix 2C).

Linear equalization has long been a cornerstone of data reception and is still widely used. Various books [4], [21], [26], [20] and a review article [27] testify to this prominence. Other basic references on the subject include [24], [32], [11], and [1]. Historical perspectives may be found in [4] and [32]. In the past few decades, emphasis in the literature has shifted gradually to more sophisticated reception techniques, several of which will be discussed in the chapters ahead.

APPENDIX 5A

Effect of bias on receiver performance

This appendix follows the line of thought laid down in [8]. Fig. 5A.1 depicts a generic model of a system that exhibits bias. A detector input $\tilde{a}_k$ consists of a desired data component βa_k and a disturbance component βn_k where a_k is the data sequence that is to be detected and n_k accommodates both noise and residual intersymbol interference. The signal suppression factor β accounts for bias and amounts to unity for an unbiased receiver. An error e_k is formed by subtracting a_k from $\tilde{a}_k$ and in the unbiased case e_k

has variance

$$\varepsilon_{\mathrm{U}} = \sigma_n^2 \tag{5A.1}$$

where σ_n^2 denotes the variance of n_k. In an MMSE receiver β tends to be smaller than unity and the corresponding error variance ε_{B} will be smaller than ε_{U}. Even so, bit-error performance will, in general, be somewhat worse than than for an unbiased system (see Section 5.4). A direct implication is that ε_{U} is a more accurate performance indicator than ε_{B}.

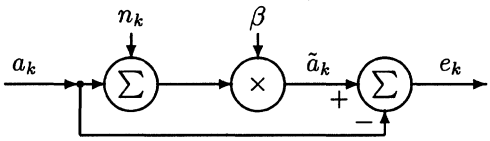


Figure 5A.1: Generic model of a system that exhibits bias.

It is easy to interrelate ε_{U} and ε_{B} in the event that data a_k is uncorrelated. In this case the residual ISI component of n_k will have no cross-correlation with a_k and by assumption this also applies to the noise component of n_k. The variance ε of e_k may then be expressed as

$$\varepsilon = (1-\beta)^2\sigma_a^2 + \beta^2\sigma_n^2 \tag{5A.2}$$

where σ_a^2 denotes the variance of a_k. In order to find ε_B we require that the derivative of ε with respect to β vanishes. The optimizing signal suppression factor $\hat{\beta}$ is

$$\hat{\beta} = \frac{\sigma_a^2}{\sigma_a^2 + \sigma_n^2} \tag{5A.3}$$

and the corresponding error variance is

$$\varepsilon_{\mathrm{B}} = \frac{\sigma_a^2}{\sigma_a^2 + \sigma_n^2}\sigma_n^2 = \hat{\beta}\,\varepsilon_{\mathrm{U}}. \tag{5A.4}$$

The predetection signal-to-noise ratios $\rho_{\mathrm{B}} \triangleq \sigma_a^2/\varepsilon_{\mathrm{B}}$ and $\rho_{\mathrm{U}} \triangleq \sigma_a^2/\varepsilon_{\mathrm{U}}$ for the unbiased and biased situations are interrelated according to $\rho_{\mathrm{U}} = \hat{\beta}\rho_{\mathrm{B}}$ and upon re-expressing $\hat{\beta}$ as $\hat{\beta} = \rho_{\mathrm{U}}/(\rho_{\mathrm{U}}+1)$ it follows that

$$\rho_{\mathrm{U}} = \rho_{\mathrm{B}} - 1. \tag{5A.5}$$

This confirms that bias can only be removed at an expense to predetection signal-to-noise ratio. For ratios of practical interest the cost is small.

APPENDIX 5B

Structure of the MMSE transmit filter

The decision variable $\tilde{a}_k$ in Fig. 5.16 may be written $\tilde{a}_k = (a*q)_k + u_k$ where $q_k \triangleq (c_c * f*w)(kT)$ is the sampled system response and $u_k \triangleq (n*w)(kT)$ is sampled and filtered noise. The impulse response of the LE is fixed and u_k, therefore, has a fixed variance that we denote σ_u^2. We seek to identify the structure of the impulse response $\hat{c}_c(t)$ that minimizes the mean-square error $\varepsilon \triangleq E[e_k^2]$ where $e_k \triangleq \tilde{a}_k - a_k$. This minimization is subject to a constraint on the transmit power $P_s = E[s^2(t)]$. In frequency-domain notation we have

$$\begin{aligned}\varepsilon &= \sigma_u^2 + \int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega})|Q(e^{j2\pi\Omega}) - 1|^2 d\Omega \\ &= \sigma_a^2 + \sigma_u^2 + \int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega})[|Q(e^{j2\pi\Omega})|^2 - 2\mathrm{Re}\{Q(e^{j2\pi\Omega})\}]d\Omega \quad (5B.1)\end{aligned}$$

where $\mathcal{A}(e^{j2\pi\Omega})$ is the data PSD and $\sigma_a^2 = E[a_k^2]$ denotes the data variance. Now $Q(e^{j2\pi\Omega})$ is a folded version of the system transfer function $C_c(\Omega)F(\Omega)W(\Omega)$ and upon exploiting the periodicity of $\mathcal{A}(e^{j2\pi\Omega})$ one establishes after some elementary manipulations that ε may be re-expressed as

$$\begin{aligned}\varepsilon = \sigma_a^2 + \sigma_u^2 + \int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega})[\frac{1}{T^2} C_c(\Omega)F(\Omega)W(\Omega) \\ \sum_{m=-\infty}^{\infty} C_c(\Omega+m)F(\Omega+m)W(\Omega+m) - \frac{2}{T}\mathrm{Re}\{C_c(\Omega)F(\Omega)W(\Omega)\}]d\Omega. (5B.2)\end{aligned}$$

In similar vein the transmit power may be expressed as

$$P_s = \frac{1}{T^2}\int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega})|C_c(\Omega)|^2 d\Omega. \quad (5B.3)$$

The constraint on P_s may be added via a Lagrange multiplier λ, and we must minimize the Lagrangian $L \triangleq \varepsilon + \lambda T P_s$. (The factor T is included to make λ dimensionless.) Denote $C_c(\Omega) = \hat{C}_c(\Omega) + \mu V(\Omega)$ where $\hat{C}_c(\Omega)$ is the transfer function of the optimum transmit filter and $V(\Omega)$ is an arbitrary variation. In order to identify $\hat{C}_c(\Omega)$ we employ the calculus of variations, i.e. we evaluate $\partial L/\partial\mu|_{\mu=0}$ and require this derivative to vanish for all $V(\Omega)$. With the help of (5B.2) and (5B.3) it is not difficult to show that

$$\frac{\partial L}{\partial \mu}|_{\mu=0} = \frac{2}{T}\mathrm{Re}\int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega})V^*(\Omega)[F^*(\Omega)W^*(\Omega)(\hat{Q}(e^{j2\pi\Omega}) - 1) + \lambda\hat{C}_c(\Omega)]d\Omega \quad (5B.4)$$

where $\hat{Q}(e^{j2\pi\Omega}) \triangleq \sum_{m=-\infty}^{\infty} \hat{C}_c(\Omega + m)F(\Omega + m)W(\Omega + m)$ denotes the optimum system characteristic. In order for this derivative to vanish for all $V(\Omega)$ it is sufficient that the term in square brackets is zero for all Ω. It follows that

$$\hat{C}_c(e^{j2\pi\Omega}) = F^*(\Omega)W^*(\Omega)\frac{1 - \hat{Q}(e^{j2\pi\Omega})}{\lambda}. \tag{5B.5}$$

This may be recognized as the cascade of a matched filter with transfer function $F^*(\Omega)W^*(\Omega)$ and a T-spaced transversal filter with transfer function $[1 - \hat{Q}(e^{j2\pi\Omega})]/\lambda$. The same structure can be derived for the ZF transmit filter with minimum noise enhancement. All that differs is the transfer function of the transversal filter.

PROBLEMS

5.1 In this problem we are concerned with the eye width of a full-response system with uncorrelated data $a_k \in \{-1, 1\}$ and no excess bandwidth. A system of this type has system response $q(t) = [\sin(\pi t/T)]/[\pi t/T]$.

1. Denote by $q_k^\Delta \triangleq q((k + \Delta)T)$ the system response when sampled with a normalized sampling phase error Δ. Use the periodicity of the sine function along with a series expansion to find an approximation for $|q_k^\Delta|$ when $|\Delta|$ is small.
2. The worst-case ISI $I_{\max}$ that can emerge due to a sampling phase error Δ is made up of all unintended contributions q_k^Δ $(k \neq 0)$ according to $I_{\max} = \sum_{k\neq 0} |q_k^\Delta|$. Find an approximation to $I_{\max}$ for small $|\Delta|$. Is there any eye opening?

5.2 This problem considers transmission of NRZ across a distortionless channel with noise. The received signal $r(t)$ is assumed to be of the form $r(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT) + n(t)$ where $n(t)$ is white Gaussian noise with power spectral density N_0 and $h(t)$ is the NRZ pulse

$$h(t) = \begin{cases} 1 & \text{for } |t| < T/2, \text{ and} \\ 0 & \text{for } |t| \geq T/2. \end{cases} \tag{5P.1}$$

1. Derive the continuous-time ZF LE with minimum noise enhancement for this system. What is the continuous-time impulse response of the LE? How can this LE be interpreted? Show that the scheme can be implemented with an integrate-and-dump circuit. What is the loss with respect to the matched-filter bound?
2. Assume that $r(t)$ is equalized into a raised-cosine characteristic as in Section 5.2. Determine the resulting loss with respect to the matched-filter bound for roll-off factors $\beta = 0$ and $\beta = 1$.

5.3 Assume that the received signal $r(t)$ is of the form $r(t) = \sum_{k=-\infty}^{\infty} a_k h(t - kT) + n(t)$ where a_k is the data sequence, $h(t)$ is a pulse whose width is at most T seconds, and $n(t)$ is white Gaussian noise. Prove that the ZF LE with minimum noise enhancement for this system achieves the matched-filter bound.

5.4 Redo Problem 5.2 for the Twinned-Binary pulse

$$h(t) = \begin{cases} 1 & \text{for } t \in (-T, 0), \\ -1 & \text{for } t \in (0, -T), \text{ and} \\ 0 & \text{for } |t| \geq T. \end{cases} \tag{5P.2}$$

Is zero-forcing linear equalization practicable? Does the conclusion change if noise would not be white but would instead have a $1/\Omega^2$ power spectrum?

5.5 Consider a system with uncoded binary data $a_k \in \{-1, 1\}$ and a noisy dispersive channel with transfer function

$$H(\Omega) = \begin{cases} T[1 - |\Omega|] & \text{for } |\Omega| < 1 \text{ and} \\ 0 & \text{for } |\Omega| \geq 1. \end{cases} \tag{5P.3}$$

Noise is Gaussian and white with power spectral density N_0.

1. Determine the folded signal-to-noise ratio $\text{FSN}(e^{j2\pi\Omega})$ and the average signal-to-noise ratio SNR of the channel.
2. Does the channel satisfy Nyquist's first criterion for distortionless transmission?
3. A conceptually straightforward zero-forcing receiver for this channel uses an ideal low-pass filter to suppress the out-of-band noise and a subsequent slicer with zero threshold level. What is the bandwidth of the low-pass filter? What is the noise variance σ^2 at the input of the slicer? What is the predetection signal-to-noise ratio?
4. Can the performance of this receiver be improved upon by means of a zero-forcing linear equalizer? If so, by how much? Hint: Make use of the integral identity

$$\int_0^{0.5} \frac{dx}{2x^2 - 2x + 1} = \frac{\pi}{4}. \tag{5P.4}$$

5. Is a further performance improvement fundamentally possible? If so, by how much?
6. Determine the transfer function $W(\Omega)$ of the optimum ZF LE and the equalized system characteristic $H(\Omega)W(\Omega)$, and compare these to their counterparts for the 'straightforward' receiver under 3.

5.6 Consider a discrete-time channel whose impulse response f_k has $\mathcal{D}$-transform $F(\mathcal{D}) = 1 + \alpha\mathcal{D}$ for some $\alpha \in [-1, 1]$. The channel has output $y_k = (a * f)_k + u_k$ where a_k is the transmitted data sequence and u_k is white Gaussian noise of variance $N_0 = 1$.

1. Compute the channel transfer function $F(e^{j2\pi\Omega})$ and the folded signal-to-noise ratio $\text{FSN}(e^{j2\pi\Omega}) = |F(e^{j2\pi\Omega})|^2/N_0$. Sketch $\text{FSN}(e^{j2\pi\Omega})$ for $\alpha = -1, -0.5, 0, 0.5$ and 1.
2. Compute the average signal-to-noise ratio $\text{SNR} = \int_{-0.5}^{0.5} \text{FSN}(e^{j2\pi\Omega})d\Omega$.
3. Assume that a discrete-time ZF LE operates on y_k. Compute the loss in predetection signal-to-noise ratio of this LE with respect to the matched-filter bound. For which values of α does the loss become infinite? Explain why this is so.
   Hint: Make use of the integral identity

$$\int_{-0.5}^{0.5} \frac{dx}{a + b\cos 2\pi x} = \frac{1}{\sqrt{a^2 - b^2}}. \tag{5P.5}$$

5.7 This problem is concerned with the amplitude A of the desired signal component at the output of an MMSE equalizer. For simplicity we restrict attention to uncorrelated binary data ($\mathcal{A}(e^{j2\pi\Omega}) = 1$, $a_k \in \{-1, 1\}$).

1. How does A depend on the equalized system response q_k?
2. Assume that an MMSE equalizer is used so that q_k is the inverse Fourier transform of the system characteristic $F(e^{j2\pi\Omega})\hat{C}(e^{j2\pi\Omega})$ of (5.23). Assume further that the prefilter in Fig. 5.10 is a matched filter so that the second equality in (5.23) applies. Use (5.23) along with (5.25) to show that $A = 1 - \hat{\varepsilon}_{\text{MMSE LE}}$. Verify that this value is compatible with that of (5A.3).
3. Mean-square errors of practical interest are well below -10 dB and are more typically in the order of -20 dB. What values of A does this correspond to? How much worse (in dB) are the corresponding predetection signal-to-noise ratios of the unbiased MMSE LE as compared to its biased counterpart?

5.8 Consider a discrete-time $1 - \mathcal{D}$ channel with white additive noise u_k of variance N_0. Data is binary and uncorrelated with $a_k \in \{-1, 1\}$.

1. Is zero-forcing linear equalization practicable?
2. Assume that we use a discrete-time equalizer with transfer function

$$C(e^{j2\pi\Omega}) = \begin{cases} 0 & \text{for } |\Omega| < \alpha \text{ and} \\ \frac{1}{F(e^{j2\pi\Omega})} & \text{for } \alpha < |\Omega| \leq 0.5, \end{cases} \tag{5P.6}$$

   where $F(e^{j2\pi\Omega})$ is the transfer function of the channel and $\alpha \in [0, 0.5]$ is a constant. Determine the noise power σ_n^2, the ISI power σ_I^2 and the mean-square error $\varepsilon = \sigma_n^2 + \sigma_I^2$ at the output of this equalizer as a function of α.
   Hint: make use of the integral identity $\int \sin^{-2}(ax)dx = -a^{-1}\cot(ax)$.
3. Assume that the channel has SNR $= 20$ dB. Which value of α minimizes the mean-square error? How large is the MMSE? Is this a practical value?

5.9 Consider the $1 - \mathcal{D}$ system of Problem 5.8. Assume that the channel is equalized by means of the discrete-time MMSE LE of Section 5.4. Determine the transfer function and MSE of this LE for SNR $= 20$ dB, and compare these with the results of Problem 5.8.
Hint: Make use of the integral identity

$$\int \frac{dx}{1 + q^2 \sin^2(ax)} = \frac{1}{a\sqrt{1+q^2}} \tan^{-1}[\sqrt{1+q^2}\tan(ax)]. \tag{5P.7}$$

5.10 Consider the $d = 1$ LE of Section 5.5 for the situation that α_2 is constrained to be zero. Find the value $\hat{\alpha}_1$ of α_1 that minimizes the noise variance σ^2 at the output of the equalizer. When is $\hat{\alpha}_1$ positive?

5.11 Consider the $d = 1$ LE of Section 5.5. Determine the number of data levels of the eye pattern at the detector input for $\alpha_1 > -\alpha_2 > 0$, $\alpha_1 = -\alpha_2 > 0$, and $\alpha_1, \alpha_2 > 0$.

5.12 The Compact Disc system uses a modulation code called EFM with minimum-run-length parameter $d = 2$ (see Section 4.8 for an explanation of this terminology). Redo the analysis of Section 5.5 for a code with $d = 2$. How does condition (5.31) become in this case? Verify that this new condition is less restrictive than the one for $d = 1$.

REFERENCES

[1] M.R. Aaron and D.W. Tufts, 'Intersymbol Interference and Error Probability', *IEEE Trans. Inform. Theory*, Vol. IT-12, No. 1, pp. 26-34, Jan. 1966.

[2] A.N. D'Andrea, F. Guglielmi, U. Mengali and A. Spalvieri, 'Design of Transmit and Receive Digital Filters for Data Communications', *IEEE Trans. Commun.*, Vol. 42, No. 2/3/4, pp. 357-359, Feb./March/Apr. 1994.

[3] H. Baher and J. Beneat, 'Design of Analog and Digital Data Transmission Filters', *IEEE Trans. Circuits Syst.*, Vol. 40, No. 7, pp. 449-460, Jul. 1993.

[4] W.R. Bennett and J.R. Davey, *Data Transmission*. New York: McGraw-Hill, 1965.

[5] J.W.M. Bergmans and A.J.E.M. Janssen, 'Robust Data Equalization, Fractional Tap Spacing and the Zak Transform', *Philips J. Res.*, Vol. 42, No. 4, pp. 351-398, 1987.

[6] J.W.M. Bergmans, 'SNR Merits of Binary Modulation Codes in Equalized Digital Recording Systems', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 157-167, Jan. 1992.

[7] P.R. Chevillat and G. Ungerboeck, 'Optimum FIR Transmitter and Receiver Filters for Data Transmission over Band-Limited Channels', *IEEE Trans. Commun.*, Vol. COM-30, No. 8, pp. 1909-1915, Aug. 1982.

[8] J.M. Cioffi, G.P. Dudevoir, M.V. Eyuboğlu, and G.D. Forney, Jr., 'MMSE Decision-Feedback Equalizers and Coding - Part I: Equalization Results', *IEEE Trans. Commun.*, Vol. 43, No. 10, pp. 2582-2594, Oct. 1995.

[9] A. Duel-Hallen, 'Equalizers for Multiple Input/Multiple Output Channels and PAM Systems with Cyclostationary Input Sequences', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 3, pp. 630-639, Apr. 1992.

[10] W. van Etten, 'An Optimum Linear Receiver for Multiple Channel Digital Transmission Systems', *IEEE Trans. Commun.*, Vol. COM-23, No. 8, pp. 828-834, Aug. 1975.

[11] R.A. Gibby and J.W. Smith, 'Some Extensions of Nyquist's Telegraph Transmission Theory', *Bell Syst. Tech. J.*, Vol. 44, pp. 1487-1510, Sept. 1965.

[12] R.D. Gitlin and S.B. Weinstein, 'Fractionally-Spaced Equalization: An Improved Digital Transversal Equalizer', *Bell Syst. Tech. J.*, Vol. 60, No. 2, pp. 275-296, Feb. 1981.

[13] L. Guidoux, 'Egalisateur Autoadaptif á Double Èchantillonage', *L'Onde Èlectronique*, Vol. 55, No. 1, pp. 9-13, 1975.

[14] M.L. Honig, P. Crespo and K. Steiglitz, 'Suppression of Near-End and Far-End Crosstalk by Linear Pre- and Post-Filtering' *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 3, pp. 614-629, Apr. 1992.

[15] G.E. Jacoby, 'Signal Equalization in Digital Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-4, No. 3, pp. 302-305, Sept. 1968.

[16] J. Jess and H.W. Schüssler, 'A Class of Pulse-Forming Networks', *IEEE Trans. Circuit Theory*, Vol. CT-12, pp. 393-400, Sept. 1965.

[17] A.R. Kaye and D.A. George, 'Transmission of Multiplexed PAM Signals Over Multiple Channel and Diversity Systems', *IEEE Trans. Commun. Technol.*, Vol. COM-18, pp. 520-526, Oct. 1970.

[18] M. Kavehrad, 'Timing-Jitter-Suppressed Partial-Response Signals', *Electronic Circuits and Systems*, Vol. 3, No. 5, pp. 201-207, Sept. 1979.

[19] K. Küpfmüller, 'Transient Phenomena in Wave Filters', *Elek. Nachr.-Techn.*, Vol. 1, p. 141, 1924.

[20] E.A. Lee and D.G. Messerschmitt, *Digital Communication*. Boston: Kluwer Academic Publishers, 1988.

[21] R.W. Lucky, J. Salz and E.J. Weldon, Jr., *Principles of Data Communication*. New York: McGraw-Hill, 1968.

[22] S.E. Nader and L.F. Lind, 'Optimal Data Transmission Filters', *IEEE Trans. Circuits Syst.*, Vol. CAS-26, pp. 36-45, Jan. 1979.

[23] H. Nyquist, 'Certain Factors Affecting Telegraph Speed', *Bell Syst. Tech. J.*, Vol. 3, No. 2, pp. 324-346, Apr. 1924.

[24] H. Nyquist, 'Certain Topics in Telegraph Transmission Theory', *Trans. AIEE (Commun. and Electronics)*, Vol. 47, pp. 617-644, Apr. 1928.

[25] P.K.D. Pai, A.A. Abidi and R.A. Gomez, 'A Simple Continuous-Time Equalizer for Use in Magnetic Storage Read Channels', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 289-299, Jan. 1992.

[26] J.G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1983 (2nd ed. 1989).

[27] S.U.H. Qureshi, 'Adaptive Equalization', *Proc. IEEE*, Vol. 73, No. 9, pp. 1349-1387, Sept. 1985.

[28] J. Salz, 'Digital Transmission over Cross-Coupled Linear Channels', *AT&T Technical Journal*, Vol. 64, No. 6, pp. 1147-1159, July-Aug. 1985.

[29] D. Shnidman, 'A Generalized Nyquist Criterion and an Optimum Linear Receiver for a Pulse Modulation System', *Bell Syst. Tech. J.*, Vol. 46, pp. 2163-2177, Nov. 1967.

[30] D.A. Spauling, 'Synthesis of Pulse-Shaping Networks in the Time Domain', *Bell Syst. Tech. J.*, Vol. 48, pp. 2425-2444, Sept. 1969.

[31] Y.P. Tsividis and J.O. Voorman (Eds.), *Integrated Continuous-Time Filters*. New York: IEEE Press, 1993.

[32] D.W. Tufts, 'Nyquist's Problem - The Joint Optimization of Transmitter and Receiver in Pulse Amplitude Modulation', *Proc. IEEE*, Vol. 53, pp. 248-259, March 1965.

[33] D.W. Tufts and T. Berger, 'Optimum Pulse Amplitude Modulation - Part I: Transmitter-Receiver Design and Bounds from Information Theory', *IEEE Trans. Inform. Theory*, Vol. IT-13, No. 2, pp. 196-208, Apr. 1967.

[34] D.W. Tufts and T. Berger, 'Optimum Pulse Amplitude Modulation - Part II: Inclusion of Timing Jitter', *IEEE Trans. Inform. Theory*, Vol. IT-13, No. 2, pp. 209-216, Apr. 1967.

[35] G. Ungerboeck, 'Fractional Tap-Spacing Equalizer and Consequences for Clock Recovery in Data Modems', *IEEE Trans. Commun.*, Vol. COM-24, No. 8, pp 856-864, Aug. 1976.

[36] P. Vandamme, 'On the Synthesis of Digital Transmit Filters', *IEEE Trans. Commun.*, Vol. COM-39, No. 4, pp. 485-487, Apr. 1991.

[37] R.W. Wood, 'New Detector for 1,k Codes Equalized to Class II Partial Response', *Proc. INTERMAG'89*, March 28-31, 1989, Washington, D.C., U.S.A.

6

PARTIAL-RESPONSE AND DECISION-FEEDBACK EQUALIZATION

6.1 INTRODUCTION

The perpetual push for higher bit rates and higher storage densities spurs a steady increment of the amplitude distortion of many types of transmission and storage channels. The linear equalizer (LE), discussed in Chapter 5, is ill-matched to this trend because its noise enhancement will increase and eventually become untolerable. In the present chapter we discuss two receiver types that run much less rapidly out of steam. These are the partial-response equalizer (PRE) and the decision-feedback equalizer (DFE). Both are rooted in old telegraph tricks and just as the LE they take instantaneous decisions with respect to the incoming data. The DFE is more powerful than the PRE and is, in fact, the optimum receiver with no detection delay. Performance improvements beyond the DFE rest in the capability to defer decisions for some time. This capability is a distinguishing feature of the Viterbi detector, discussed in Chapter 7.

PRE and DFE are usually treated separately. In the present chapter we treat them jointly in an attempt to highlight their intimate relationship. Basic to the PRE is the fact that a controlled amount of ISI is not suppressed by the equalizer but rather left for a simple detector to handle (Section 6.2). The nature of the controlled ISI is defined by a *partial response*. A proper match of this response to the channel permits noise enhancement to remain small even when amplitude distortion is severe. The DFE may be regarded as a generalization of the PRE (Section 6.3). In the DFE, the trailing portion of the ISI is not suppressed by a forward equalizer but rather canceled by a feedback filter that is excited by past decisions. This cancellation comes for free in terms of noise enhancement. A disadvantage is that erroneous decisions propagate via the feedback filter and can corrupt subsequent decisions. Fortunately, error propagation is typically only a minor problem and it can, in fact, be altogether avoided through a technique that is called Tomlinson/Harashima precoding (Section 6.4). A slightly modified DFE topology facilitates optimization of the equalizer with respect to the noise spectrum (Section 6.5). A similar technique as for the LE can be used to tailor the DFE to run-length-limited codes (Section 6.6). In terms of performance the DFE is typically much closer to the maximum-likelihood sequence detector than to the LE. Chapter 7 will shed more

light on this closeness by showing that the DFE can, in fact, be regarded as a "poor man's" Viterbi detector that keeps track of only a single candidate data sequence.

We will see that PRE and DFE, just as the LE, come in both zero-forcing and minimum mean-square error tastes. Performance differences between these tastes tend to be considerably smaller than for the LE, and as a result it becomes more difficult to reap SNR benefits from the modulation code. Again a transmit filter may be used to improve performance, be it that optimum filter characteristics differ from those for the LE.

Like Chapter 5, the present chapter deals mainly with basics of the approaches and to a much lesser extent with implementation aspects. Again we assume that the channel characteristics are fixed and known, so that PRE and DFE need not be adaptive. Our treatment of adaptivity is deferred until Chapter 8. The first part of the chapter again builds on the notions and concepts of Sections 3.2 and 3.10, and the reader may wish to review these sections before passing on.

6.2 PARTIAL-RESPONSE EQUALIZATION

In order to develop the partial-response equalizer we will first discuss *partial-response techniques*. These techniques originated as a coding or signaling method that aims at spectrum control or an enlargement of the signaling rate [27, 25, 22]. The intention is to transform the data sequence a_k, which typically has a flat power spectrum, into a sequence b_k whose spectral characteristics are better matched to those of the channel. For simplicity both sequences are desired to have the same data rate $1/T$.

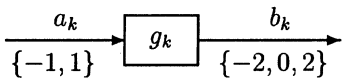


Figure 6.1: Partial-response transformation.

Basic to partial-response techniques is that b_k is obtained from a_k via a simple linear filter (Fig. 6.1). More specifically, the impulse response g_k of this filter is constrained to have only a few integer-valued coefficients, and is sometimes called a *partial response* when this condition is met. Provided that g_k is a partial response, the transformation of a_k into b_k involves no multiplications. Any nontrivial transformation causes the *partial-response sequence* b_k to have more amplitude levels than a_k. The simplest possible (and hence most important) case is that where a_k is binary with $a_k \in \{-1, 1\}$ and b_k is ternary with $b_k \in \{-2, 0, 2\}$. Here g_k has only two nonzero components of unit magnitude. Thus, apart from a delay without further relevance, g_k should have $\mathcal{D}$-transform $G(\mathcal{D}) = 1 \pm \mathcal{D}^K$ for some integer $K \geq 1$. Of greatest practical interest are the responses $1 + \mathcal{D}$, $1 - \mathcal{D}$ and $(1 + \mathcal{D})(1 - \mathcal{D}) = 1 - \mathcal{D}^2$ (Table 6.1).

The peculiar nomenclature of Table 6.1 has historical roots rather than a profound logical basis [27, 25, 19]. Besides the responses of Table 6.1, a plethora of additional responses has been described and studied (see [27, 25, 19, 22] for a small sample).

$G(\mathcal{D})$	Name	b_k	$\lvert G(e^{j2\pi\Omega})\rvert^2$
$1+\mathcal{D}$	Duobinary (Dicode)	$a_k + a_{k-1}$	$4\cos^2 \pi\Omega$
$1-\mathcal{D}$	Bipolar	$a_k - a_{k-1}$	$4\sin^2 \pi\Omega$
$1-\mathcal{D}^2$	Class IV (PRIV, Modified Duobinary)	$a_k - a_{k-2}$	$4\sin^2 2\pi\Omega$

Table 6.1: Characteristics of three partial responses.

Most of these yield a sequence b_k with more than 3 amplitude levels, and their practical significance is restricted. Two responses of some interest are $(1-\mathcal{D})(1+\mathcal{D})^2$ and $(1-\mathcal{D})(1+\mathcal{D})^3$, which have found application in digital magnetic recording systems with run-length-limited modulation code, and are sometimes called EPRIV and EEPRIV in recording jargon [38], [4]. Problem 6.1 addresses these two responses.

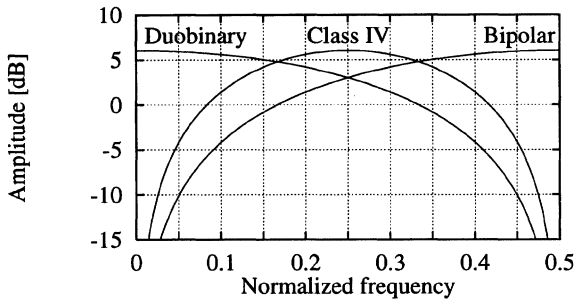


Figure 6.2: Amplitude-frequency characteristics of the partial responses of Table 6.1.

The scheme of Fig. 6.1 transforms the data spectrum $\mathcal{A}(e^{j2\pi\Omega})$ into a spectrum $\mathcal{A}(e^{j2\pi\Omega})|G(e^{j2\pi\Omega})|^2$, where $G(e^{j2\pi\Omega})$ is the discrete-time Fourier transform of g_k. Duobinary enhances low frequencies and suppresses those near the Nyquist frequency $\Omega = 0.5$ (Fig. 6.2). The opposite applies to Bipolar. Class IV has a bandpass characteristic and rejects the Nyquist frequency as well as DC. By applying one of these responses, the bulk of the data energy can be concentrated at low, medium or high frequencies (i.e. $\Omega = 0, 0.25$ and 0.5). This freedom can be exploited in the receiver for improved performance, as we will see.

In principle, the spectral zeros of $G(e^{j2\pi\Omega})$ can be undone via a linear equalizer, but this would excessively enhance any noise components added to b_k. Two alternative approaches for recovering a_k from b_k avoid this problem. In describing these we restrict attention to Duobinary $(1+\mathcal{D})$. The reasoning for the other responses is similar.

Precoding From Table 6.1, we see for Duobinary that $a_k = b_k - a_{k-1}$. This shows that a first-order recursive filter can in principle be used to transform b_k into an estimate $\hat{a}_k$ of a_k (Fig. 6.3.a). Unfortunately, if an erroneous estimate $\hat{a}_k$ is produced at transmission startup or at some other moment k, then all subsequent estimates will be wrong (in fact,

they will no longer be in the alphabet $\{-1, 1\}$, and this enables error monitoring and simple forms of error correction [22]). To avoid this *catastrophic error propagation*, resort can be taken to a *precoder* (Fig. 6.3.b).

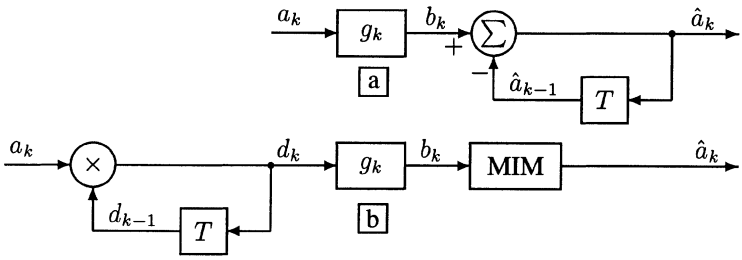


Figure 6.3: a. A one-pole recursive filter transforms b_k into an estimate $\hat{a}_k$ of a_k. b. To avoid error propagation, the filter is replaced by a precoder and a memoryless inverse mapping (MIM).

The precoder transforms a_k into a binary sequence $d_k = a_k d_{k-1}$ to which the partial-response transformation is applied. This yields a ternary sequence

$$b_k = (d * g)_k = d_k + d_{k-1} = a_k d_{k-1} + d_{k-1} = (a_k + 1)d_{k-1}. \tag{6.1}$$

Since d_{k-1} cannot be zero, b_k is zero if and only if $(a_k+1) = 0$, i.e. if and only if $a_k = -1$. Thus we can form an estimate $\hat{a}_k$ of a_k by means of the memoryless inverse mapping

$$\hat{a}_k = \begin{cases} -1 & \text{if } b_k = 0 \text{ and} \\ 1 & \text{else.} \end{cases} \tag{6.2}$$

This decoding rule does not rely on past data estimates and thereby avoids error propagation altogether. In practice the sequences a_k and d_k are in the alphabet $\{0, 1\}$ rather than $\{-1, 1\}$, and the multiplication in Fig. 6.3.b becomes a modulo-2 addition. Thus the cost of avoiding error propagation amounts to a binary delay element and an exclusive-or operation. Problem 4.15 reveals that the precoder does not affect the spectral characteristics of an uncorrelated data sequence, i.e. that d_k is uncorrelated whenever a_k is uncorrelated. For correlated data, precoding need not be spectrally neutral.

It is instructive to think of the precoder as a first-order recursive filter with a pole at the Nyquist frequency that cancels the zero of the partial response. The filter uses a modulo-2 addition instead of a normal addition and as a result the cascade of filter and partial response, while memoryless, has a ternary rather than binary output. The MIM serves to repair this 'deficiency'.

Exercise 6.1 What precoder and MIM are suited for the Bipolar $(1-\mathcal{D})$ partial response? How about Class IV $(1-\mathcal{D}^2)$?

An added advantage of the precoding/MIM scheme is that it renders the system polarity-insensitive. The precoder output d_k reflects polarity *changes* of the data a_k and

the output of the MIM will not change if all digits b_k are inverted in polarity as a result, for example, of erroneous wiring somewhere within the transmission chain. Feedback detection, to be described next, goes without this advantage.

Feedback detection Catastrophic error propagation can be avoided without precoder by forcing the output of the recursive filter of Fig. 6.3 to be binary (Fig. 6.4.a). An erroneous estimate $\hat{a}_{k-1} = -a_{k-1}$ leads to a digit

$$\tilde{a}_k = b_k - \hat{a}_{k-1} = a_k + a_{k-1} - \hat{a}_{k-1} = a_k + 2a_{k-1} \tag{6.3}$$

whose polarity is obviously determined by a_{k-1}. Thus the decision $\hat{a}_k$ that is taken by the slicer in Fig. 6.4.a will be correct if and only if a_k happens to equal a_{k-1}. If data is uncorrelated then this will be so with a probability of 50%, and error propagation will cease. If not then it will stop in the next symbol interval with a probability of 50%, and so on. The average number of errors in a burst amounts to $1 + 0.5 + (0.5)^2 + \ldots = 2$. Thus error propagation is not a serious probem.

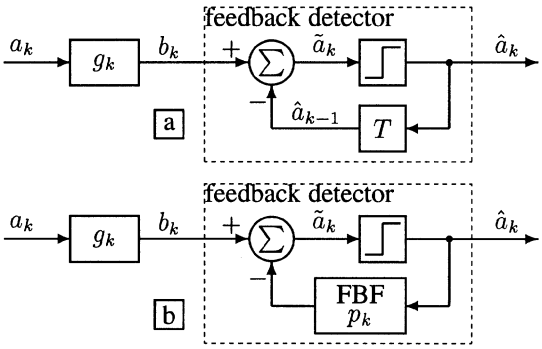


Figure 6.4: a. A slicer is added to the recursive filter of Fig. 6.3.a in order to force decisions $\hat{a}_k$ to be in the alphabet $\{-1, 1\}$. Erroneous decisions $\hat{a}_k$ may propagate for some time but are no longer catastrophic. b. This idea can be generalized to arbitrary causal partial responses g_k. The 'tail' $p_k = g_k - \delta_k$ of g_k is canceled by a feedback filter with impulse response p_k.

The *feedback detector* of Fig. 6.4.a is easily generalized to arbitrary partial responses g_k. For purposes of normalization g_k is assumed to be causal and monic (i.e. $g_k = 0$ for $k < 0$ and $g_0 = 1$). The nontrivial samples $g_1, g_2, \ldots$ together form the 'tail' of g_k. For notational convenience we collect this tail in an impulse response p_k, with $p_k = 0$ for $k \leq 0$ and $p_k = g_k$ for $k \geq 1$. Hence $g_k = \delta_k + p_k$, where the Kronecker delta function δ_k represents the component $g_0 = 1$. Thus

$$b_k = (a * g)_k = (a * (\delta + p))_k = a_k + (a * p)_k. \tag{6.4}$$

The term $(a * p)_k$ depends exclusively on past digits $a_{k-1}, a_{k-2}, \ldots$ that can be replaced by decisions $\hat{a}_{k-1}, \hat{a}_{k-2} \ldots$. We can, therefore, form an estimate $\tilde{a}_k$ of the current digit

a_k according to $\tilde{a}_k = b_k - (\hat{a} * p)_k$ as in Fig. 6.4.b. As before, a slicer quantizes $\tilde{a}_k$ into binary decisions $\hat{a}_k$ so as to avoid catastrophic error propagation. The average length of bursts of errors nevertheless tends to increase rapidly as the duration of g_k grows. Even so, error propagation is not normally a serious problem [22, 13, 36].[1] In essence, the feedback detector avoids noise enhancement by exploiting past decisions. In Section 6.3 we will see that viewpoint is central to decision-feedback equalization.

To simplify the exposition we have so far considered only binary data a_k. Partial-response techniques are, however, equally applicable to multilevel data a_k. Extension of precoding to L-level data requires that we replace the multiplier (or, equivalently, the modulo-two adder) in Fig. 6.3 by a modulo-L adder [25],[22]. The MIM must then take decisions modulo L. All that is needed for the feedback detectors of Fig. 6.4 is a replacement of the slicers by multilevel threshold detectors with $L-1$ decision thresholds that are located halfway between the L nominal data levels.

6.2.1 Zero-forcing partial-response equalization

In Chapter 5 we have observed that linear equalizers inevitably incur serious noise enhancement in undoing severe amplitude distortion. By comparison, we now know that binary data may be recovered without any equalization in the presence of the specific (and indeed severe) distortion that is introduced by partial-response transformations. To apply this possibility in a practical system we must equalize the transfer function of the channel into a partial-response characteristic. This is often possible with less equalization effort and thereby less noise enhancement than for the *full-response* equalization techniques considered so far.

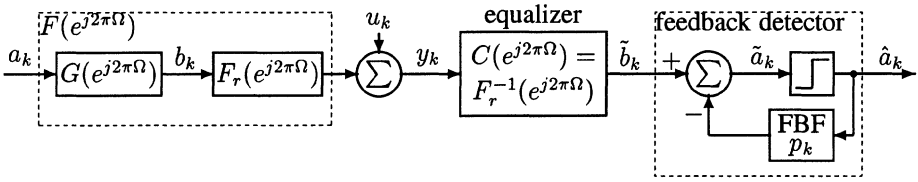


Figure 6.5: The channel transfer function $F(e^{j2\pi\Omega})$ is thought of as comprising two parts: a partial response with transfer function $G(e^{j2\pi\Omega})$ that captures the bulk of the amplitude distortion, and a residual response with transfer function $F_r(e^{j2\pi\Omega}) = F(e^{j2\pi\Omega})/G(e^{j2\pi\Omega})$ which accounts for the remaining distortion. The equalizer equalizes only $F_r(e^{j2\pi\Omega})$ and in doing so incurs little noise enhancement. Its output $\tilde{b}_k$ is applied to the feedback detector of Fig. 6.4.

In Fig. 6.5, a discrete-time channel with transfer function $F(e^{j2\pi\Omega})$ transforms a_k into a sequence $y_k = (a * f)_k + u_k$, where u_k is noise with power spectral density $\mathcal{U}(e^{j2\pi\Omega})$. Just as in Section 3.10, y_k could represent the sampled output of a matched filter or whitened matched filter. Let us assume for the sake of the argument that noise

[1]Error propagation is virtually absent in various extensions of the DFE, such as the Viterbi detector with sequence feedback of Section 7.7 and the dual DFE of Section 7.9.1. These extensions also achieve a better predetection signal-to-noise ratio.

is approximately white and that the channel has a low-pass nature, so that $|F(e^{j2\pi\Omega})|$ somewhat resembles the transfer magnitude $|G(e^{j2\pi\Omega})|$ of the Duobinary response $1+\mathcal{D}$ (cf. Fig. 6.2). Then we might think of $F(e^{j2\pi\Omega})$ as comprising two parts: a transfer function $G(e^{j2\pi\Omega})$ that captures most of the amplitude distortion of the channel, and a function $F_r(e^{j2\pi\Omega}) = F(e^{j2\pi\Omega})/G(e^{j2\pi\Omega})$ that accounts for the remaining distortion. The latter distortion has only a small amplitude component and can thus be undone without much noise enhancement by a linear equalizer with transfer function

$$C(e^{j2\pi\Omega}) = \frac{1}{F_r(e^{j2\pi\Omega})} = \frac{G(e^{j2\pi\Omega})}{F(e^{j2\pi\Omega})}. \tag{6.5}$$

It should be stressed that the subdivision of Fig. 6.5 is purely conceptual: the partial-response sequence b_k is, in general, nowhere to be found in the actual channel. The equalizer output $\tilde{b}_k$ is a noisy version of b_k, and is applied to the feedback detector of Fig. 6.4 to obtain decision variables $\tilde{a}_k$ and decisions $\hat{a}_k$. (The precoder and memoryless inverse mapping of Fig. 6.3 are, of course, also applicable and yield essentially the same performance.) We can interpret g_k as a *target response* that defines the desired characteristics of the equalized system. The partial-response equalizer (PRE) is designed to transform the actual response of the system into this target response.

To appraise the performance of the PRE we neglect error propagation in the feedback detector, i.e. we assume that all past decisions that affect $\tilde{a}_k$ are correct. Then the only difference between $\tilde{a}_k$ and a_k is a filtered noise component $(u * c)_k$ with variance

$$\varepsilon_{\text{ZF PRE}} = \int_{-0.5}^{0.5} \mathcal{U}(e^{j2\pi\Omega})|C(e^{j2\pi\Omega})|^2 d\Omega = \int_{-0.5}^{0.5} \frac{\mathcal{U}(e^{j2\pi\Omega})|G(e^{j2\pi\Omega})|^2}{|F(e^{j2\pi\Omega})|^2} d\Omega. \tag{6.6}$$

Since $|G(e^{j2\pi\Omega})|$ was selected to be small wherever $|F(e^{j2\pi\Omega})|$ is small, the integrand of (6.6) never becomes disproportionately large, and $\varepsilon_{\text{ZF PRE}}$ will be small. This compares favorably with the approach of Chapter 5, that will henceforth be designated *full-response equalization* for reasons of symmetry. Here $G(e^{j2\pi\Omega}) = 1$ for all Ω, and the integrand of (6.6) can become large at frequencies where $|F(e^{j2\pi\Omega})|$ is small. Intuitively, one would argue that the smallest possible noise enhancement accrues if $G(e^{j2\pi\Omega})$ is selected in order for the integrand to be independent of frequency, i.e. for noise at the output of the PRE to be white. This is, in general, not possible if $G(e^{j2\pi\Omega})$ is restricted to the responses of Table 6.1. The generalized feedback detector of Fig. 6.4.b, by comparison, allows a wide variety of causal responses to be used, and here $|G(e^{j2\pi\Omega})|$ can be chosen at liberty. Exploitation of this freedom leads to the DFE (see Section 6.3).

The PRE is superior to the LE, but is still suboptimum even when noise $(u * c)_k$ at the equalizer output is white. The reason is that the detector effectively discards all data power that stems from past data symbols. Specifically, the feedback detector of Fig. 6.4 cancels all trailing ISI, which accounts for half of all data power for the responses of Table 6.1. As a result, the detector lags 3 dB behind the matched-filter bound if $(u*c)_k$ is white. The precoder/MIM combination of Fig. 6.3.b incurs the same loss because it is equivalent to the feedback detector except for error propagation. Simplified versions of the Viterbi detector may be used to recover this loss (see Chapter 7).

Continuous-time ZF PRE According to (6.5), we can transform any full-response equalizer into a partial-response equalizer by multiplying its transfer function by a factor $G(e^{j2\pi\Omega})$. This also applies to continuous-time equalizers. Thus, for example, the continuous-time ZF PRE with minimum noise enhancement has transfer function

$$W(\Omega) = \frac{H^*(\Omega)}{\mathcal{N}(\Omega)} \frac{G(e^{j2\pi\Omega})}{\mathrm{FSN}(e^{j2\pi\Omega})} \tag{6.7}$$

and achieves a mean-square error

$$\hat{\varepsilon}_{\mathrm{ZF\,PRE}} = \int_{-0.5}^{0.5} \frac{|G(e^{j2\pi\Omega})|^2}{\mathrm{FSN}(e^{j2\pi\Omega})} d\Omega, \tag{6.8}$$

where $H(\Omega)$, $\mathcal{N}(\Omega)$ and $\mathrm{FSN}(e^{j2\pi\Omega})$ represent the transfer function, the noise PSD and the folded signal-to-noise ratio of the hybrid channel, respectively. This extends expressions (5.11) and (5.13). The time-domain effect of the multiplication by $G(e^{j2\pi\Omega})$ is that the original system response $q(t)$ is transformed into a new response $q_{\mathrm{PRE}}(t) = \sum_k g_k q(t - kT)$. Not surprisingly, if $q(t)$ is a Nyquist-1 function, i.e. if $q(kT) = \delta_k$, then $q_{\mathrm{PRE}}(t)$ has sampled version $q_{\mathrm{PRE}}(kT) = g_k$.

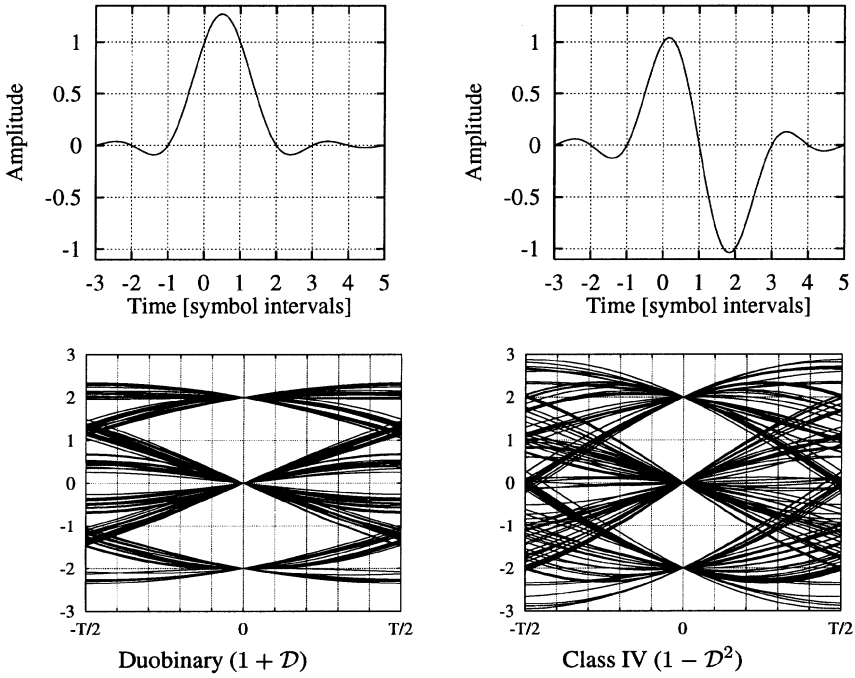
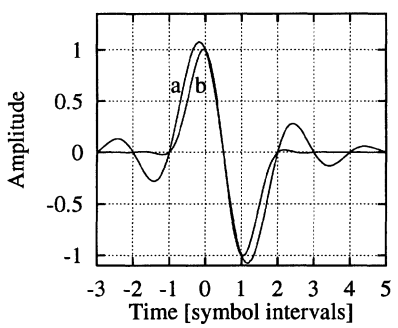


Figure 6.6: Minimum-bandwidth system responses $q_{\mathrm{PRE}}(t)$ for the Duobinary $(1 + \mathcal{D})$ and Class IV $(1 - \mathcal{D}^2)$ partial responses. Corresponding noiseless eye patterns are also depicted.

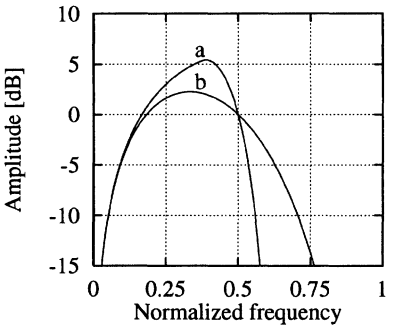
Example 6.1 Duobinary and Class IV are usually applied in systems where excess bandwidth is at a premium. Here $q(t)$ is typically a minimum-bandwidth Nyquist-1 function, whence

$$Q_{\text{PRE}}(\Omega) = \begin{cases} TG(e^{j2\pi\Omega}) & \text{for } |\Omega| < 0.5 \text{ and} \\ 0 & \text{for } |\Omega| > 0.5. \end{cases} \tag{6.9}$$

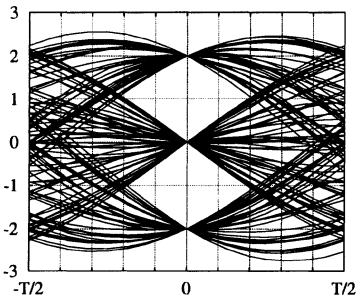
The transfer magnitude $|Q_{\text{PRE}}(\Omega)|$, therefore, is shaped just as $|G(e^{j2\pi\Omega})|$ in Fig. 6.2. Spectral content declines gradually towards the Nyquist frequency $1/(2T)$ and is zero from there on. System responses $q_{\text{PRE}}(t)$ and noiseless eye patterns are depicted in Fig. 6.6. Both responses clearly reflect the underlying partial responses g_k, and decay rapidly even though $q(t)$ decays only slowly (see the $\beta = 0$ response in Fig. 5.3). The eye patterns have a distinctly positive width even though excess bandwidth is absent. This is quite unlike full-response signaling, where eye width vanishes along with excess bandwidth (see Problem 5.1). That undesirable effect also occurs with Bipolar $(1 - \mathcal{D})$ signaling and this explains in part why Bipolar system responses $q_{\text{PRE}}(t)$ invariably have significant excess bandwidth. In the example responses of Fig. 6.7, this excess bandwidth stems from a raised-cosine function $q(t)$. We refer to [22] for a more extensive review of the characteristics of continuous-time partial-response signals.



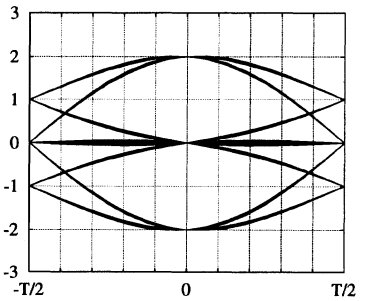
System responses $q_{\text{PRE}}(t)$ for $\beta = 0.25$ (a.) and $\beta = 1$ (b.)



Amplitude characteristics $|Q_{\text{PRE}}(\Omega)|$ for $\beta = 0.25$ (a.) and $\beta = 1$ (b.)



Noiseless eye pattern ($\beta = 0.25$)



Noiseless eye pattern ($\beta = 1$)

Figure 6.7: Characteristics of a Bipolar $(1 - \mathcal{D})$ system. Excess bandwidth is defined by a roll-off parameter β.

Example 6.2 In Section 2.3.5 we have seen that digital magnetic recording systems with a differentiating playback head inevitably reject DC. For this type of channel, zero-forcing full-response equalization leads to catastrophic noise enhancement. This problem may be mitigated by means of MMSE techniques at the cost of residual ISI (see Example 5.7). Alternatively, a partial response that rejects DC may be used. For example, for the Class IV response $1 - \mathcal{D}^2$ and a channel transfer function $H(\Omega)$ according to (2.5) we need an equalizer transfer function

$$W(\Omega) = \frac{2j \sin 2\pi\Omega Q(\Omega)}{H(\Omega)} = \frac{4}{\sqrt{T}} \cos \pi\Omega e^{\pi D|\Omega|} Q(\Omega), \tag{6.10}$$

where $Q(\Omega)$ is some Nyquist-1 characteristic. This function is well-behaved at all frequencies, including DC, and noise enhancement remains restricted. The same applies for the Bipolar response $1 - \mathcal{D}$. The performance of both schemes will be appraised in Example 6.9.

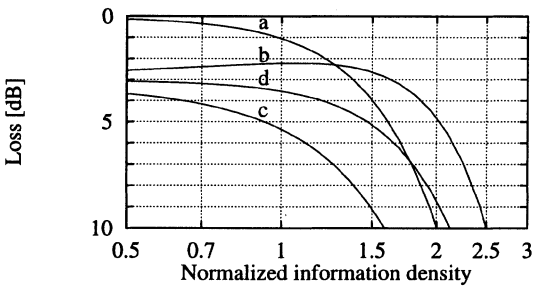


Figure 6.8: Loss $\mathcal{L}$ with respect to the matched-filter bound versus normalized information density S for various zero-forcing equalizers operating on the idealized optical recording channel of Example 5.3: a. full-response ($G(\mathcal{D}) = 1$); b. duobinary ($G(\mathcal{D}) = 1 + \mathcal{D}$); c. bipolar ($G(\mathcal{D}) = 1 - \mathcal{D}$); d. class IV ($G(\mathcal{D}) = 1 - \mathcal{D}^2$).

Example 6.3 The idealized optical recording channel of Examples 5.3 and 5.5 has a low-pass nature, just as the Duobinary $(1 + \mathcal{D})$ partial response. This suggests that Duobinary is well suited to this channel. Fig. 6.8 confirms this expectation. The figure compares losses $\mathcal{L}$ with respect to the matched-filter bound (MFB) for the Duobinary PRE with those of other PREs and of the full-response LE. All equalizers are dimensioned to have minimum noise enhancement as in (5.11) and (6.7). The Duobinary equalizer is superior to all other equalizers for densities $S > 1.2$, yet still lags at least 2 dB behind the MFB. In Chapter 7 we will see that a 2-state Viterbi detector may be used to lower this minimum loss to a fraction of a dB. For $S > 1$ the channel has negligible excess bandwidth and this carries over to the equalized system responses. In this density range the Duobinary and Class IV responses are, therefore, essentially equal to the ones of Fig. 6.6. Bipolar is impractical for $S > 1$ because eye width vanishes along with excess bandwidth, and also because of the large noise enhancement that it brings about. By way of illustration Fig. 6.9 depicts the Duobinary eye pattern for a density $S = 1.5$ and SNR $= 20$ dB. This pattern may be compared to the full-response patterns of Figs. 5.4. The improved noise immunity is apparent.

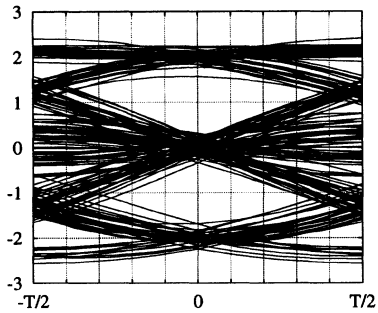


Figure 6.9: Eye pattern at the output of the Duobinary $(1+\mathcal{D})$ equalizer with minimum noise-enhancement when operating at the idealized optical recording channel of Example 5.3 at a normalized information density $S = 1.5$ and SNR = 20 dB.

6.2.2 MMSE Partial-Response Equalization

The equalizer of Fig. 6.5 can be dimensioned in an MMSE rather than ZF fashion. Its output sequence $\tilde{b}_k$ is then desired to equal the partial response sequence $b_k = (a*g)_k$ as well as possible in the mean-square sense. Thus the error sequence $e_k = \tilde{b}_k - b_k$ should have minimum power, i.e. its power spectral density $\mathcal{E}(e^{j2\pi\Omega})$ should be minimized for every frequency Ω. Now $e_k = (a * f * c)_k + (u * c)_k - (a * g)_k = (a * (f * c - g))_k + (u * c)_k$. Therefore

$$\mathcal{E} = \mathcal{A}|FC - G|^2 + \mathcal{U}|C|^2 = |G|^2[\mathcal{A}|F\frac{C}{G} - 1|^2 + \mathcal{U}|\frac{C}{G}|^2] \tag{6.11}$$

where the dependence on Ω has been suppressed for notational convenience. It follows that only the ratio of C and G counts in minimizing $\mathcal{E}$ with respect to C. Hence any full-response MMSE LE can be transformed into its partial-response counterpart by multiplying its transfer function by a factor $G(e^{j2\pi\Omega})$. This is the same transformation as for the ZF PRE, and again it also applies to continuous-time equalizers. The factor $G(e^{j2\pi\Omega})$ is typically smallest in frequency regions where ZF and MMSE full-response equalizers differ most from each other. As a result, differences between the ZF and MMSE PREs tend to be small.

Example 6.4 The discrete-time MMSE PRE has transfer function

$$C = \frac{GF^*}{\mathcal{U}} \frac{1}{\frac{|F|^2}{\mathcal{U}} + \frac{1}{\mathcal{A}}} \tag{6.12}$$

and the corresponding error e_k has PSD

$$\mathcal{E} = \frac{|G|^2}{\frac{|F|^2}{\mathcal{U}} + \frac{1}{\mathcal{A}}}. \tag{6.13}$$

Intuitively, one would expect that $|G|$ should ideally be chosen for e_k to be white, i.e. for $\mathcal{E}$ to be independent of frequency. This requires that $|G|^2 \propto \frac{|F|^2}{\mathcal{U}} + \frac{1}{\mathcal{A}}$. In Section 6.3 we will find that the discrete-time MMSE DFE achieves this ideal.

Example 6.5 For a continuous-time channel with transfer function $H(\Omega)$ and additive noise with power spectral density $\mathcal{N}(\Omega)$, the MMSE full-response LE of infinite length was derived in Section 5.4 (eq. (5.26)). Multiplication of its transfer function by $G(e^{j2\pi\Omega})$ gives the transfer function of the continuous-time MMSE PRE:

$$W(\Omega)\hat{C}(e^{j2\pi\Omega}) = \frac{H^*(\Omega)}{\mathcal{N}(\Omega)} \frac{G(e^{j2\pi\Omega})}{\mathrm{FSN}(e^{j2\pi\Omega}) + \mathcal{A}^{-1}(e^{j2\pi\Omega})}. \tag{6.14}$$

The MMSE amounts to

$$\hat{\varepsilon}_{\text{MMSE PRE}} = \int_{-0.5}^{0.5} \frac{|G|^2}{\mathrm{FSN} + \mathcal{A}^{-1}} d\Omega \tag{6.15}$$

and the corresponding predetection signal-to-noise ratio $\hat{\rho}_{\text{MMSE PRE}} = 1/\hat{\varepsilon}_{\text{MMSE PRE}}$ may be recognized as the harmonic average (see Section 1.5) of $[\mathrm{FSN} + \mathcal{A}^{-1}]/|G|^2$, i.e.

$$\hat{\rho}_{\text{MMSE PRE}} = \overline{\left[\frac{\mathrm{FSN} + \mathcal{A}^{-1}}{|G|^2}\right]}^{H}. \tag{6.16}$$

For optimum performance, $|G|^2$ should be chosen to resemble $\mathrm{FSN} + 1/\mathcal{A}$ as well as possible. We will see in Section 6.3 that the continuous-time MMSE DFE exactly reaches this goal.

Bias The MMSE PRE, just as the MMSE LE, generally exhibits bias, i.e. the amplitude A of the desired data component at the output of the PRE tends to drop somewhat below unity. Equation (6.16) does not account for the effect of bias on performance and is in this sense slightly optimistic, especially at poor SNRs (see also Appendix 5A).

6.2.3 Transmit filter for the PRE

The considerations of Section 5.7 carry over to the PRE with only minor modifications. In particular, the structure of the optimum transmit filter (TF) does not change, i.e. $C_c(\Omega) = F^*(\Omega)W^*(\Omega)C_t(e^{j2\pi\Omega})$ where C_c, F, W, and C_t are defined as in Section 5.7. The equalized system characteristic $Q(e^{j2\pi\Omega})$ again satisfies (5.35). The transfer function of the optimum TF, just as that of the PRE, is obtained by multiplying the full-response function by a factor $G(e^{j2\pi\Omega})$. Thus, for example, for the ZF PRE we have $C_t(e^{j2\pi\Omega}) = G(e^{j2\pi\Omega})/X(e^{j2\pi\Omega})$ where $X(e^{j2\pi\Omega})$ is defined as in Section 5.7. The resulting equalized system characteristic is $Q(e^{j2\pi\Omega}) = C_t(e^{j2\pi\Omega})X(e^{j2\pi\Omega}) = G(e^{j2\pi\Omega})$ (compare (5.35)).

Joint optimization of TF and PRE leads to essentially the same conclusions as in Section 5.7. In particular, transfer functions of TF and PRE are again nonzero across a frequency range $\mathcal{R}$ of total width $1/T$, and filtering effort should be split equally among TF and PRE when data and noise are both uncorrelated. For the ZF criterion, their amplitude characteristics are then both proportional to $\sqrt{|G(e^{j2\pi\Omega})/F(\Omega)|}$ throughout $\mathcal{R}$. The effects of a nonequal split are studied in [33] for a similar system as in Example 5.9.

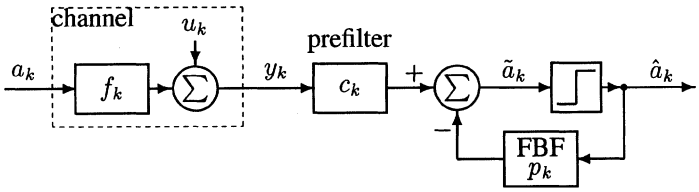


Figure 6.10: Discrete-time decison feedback equalizer (DFE). A prefilter with transfer function $C(e^{j2\pi\Omega})$ transforms the channel transfer function $F(e^{j2\pi\Omega})$ into a characteristic $G(e^{j2\pi\Omega})$ with minimum phase. A feedback filter with impulse response p_k cancels trailing ISI.

6.3 DECISION-FEEDBACK EQUALIZATION

Discrete-time ZF DFE The decision-feedback equalizer (DFE), shown in Fig. 6.10, is just the combination of a prefilter and the feedback detector of Fig. 6.4.b. This combination was analyzed in Section 6.2.1 for the situation that the prefilter transforms the symbol response of the channel into a prescribed target response g_k. This response is constrained to be causal so that *pre-cursive ISI*, i.e. interference due to symbols that have not yet been detected, is absent. The constraint $g_0 = 1$ ensures that $\tilde{a}_k$ coincides with a_k in the absence of noise and *post-cursive* or *trailing* ISI, i.e. ISI due to symbols that have already been detected. To be able to cancel the latter ISI, the feedback filter (FBF) has a causal impulse response p_k that should match g_k for all $k \geq 1$. This match should be accurate so as to minimize residual ISI, and is usually accomplished by making the FBF adaptive (see Chapter 8). The optimum zero-forcing target response g_k may be identified as follows:

1. In Section 6.2.1 it was argued that noise $e_k = \tilde{a}_k - a_k = (u * c)_k$ at the output of the prefilter is ideally white or, equivalently, that $|G(e^{j2\pi\Omega})|^2$ is ideally of the form $\mu|F(e^{j2\pi\Omega})|^2/\mathcal{U}(e^{j2\pi\Omega})$, where μ is some constant of proportionality. Thus only μ and the phase characteristics of $G(e^{j2\pi\Omega})$ remain to be determined.

2. Of all causal impulse responses g_k with prescribed amplitude characteristics $|G(e^{j2\pi\Omega})|$, the one with minimum phase has its energy optimally concentrated near the time-origin and has thereby the largest possible amplitude g_0 (see Section 1.6). Equivalently, for a constrained value $g_0 = 1$, a minimum-phase response will yield the smallest possible constant of proportionality μ and thereby the smallest possible noise enhancement.

Thus g_k should be a scaled version of the *minimum-phase root* of $|F|^2/\mathcal{U}$. This root is the unique impulse response γ_k with minimum phase and amplitude characteristics $|\Gamma|^2 = |F|^2/\mathcal{U}$, and may be identified via the recursive procedure of Section 1.6. In terms of γ_k we may express the optimum system response as $g_k = \gamma_k/\gamma_0$. Clearly $g_0 = 1$, as desired. The prefilter has transfer magnitude

$$|C| = \frac{|G|}{|F|} = \frac{1}{\gamma_0}\frac{|\Gamma|}{|F|} = \frac{1}{\gamma_0}\frac{1}{\sqrt{\mathcal{U}}} \tag{6.17}$$

and noise e_k at its output, therefore, has PSD $\mathcal{U}|C|^2 = \gamma_0^{-2}$, i.e. e_k is indeed white. We infer that the ZF DFE achieves a minimum mean-square error

$$\varepsilon_{\text{ZF DFE}} = \gamma_0^{-2} = \exp\{-\int_{-0.5}^{0.5} \ln \frac{|F|^2}{\mathcal{U}} d\Omega\}. \tag{6.18}$$

The second equality is a standard property of minimum-phase functions (see Section 1.6) and shows that the performance of the DFE may be expressed directly in terms of the channel parameters.

Example 6.6 Assume that u_k is white. Then $|C| = 1$ for all Ω, i.e. the prefilter becomes a pure phase equalizer, which neither colors nor enhances noise. The phase characteristics of the prefilter are matched to those of the channel in such a way that the equalized system response g_k has minimum phase. The underlying aim is to concentrate as much energy as possible in the cursor g_0, which is the only sample of g_k that is exploited by the feedback detector.

Finite-length ZF DFE The above results do not account for the fact that long responses g_k tend to induce more error propagation than short ones. This propagation is typically not serious, but can be restricted by restricting the length of the feedback filter. Appendix 6B derives the corresponding ZF DFE. In many instances, FBF lengths as small as 3 to 5 already permit performances close to (6.18).

Continuous-time ZF DFE To translate the above results to the continuous-time situation we regard y_k as the sampled output of a matched filter with transfer function $H^*(\Omega)/\mathcal{N}(\Omega)$. Then $F(e^{j2\pi\Omega}) = |F(e^{j2\pi\Omega})|^2/\mathcal{U}(e^{j2\pi\Omega}) = \text{FSN}(e^{j2\pi\Omega})$ (see Section 3.10.1), whence $|\Gamma(e^{j2\pi\Omega})|^2 = \text{FSN}(e^{j2\pi\Omega})$. Thus the ZF DFE with minimum noise enhancement has a continuous-time prefilter with transfer function

$$\begin{aligned} W(\Omega)C(e^{j2\pi\Omega}) &= \frac{H^*(\Omega)}{\mathcal{N}(\Omega)}\frac{G(e^{j2\pi\Omega})}{F(e^{j2\pi\Omega})} = \frac{H^*(\Omega)}{\mathcal{N}(\Omega)}\frac{\Gamma(e^{j2\pi\Omega})}{\gamma_0 \text{FSN}(e^{j2\pi\Omega})} \\ &= \frac{1}{\gamma_0}\frac{H^*(\Omega)}{\mathcal{N}(\Omega)}\frac{1}{\Gamma^*(e^{j2\pi\Omega})} \end{aligned} \tag{6.19}$$

and achieves a minimum mean-square error

$$\hat{\varepsilon}_{\text{ZF DFE}} = \exp\{-\int_{-0.5}^{0.5} \ln \text{FSN}(e^{j2\pi\Omega}) d\Omega\}. \tag{6.20}$$

The transfer function of (6.19) describes a matched filter that is followed by a T-spaced anti-causal discrete-time filter. Sampled noise e_k at the output of this cascade is white and the sampled system response g_k has minimum phase. These two properties together characterize the *canonical whitened matched filter* (see Section 3.10.2). In Chapter 7 we will see that this WMF is also the optimum prefilter for several Viterbi detectors.

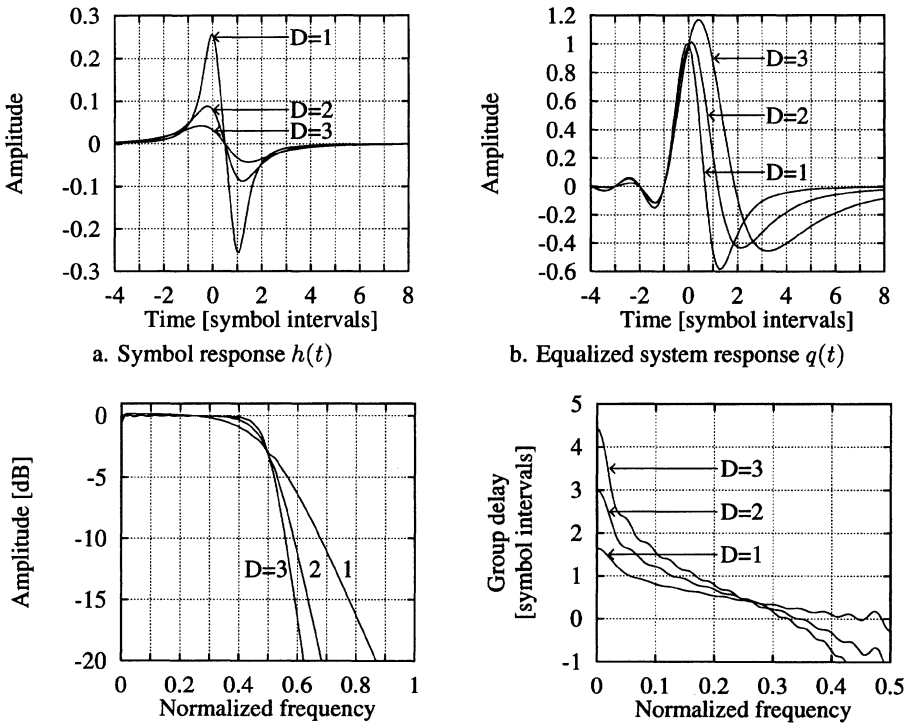
The predetection signal-to-noise ratio $\hat{\rho}_{\text{ZF DFE}} = 1/\hat{\varepsilon}_{\text{ZF DFE}}$ corresponding to (6.20) may be recognized as the *geometric average* of $\text{FSN}(e^{j2\pi\Omega})$ that we have denoted $\overline{\text{FSN}}^G$

in Section 1.5. This ratio may be compared the optimum ratio $\rho_{\text{MFB}} = \text{SNR} = \overline{\text{FSN}}^A$ for the matched-filter bound (see Section 3.9) as well as to the ratio $\hat{\rho}_{\text{ZF LE}} = \overline{\text{FSN}}^H$ for the optimum ZF LE (see Section 5.3):

$$\rho_{\text{MFB}} = \overline{\text{FSN}}^A \geq \hat{\rho}_{\text{ZF DFE}} = \overline{\text{FSN}}^G \geq \hat{\rho}_{\text{ZF LE}} = \overline{\text{FSN}}^H. \tag{6.21}$$

The inequalities are standard properties of averages (see Section 1.5), and become equalities if and only if $\text{FSN}(e^{j2\pi\Omega})$ is independent of Ω. Then the ZF DFE degenerates into a ZF LE because all FBF coefficients p_k become zero. In all other cases the ZF DFE is superior to the ZF LE, and typically approaches the MFB within a few dB.

Example 6.7 Assume that noise is white and that the channel has negligible excess bandwidth. Then $|\Gamma(e^{j2\pi\Omega})|^2 = \text{FSN}(e^{j2\pi\Omega}) \propto |H(\Omega)|^2$ for $|\Omega| \leq 0.5$, i.e. $|W(\Omega)C(e^{j2\pi\Omega})|$ is constant for $|\Omega| \leq 0.5$. Frequencies beyond $|\Omega| = 0.5$ are all rejected by the matched filter. It follows that the amplitude characteristics of the prefilter are those of an ideal anti-aliasing filter. The phase characteristics are tailored for the sampled system response q_k to have minimum phase.



a. Symbol response $h(t)$

b. Equalized system response $q(t)$

c. Amplitude characteristics of WMF

d. Group-delay characteristics of WMF

Figure 6.11: Time- and frequency-domain characteristics of a Lorentzian recording system with a canonical whitened matched filter (WMF).

Example 6.8 Fig. 6.11.a depicts symbol responses $h(t)$ of the Lorentzian recording channel of Section 2.3.5 for three different normalized information densities $D = t_{50}/T$. Corresponding equalized system responses $q(t)$ at the output of a canonical WMF are depicted in Fig. 6.11.b for the situation that noise of the channel is white. Sampled system responses $g_k = q(kT)$ are all causal and monic, and they all have minimum phase. Their effective duration increases rapidly with D. As a result, a longer FBF will be needed, and error propagation will eventually become severe.

As conjectured in Example 6.7, amplitude characteristics of the WMF indeed approach those of an ideal anti-aliasing filter, especially for large D (Fig. 6.11.c). Amplitude characteristics of $q(t)$ are, therefore, essentially identical to those of $h(t)$ as long as $H(\Omega)$ has little out-of-band content, which is true for densities of practical interest (compare Fig. 2.25). Group-delay characteristics of the WMF are a strong function of D (Fig. 6.11.d). As D increases, the extent of $h(t)$ becomes larger, and more group delay is needed for the WMF to transform $h(t)$ into a 'causal' pulse $q(t)$, particularly at low frequencies.

Example 6.9 We return to the Lorentzian magnetic recording channel of Section 2.3.5 (see also Appendix 3A). With expression (3A.7) and (6.20) we can compute the loss that the ZF DFE incurs with respect to the matched-filter bound. The result of this numerical computation is shown in Fig. 6.12 as a function of the normalized information density D. Also shown are the losses of the maximum-likelihood sequence detector (MLSD; see Example 3.11), the full-response MMSE LE [eq. (5.28), SNR=20 dB], and the Bipolar $(1 - \mathcal{D})$ and Class IV $(1 - \mathcal{D}^2)$ ZF PRE's (see Example 6.2). For densities up to about 1, $1 - \mathcal{D}$ equalization approaches the MFB within 3 dB, just as $1 - \mathcal{D}^2$ at more practical densities of around 2. The DFE improves upon all linear equalizers, and comes within approximately 1 dB of the MFB for densities around $D = 1.5$. We recall from Example 3.11 that the MLSD achieves the MFB for densities D up to $\sqrt{6} \simeq 2.45$, beyond which multiple bit-errors prevail.

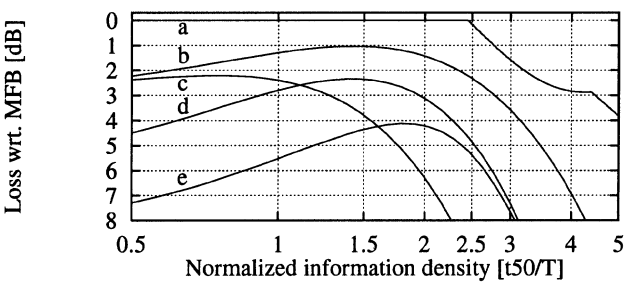


Figure 6.12: Loss in predetection signal-to-noise ratio relative to the matched-filter bound versus normalized information density $D = t_{50}/T$ for various receivers operating on a Lorentzian recording channel: a. MLSD; b. ZF DFE; c. Bipolar $(1 - \mathcal{D})$ ZF PRE; d. Class IV $(1 - \mathcal{D}^2)$ ZF PRE; e. MMSE full-response LE.

Discrete-time MMSE DFE We proceed to derive the DFE that minimizes the mean-square error $\varepsilon = E[(\tilde{a}_k - a_k)^2]$ between $\tilde{a}_k$ and a_k in Fig. 6.10. The error sequence $e_k = \tilde{a}_k - a_k$ is now allowed to contain residual ISI in addition to noise. In Example 6.4

we have argued that $|G(e^{j2\pi\Omega})|^2$ is ideally proportional to

$$|\Gamma(e^{j2\pi\Omega})|^2 \triangleq \frac{|F(e^{j2\pi\Omega})|^2}{\mathcal{U}(e^{j2\pi\Omega})} + \frac{1}{\mathcal{A}(e^{j2\pi\Omega})}. \tag{6.22}$$

Just as for the ZF DFE, g_k should have minimum phase and g_0 should be unity. Thus $g_k = \gamma_k/\gamma_0$ where γ_k is the minimum-phase root of $|\Gamma(e^{j2\pi\Omega})|^2$. As before, the feedback filter cancels the trailing part of g_k, i.e. $p_k = g_k - \delta_k$ or, equivalently,

$$G(e^{j2\pi\Omega}) = 1 + P(e^{j2\pi\Omega}) \text{ for all } \Omega. \tag{6.23}$$

With (6.12) one finds that the optimum prefilter has transfer function

$$C = G\frac{F^*}{\mathcal{U}}\frac{1}{|\Gamma|^2} = \frac{F^*}{\mathcal{U}}\frac{1}{\gamma_0\Gamma^*}. \tag{6.24}$$

Not surprisingly, the resulting error sequence e_k is white and has variance

$$\varepsilon_{\text{MMSE DFE}} = \frac{1}{\gamma_0^2} = \exp\{-\int_{-0.5}^{0.5} \ln[\frac{|F|^2}{\mathcal{U}} + \frac{1}{\mathcal{A}}]d\Omega\}. \tag{6.25}$$

Again, the second equality is a standard property of minimum-phase functions (see Section 1.6) and shows that the performance of the DFE may be expressed directly in terms of the channel parameters.

This brief derivation highlights the close link of the MMSE DFE with the MMSE PRE and its similarity to the ZF DFE. A more rigorous and self-contained derivation may be found in Appendix 6A. Problems 6.10 and 6.11 are concerned with various additional properties of the MMSE DFE.

Continuous-time MMSE DFE To translate the above results to the continuous-time situation, we again interpret y_k as the sampled output of a matched filter. Then $|F| = \mathcal{U} = |F|^2/\mathcal{U} = \text{FSN}$ and $|\Gamma|^2 = \text{FSN} + \mathcal{A}^{-1}$. Thus the MMSE DFE with minimum noise enhancement has a continuous-time prefilter with transfer function

$$W(\Omega)C(e^{j2\pi\Omega}) = \frac{H^*(\Omega)}{\mathcal{N}(\Omega)}\frac{1}{\Gamma^*(e^{j2\pi\Omega})}. \tag{6.26}$$

This filter may be thought of as the cascade of a matched filter and a T-spaced discrete-time filter. It is no longer a whitened matched filter because the discrete-time filter performs error-whitening rather than noise-whitening. The MMSE amounts to

$$\hat{\varepsilon}_{\text{MMSE DFE}} = \exp\{-\int_{-0.5}^{0.5} \ln[\text{FSN} + \mathcal{A}^{-1}]d\Omega\}. \tag{6.27}$$

The corresponding predetection signal-to-noise ratio $\hat{\rho}_{\text{MMSE DFE}} = 1/\hat{\varepsilon}_{\text{MMSE DFE}}$ may be recognized as the geometric average of $\text{FSN} + \mathcal{A}^{-1}$, and is compared with other ratios in Table 6.2.

	LE	DFE	MFB
ZF	$\overline{\text{FSN}}^H$	$\overline{\text{FSN}}^G$	$\overline{\text{FSN}}^A$
MMSE	$\overline{\text{FSN} + \mathcal{A}^{-1}}^H$	$\overline{\text{FSN} + \mathcal{A}^{-1}}^G$	

Table 6.2: Optimum predetection signal-to-noise ratios of various ZF and MMSE equalizers. Also included is the ratio that corresponds to the matched-filter bound (MFB).

All performances are spectral averages of the type defined in Section 1.5. They depend primarily on $\text{FSN}(e^{j2\pi\Omega})$ and for MMSE receivers also on $\mathcal{A}^{-1}(e^{j2\pi\Omega})$. MMSE performances do not account for the effect of bias and decrease marginally if bias is reckoned with (see Appendix 5A). Performances are non-decreasing from top to bottom and from left to right. The gap between LE, DFE and MFB increases with the degree of non-flatness of FSN (or of FSN + $\mathcal{A}^{-1}$ in the MMSE case). In Section 1.5 we have argued that the geometric average of a function is typically closer (on a dB scale) to the arithmetic than to the harmonic average. Translated to Table 6.2 this means that the DFE is bound to be closer to the MFB than to the LE in terms of performance. This is evidenced, for example, by Fig. 6.12.

MMSE DFE of finite length The foregoing development has not accounted for error propagation via the feedback filter. While this propagation is not normally a serious problem, it can be restricted by restricting the length of the feedback filter to some prescribed value N_{FB}. Appendix 6B derives the MMSE DFE for this situation. The resulting MMSE is evidently smaller than that of (6.27), but in many instances FBF lengths as small as 3 to 5 will already permit MMSEs that are within a fraction of a dB from (6.27).

FBF implementation In the event that data a_k is binary, the input of the FBF will also be binary, and its output can be determined with only sign inversions and additions. These operations also suffice to make the filter adaptive (see Section 8.8). An even simpler approach -at least conceptually- involves the use of a table look-up filter (see Sections 8.2.5 and 8.8.3). In such a filter, the output is looked up in a table rather than computed. Any input-ouput relation can be stored so that nonlinear ISI of the trailing type can, in principle, be handled as well. A disadvantage of adaptive look-up tables is that they tend to contain many entries, all of which need to be updated. This leads to slow convergence. The hybrid and split filter structures of Problems 8.44 and 8.45 provide some relief in this respect. Table look-up FBFs have long been used in digital subscriber lines [20], and have more recently found their way into digital magnetic recording [16], [3], [28].

Transmit filter for the DFE The optimum transmit filter for the DFE is identical to the one for the PRE as in Section 6.2.3, be it that the class of allowable functions $G(e^{j2\pi\Omega})$ has grown in size. Just as for the LE and PRE, joint optimization of TF and DFE leads to a TF transfer function that is nonzero across a frequency range $\mathcal{R}$ of total

width $1/T$. For every $\Omega \in [-0.5, 0.5]$, $\mathcal{R}$ again contains exactly one frequency $\Omega + n$ for some $n \in \mathbb{Z}$, depending on which value of n maximizes the signal-to-noise ratio $|F(\Omega + n)|^2/\mathcal{N}(\Omega + n)$. Thus $\mathcal{R}$ needs not be continuous. Within $\mathcal{R}$, the optimum amplitude characteristics of TF and prefilter are identical except for a scaling factor, provided that data and noise are both uncorrelated. For the ZF criterion both characteristics are, in fact, flat throughout $\mathcal{R}$, i.e. within $\mathcal{R}$ the optimum TF and DFE prefilter are both phase equalizers [35]. In particular when $\mathcal{R} = [-0.5, 0.5]$, signaling with sinc-pulses as in Section 3.3 will do. The MMSE solution is similar in nature but rather more complicated in analytical terms [37, 5].

Losses with respect to channel capacity In the absence of past decision errors, the DFE detector input $\tilde{a}_k$ is just a noisy version of a_k as in Fig. 6.13.

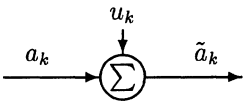


Figure 6.13: Equivalent discrete-time model of transmit filter, channel and DFE up until the detector input. All pre-cursive ISI is suppressed by the DFE prefilter, and all post-cursive ISI is eliminated via the FBF.

The optimum ZF prefilter for the DFE is a whitened matched filter, which causes noise u_k to be white. If u_k is Gaussian then the channel of Fig. 6.13 is just an additive Gaussian white noise (AGWN) channel. The capacity C of this channel was analyzed in Section 3.4 and may be compared to that of the underlying continuous-time channel. Capacity losses are smallest when TF and DFE are optimized jointly and then rarely exceed 10 to 15% for a wide class of channels [2]. Corresponding SNR penalties are in the order of 1 dB or less. Further results in this context may be found in [9].

Example 6.10 In Section 3.4 we have seen that uncoded transmission with an L-level data alphabet across the discrete-time channel of Fig. 6.13 requires some 6 to 7 dB more SNR for a symbol-error rate of 10^{-5} than an optimum coded scheme that conveys $\log_2(L)$ bits per symbol interval (compare Fig. 3.11; the coded scheme would, of course, require more than L data levels). This corresponds to an SNR penalty of around 8 dB with respect to the underlying continuous-time channel. An early analysis of this penalty was given by Price [35].

6.4 TOMLINSON/HARASHIMA PRECODING

Error propagation of the DFE can be circumvented by a precoding technique that was developed independently by Tomlinson [39] and Harashima and Miyakawa [21]. The technique involves a relocation of the feedback detector to the transmitting end of the system. As such it is basically an extension of precoding for partial-response systems as in Section 6.2. Our development starts from the feedback detector of Fig. 6.4.b. This detector is meant for a causal system response g_k with $g_0 = 1$. Post-cursive ISI is canceled via a feedback filter whose impulse response p_k is the 'tail' of g_k, i.e. $p_k = g_k - \delta_k$.

Error propagation is due to the fact that the FBF is excited by past decisions, and may be avoided by omitting the slicer in Fig. 6.4.b. The feedback detector then becomes a linear filter with transfer function $1/[1 + P(e^{j2\pi\Omega})] = 1/G(e^{j2\pi\Omega})$. This is, in effect, a linear equalizer so that we have solved a modest problem (error propagation) at the cost of introducing a larger one (noise enhancement). The latter problem can, however, be avoided by relocating the filter to the transmitting end of the system (Fig. 6.14.a). Filter and channel are both linear so that the system remains distortionless, i.e. a slicer suffices to produce decisions $\hat{a}_k$. Noise u_k appears directly at the detector input and is no longer enhanced. Major disadvantage of the configuration of Fig. 6.14.a is that the amplitude $|s|$ and power P_s of the transmitted sequence s_k can be much larger than that of the data sequence a_k. In the usual event that $|s|$ and/or P_s are constrained, an additional attenuation factor must be included within the transmitter relative to the situation without transmit-side equalizer. As a result, the predetection signal-to-noise ratio degrades by a similar amount as incurred by removing the quantizer in Fig. 6.4.b (see also Section 6.2.3).

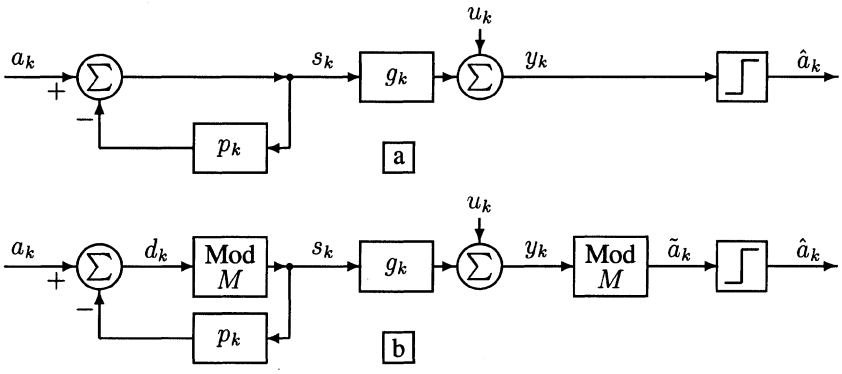


Figure 6.14: a. Discrete-time data-transmission system with causal channel and recursive transmit-side linear equalizer. b. Transmit power may be restricted via a modulo operation. The resulting recursive structure is referred to as a *Tomlinson-Harashima precoder*. A second modulo-operation is needed in the receiver to undo the effect of the one in the precoder.

Exercise 6.2 Consider a system response g_k with $\mathcal{D}$-transform $G(\mathcal{D}) = 1 + \alpha\mathcal{D}$ for some $\alpha \in [-1, 1]$. Determine $|s|$ and P_s for the situation that data a_k is uncorrelated and binary with $a_k \in \{-1, 1\}$. Show that $|s|$ and P_s grow without bound as $|\alpha| \to 1$.

Tomlinson/Harashima precoding lowers transmit power via a modulo operation that restricts s_k to a predefined interval $(-M, M]$ (Fig. 6.14.b). A direct implication is that $|s|$ and P_s cannot exceed M and M^2, respectively. The modulo operator works as follows. If the input d_k is outside the interval $(-M, M]$ then an integer times $2M$ is added to or subtracted from d_k in order to arrive at an output s_k that is within $(-M, M]$. Thus we may write $s_k = d_k + 2Mm_k$ where m_k is an integer that is chosen to 'fit' s_k within $(-M, M]$. Now $d_k = a_k - (s * p)_k$ and upon recalling that $g_k = \delta_k + p_k$ we find after

some elementary manipulations that

$$(s * g)_k = a_k + 2Mm_k. \tag{6.28}$$

The left-hand side may be recognized as the data component of the channel output, whence $y_k = a_k + 2Mm_k + u_k$. The linear system of Fig. 6.14.a, by comparison, has $y_k = a_k + u_k$. The net effect of the modulo operator, then, is that an additional data component $2Mm_k$ emerges in the channel output. This component increases the power and amplitude of y_k relative the situation of Fig. 6.14.a (see Problem 6.12 for an assessment). The increment is usually small enough to be perfectly acceptable. A second effect of the modulo operator is that the added component $2Mm_k$ may obscure detection of a_k. This is, however, not true if M is chosen sufficiently large relative to the range A of allowable digits a_k. For any $m_k \in \mathbb{Z}$, A gets mapped into a received range $B(m_k) \triangleq A + 2Mm_k$ and one easily verifies that the ranges $B(m)|_{m\in\mathbb{Z}}$ are all disjunct if A has width $W_A \leq 2M$. A case in point is the PAM alphabet $\{\pm 1, \pm 3, \ldots, \pm(M-1)\}$, for which $A = [-(M-1), (M-1)]$. More generally, if $A = (-M, M]$ then detection boils down to reduction of y_k modulo $2M$ and subsequent selection of the nearest valid data level. This 'decision modulo $2M$' is the counterpart of the memoryless inverse mapping (MIM) in Section 6.2. The resulting predetection signal-to-noise ratio is the same as for a system with ZF DFE, but error rates can be somewhat better because error propagation is altogether absent.

By construction, the transmitted symbol s_k lies in the interval $(-M, M]$, and is typically uniformly distributed across this interval so that $P_s \simeq M^2/3$ [18]. By comparison, if a_k is uncorrelated and in the PAM alphabet $\{\pm 1, \pm 3, \ldots, \pm(M-1)\}$ then a_k has power $P_a = (M^2-1)/3$. The power increment P_s/P_a due to the precoder amounts to 3 dB for binary signaling ($M = 2$) and decreases rapidly with M.

It can be noted that the transmit-side equalizer in Fig. 6.14.a becomes unstable if g_k is not a minimum-phase function. The Tomlinson/Harashima precoder of Fig. 6.14.b remains applicable in this case, be it that received data power can become relatively large (see Problem 6.12).

A disadvantage of Tomlinson/Harashima precoding is that the precoded data sequence s_k is, in general, non-integer-valued. This disqualifies the technique for systems with a binary write signal, such as most digital recording systems. Also, the non-integer nature of s_k and d_k renders a digital implementation rather complex, be it that simplifications are possible for responses g_k whose $\mathcal{D}$-transform is a rational function of $\mathcal{D}$ [31]. Problem 6.13 addresses another exception. A further disadvantage is that the technique requires knowledge about the channel characteristics. This knowledge is not usually available at the transmitting end of the system and must then be established via a separate feedback path. Since the DFE goes without these disadvantages and achieves comparable performance, it is usually preferred to Tomlinson/Harashima precoding. Even so, Tomlinson/Harashima precoding is currently enjoying renewed interest because it can be combined with certain trellis codes to the effect that the receiver only needs to decode the trellis code [18]. This greatly simplifies the receiver relative to the situation that it would also have to deal with ISI. A recent alternative to Tomlin-

son/Harashima precoding, called *flexible precoding*, is more broadly applicable in this context yet commands a modest performance penalty [26], [15].

6.5 IN-LOOP NOISE WHITENING

The superiority of the DFE over the PRE accrues primarily from the fact that the DFE prefilter performs noise whitening, while noise at the output of a PRE is, in general, colored. We will now see that whitening is also possible by adding a simple recursive filter to the feedback detector of Fig. 6.4.b. The extended detector improves the performance of the PRE towards that of the DFE at a modest hardware cost. We develop the approach for a Class IV partial-response system with first-order whitening and leave it up to the reader to generalize the development to other target responses and higher-order whitening (Problems 6.15 and 6.16). For simplicity we take the PRE to be of the ZF type and assume that the output of the equalizer is sampled at the proper instants under the control of a timing recovery subsystem. The cascade of continuous-time channel, PRE and sampler can then be modeled as a discrete-time channel (Fig. 6.15). This channel is Class IV, i.e. its impulse response g_k has $\mathcal{D}$-transform $g(\mathcal{D}) = 1 - \mathcal{D}^2$. Additive noise u_k is, in general, colored, especially when the folded signal-to-noise ratio of the channel is ill-matched to a Class IV characteristic (see Section 6.2). Let us assume for the sake of the argument that u_k has a high-frequency nature. Then noise whitening requires a low-pass filter (LPF). A particularly simple LPF is the first-order recursive filter whose impulse response w_k has $\mathcal{D}$-transform $w(\mathcal{D}) = 1/(1 - \alpha\mathcal{D})$. For $\alpha \in [-1, 1]$ this filter is stable and for $\alpha \in (0, 1]$ it has a low-pass nature. (The development at hand extends trivially to the high-pass case $\alpha \in [-1, 0)$.)

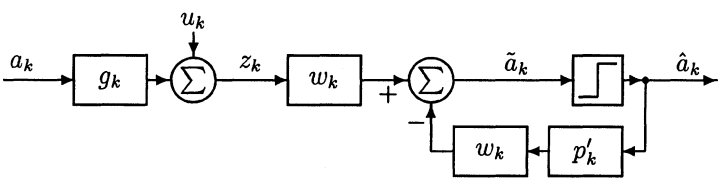


Figure 6.15: Discrete-time system with colored noise u_k, low-pass noise-whitening filter and feedback detector with two-stage feedback filter.

The LPF transforms $g(\mathcal{D})$ into a new system response with $\mathcal{D}$-transform $q(\mathcal{D}) = g(\mathcal{D})w(\mathcal{D})$. By virtue of its causality this response yields no pre-cursive ISI, but it does cause trailing ISI that may extend far into the past. Compensation of that ISI is possible via a feedback detector as in Fig. 6.4.b, and requires an FBF whose impulse response has $\mathcal{D}$-transform $p(\mathcal{D}) = q(\mathcal{D}) - q_0\mathcal{D}^0 = q(\mathcal{D}) - 1$. This may be rewritten

$$p(\mathcal{D}) = [g(\mathcal{D}) - w^{-1}(\mathcal{D})]w(\mathcal{D}) = [g(\mathcal{D}) - 1 + \alpha\mathcal{D}]w(\mathcal{D}) = [\alpha\mathcal{D} - \mathcal{D}^2]w(\mathcal{D}). \quad (6.29)$$

We may apparently regard the FBF as a cascade of: a) a two-tap transversal filter whose impulse response p'_k has $\mathcal{D}$-transform $p'(\mathcal{D}) = g(\mathcal{D}) - 1 + \alpha\mathcal{D} = \alpha\mathcal{D} - \mathcal{D}^2$, and b) the

LPF (Fig. 6.15). The LPF is common to both inputs of the rightmost summator and can be shifted across it. The resulting detector is shown in Fig. 6.16. It operates on a Class IV partial-response sequence z_k and performs noise whitening via an in-loop LPF. The FBF has two taps, of which only p'_1 is coupled to the LPF bandwidth. Both filters are simple, and so is the entire detector. Accurate compensation of all trailing ISI requires adjustment of only one parameter p'_1. The detector may be recast in a largely analog form. In particular, the summator and LPF can be analog, while the FBF will normally be largely digital. A performance assessment and further details may be found in [7].

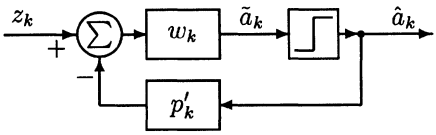


Figure 6.16: Feedback detector with in-loop noise-whitening filter.

The approach is readily extended to arbitrary target responses $g(\mathcal{D})$. It is also applicable to full-response equalization ($g(\mathcal{D}) = 1$). Here $p'(\mathcal{D}) = g(\mathcal{D}) - w^{-1}(\mathcal{D}) = \alpha\mathcal{D}$, i.e. the FBF in Fig. 6.16 has only a single tap $p'_1 = \alpha$. Performance may surpass that of the full-response LE by a considerable margin when noise is highly colored, such as in data transmission across FM radio links.

6.6 DECISION-FEEDBACK EQUALIZATION FOR RUN-LENGTH-LIMITED CODES

The MMSE DFE accounts for the modulation code by de-emphasizing frequencies where the code has little spectral content (see (6.22) and (6.24)). As a result, noise is not enhanced at these frequencies, but some residual ISI emerges instead. We will now see that it is sometimes possible to exploit the properties of the modulation code in such a fashion as to obtain a reduction in noise enhancement at no cost to residual ISI. More specifically, we will tailor the DFE along the lines of [23] and [8] to run-length-limited codes with minimum-run-length parameter $d = 1$. Examples of such codes are MFM, M^2 and $(1, 7)$ (see Section 4.8). Our treatment of the $d = 1$ DFE builds upon the development of the $d = 1$ LE in Section 5.5. There we have considered equalized and sampled system responses q_k of the form

$$q_k = \begin{cases} 1 & \text{for } k = 0, \\ \alpha_1 & \text{for } |k| = 1, \\ \alpha_2 & \text{for } |k| = 2, \text{ and} \\ 0 & \text{otherwise,} \end{cases} \tag{6.30}$$

and we have shown that only constructive ISI (i.e. ISI that does not degrade the eye opening relative to a system without ISI) will arise provided that

$$\alpha_1 \geq -\alpha_2 \geq 0. \tag{6.31}$$

By comparison, for a system without ISI we would have $\alpha_1 = \alpha_2 = 0$, and condition (6.31) is clearly much more permissive. Even so, DC-free system responses are all excluded and such responses are unavoidable for systems that reject DC. The addition of a feedback filter provides relief (Fig. 6.17). In effect, the presence of an FBF enables the 'tail' of the equalized system response to be moulded in any desired fashion, and this freedom can be used to arrive at a system response without DC while minimizing noise enhancement.

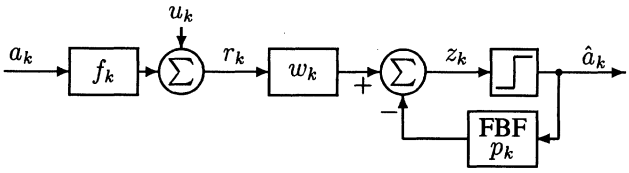


Figure 6.17: Discrete-time data-transmission system with noisy dispersive channel and decision-feedback equalizer.

The FBF is strictly causal and has a finite memory length $N_{FB} \geq N_p$ so that $p_k = 0$ for $k \notin \{1, \ldots, N_{FB}\}$. The prefilter has impulse response w_k and transforms the channel impulse response f_k into a system response $q_k = (f * w)_k$. If f_k is DC-free then so is q_k and condition (6.31) cannot be met. In the absence of decision errors, the net effect of the FBF is to change q_k into an effective system response $q_k^e \triangleq q_k - p_k$. This response can satisfy (6.31) even though q_k does not by a proper choice of the FBF coefficients p_k. Without loss of generality we take $q_0^e = q_0 = 1$. Since p_k can be chosen at liberty for $k = 1, \ldots, N_{FB}$ only, q_k must vanish for $k < -N_p$ and $k > N_{FB}$ where N_p denotes the number of nonzero precursors that is allowed (i.e. $N_p = 2$ if α_1 and α_2 both differ from zero, $N_p = 1$ if only α_1 differs from zero, and $N_p = 0$ if $\alpha_1 = \alpha_2 = 0$). We are free to choose q_k within the span $k \in S \triangleq \{-N_p, \ldots, -1, 1, \ldots, N_{FB}\}$ in such a way as to minimize noise enhancement, be it that q_{-2} and q_{-1} must satisfy (6.31). Once q_k is known, q_k^e and p_k are fully determined according to

$$q_k^e = \begin{cases} q_k & \text{for } k \leq 0 \text{ and} \\ q_{-k} & \text{for } k > 0, \end{cases} \tag{6.32}$$

and

$$p_k = \begin{cases} q_k - q_k^e & \text{for } k \in \{1, \ldots, N_{FB}\} \text{ and} \\ 0 & \text{else.} \end{cases} \tag{6.33}$$

Thus the FBF compensates trailing ISI in such a manner that an even effective system response q_k^e that satisfies (6.31) emerges. The ZF DFE corresponds to the limiting case where all trailing ISI is canceled and q_k^e is a Kronecker Delta function ($q_k^e = \delta_k$, $N_p =$

0). Appendix 6C.1 derives the function q_k that minimizes noise enhancement for given values of $N_p \geq 0$ and $N_{FB} \geq \max(1, N_p)$. The coefficients $q_{-N_p}, \ldots, q_{-1}$ are not *a priori* constrained and if the resulting solution does not satisfy (6.31) then there are two options:

1. N_p can be reduced by 1 and the resulting solution may then obey (6.31). For $N_p = 0$ we arrive at the ZF DFE, which is guaranteed to satisfy (6.31).

2. If $N_p = 2$, then a constrained solution with $q_{-2} = -q_{-1}$ may be obtained as in Appendix 6C.2. If this solution has $q_{-1} \geq 0$ then it will satisfy (6.31).

Based on these options we can distinguish a total of four classes of system responses (Fig. 6.18).

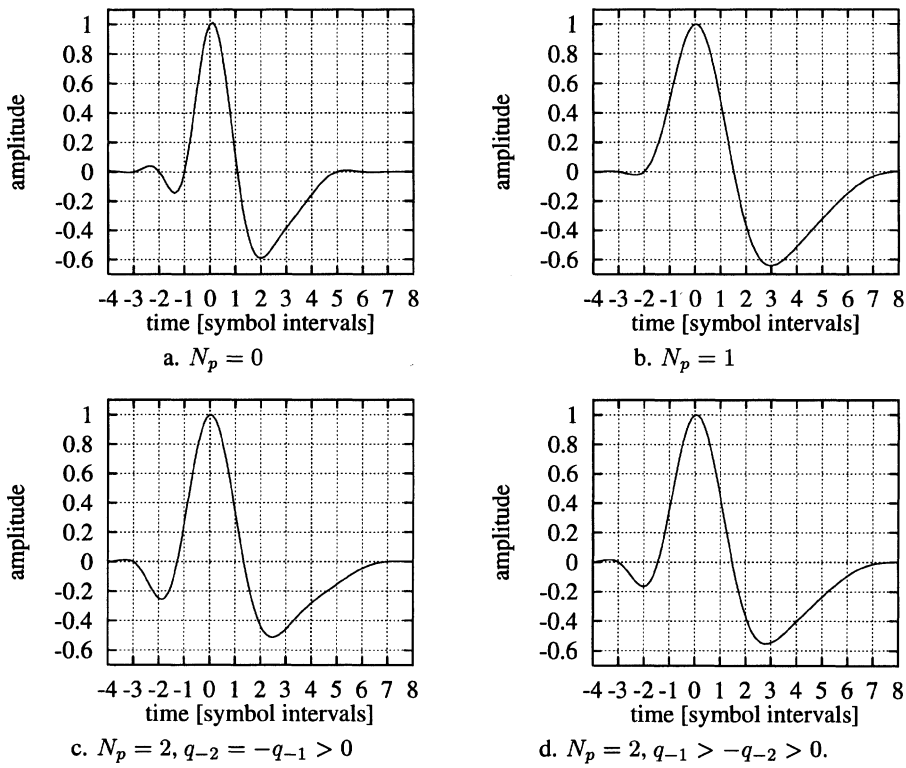


Figure 6.18: Equalized system responses that comply with condition (6.31) may be subdivided into four classes.

Depicted are continuous-time system responses $q(t)$ as they would emerge in a practical system. Responses q_k are sampled versions of $q(t)$. For $N_p = 0$ (Fig. 6.18.a) we have the classical zero-forcing DFE, which eliminates all pre-cursive ISI. The $N_p = 1$

response (Fig. 6.18.b) has a single nonzero precursor. The responses of Figs. 6.18.c and 6.18.d both have two nonzero precursors. They differ in that the first one is constrained to meet the first part of condition (6.31) with equality ($q_{-2} = -q_{-1}$) whereas the latter one is unconstrained, i.e. it meets both parts of (6.31) with strict inequality. Appendices 6C.1 and 6C.2 together cover all four classes. Traversal of this small list of possibilities allows one to identify the optimum response q_k, i.e. the one satisfying (6.31) with minimum noise enhancement.

Example 6.11 We revisit the Lorentzian channel of Appendix 3A and Examples 5.7 and 6.9. The modulation code has rate $R = 2/3$. This rate is typical for $d = 1$ codes (see Section 4.8). Information densities D are normalized with respect to the user data rate $1/T_s$, i.e. $D = t_{50}/T_s = 3t_{50}/(2T)$ where T is the symbol spacing on the channel. The noise component of the continuous-time replay signal is assumed additive and white. To facilitate a comparison of the results with those of Examples 5.7 and 6.9, we take the matched-filter bound for uncoded transmission as a performance baseline, i.e. we compute the predetection signal-to-noise ratios of $d = 1$ receivers relative to $\mathrm{SNR}|_{R=1}$. We have seen in Example 4.12 that any reduction of R below 1 will degrade SNR according to $\mathrm{SNR}_R = \mathrm{SNR}_1(1 + D^2)/[1 + (\frac{D}{R})^2]$ where the subscripts denote the code rate at hand. In Fig. 6.19, this function is compared with the optimum predetection signal-to-noise ratios $1/\hat{\sigma}^2$ of the $d = 1$ DFE for $N_p = 0$, 1 and 2. All DFEs have an FBF of length $N_{FB} = 6$. This value is chosen to restrict complexity and error propagation. Larger values of N_{FB} yield only marginally better predetection signal-to-noise ratios.

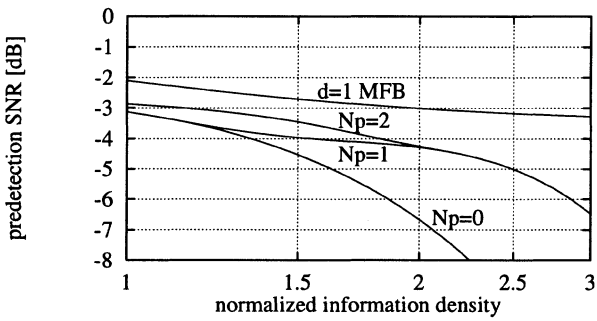


Figure 6.19: Predetection signal-to-noise ratio versus normalized information density D of $d = 1$ DFEs for a $d = 1$ code with rate $R = 2/3$. Also depicted is the matched-filter bound (MFB) for $d = 1$.

For $N_p = 2$, the constrained solution of Appendix 6C.2 (Fig. 6.18.c) is optimum for densities D below approximately 1.7. Beyond that point the unconstrained solution for $N_p = 2$ of Appendix 6C.1 takes over (Fig. 6.18.d), followed for $D > 2.2$ by the $N_p = 1$ solution (Fig. 6.18.b). The ZF DFE ($N_p = 0$, Fig. 6.18.a) seems unattractive at high densities. Throughout the density range of Fig. 6.19, DFE performances are up to 1.3 dB better than those that an MMSE DFE for the code at hand would achieve [8]. Hence minimization of the mean-square error yields a slightly suboptimum equalizer dimensioning.

6.7 FURTHER READING

Partial-response techniques find their roots in an old telegraph trick that is known as 'doubling the dotting speed' [10], [6, ch. 7]. This trick is essentially a form of Duobinary $(1+\mathcal{D})$ signaling. The modern version of this approach, with precoding included in order to circumvent error propagation, is due to Lender [27] and was soon extended to partial responses other than $1+\mathcal{D}$ [19], [25]. A survey of partial-response techniques may be found in [22].

Classical treatments of partial-response techniques all assume bit-by-bit detectors of the type considered in this chapter. Such detectors are, in principle, suboptimum. Close-to-optimum detectors of the maximum-likelihood type were first developed in the early seventies [24], [17], [14] and will be discussed in Chapter 7.

The historical origins of decision-feedback equalization also lie in the field of telegraphy [29], [10] (see also [6], [35] and [5]). A rudimentary version of the technique was meant to cancel post-cursive intersymbol interference that arises in telegraph transmission across signal cables. Considerable time elapsed before it was recognized that the technique, when combined with a suitable prefilter, would allow a significant performance improvement relative to full-response linear equalization [1]. Important steps in the analysis and understanding of the DFE were made by Monsen [32], Salz [37], Price [35], Messerschmitt [30] and others. A tutorial treatment with historical notes may be found in [5]. Extensions to more general channel models than the one considered here are developed in [34], [12], and references therein. Specifically, cyclostationary disturbances may be captured in the form of an extended channel model with multiple inputs and a single output. Just as for the LE, exploitation of cyclostationarity can be attractive if this extended channel has significant excess bandwidth but the advantage quickly erodes when this is not true (see also Appendix 2C).

APPENDIX 6A

Derivation of the discrete-time MMSE DFE

By following the procedure outlined in Section 5.4 we write the mean-square error $\varepsilon = E[e_k^2]$ in frequency-domain notation as

$$\varepsilon = \int_{-0.5}^{0.5} \{\mathcal{A}|FC - 1 - P|^2 + \mathcal{U}|C|^2\} d\Omega \tag{6A.1}$$

where we have suppressed all dependencies on Ω for notational convenience. In this expression it is assumed that error propagation is absent, i.e. that $\hat{a}_k = a_k$ for all k. In time-domain notation (6A.1) reads

$$\varepsilon = ([(f * c) - \delta - p])_- * R^a * [(f * c) - \delta - p])_0 + (c_- * R^u * c)_0 \tag{6A.2}$$

where '$_-$' denotes time-reversal and R_k^a denotes the autocorrelation function of a_k. By varying (6A.1) with respect to C as in Section 5.4 we obtain the following condition of

optimality for C:

$$[\mathcal{A}|F|^2 + \mathcal{U}]C = 1 + P \text{ for all } \Omega. \tag{6A.3}$$

Similarly, by differentiating (6A.2) with respect to the nontrivial FBF coefficients p_k, $k \geq 1$, we obtain the semi-infinite set of conditions

$$(R^a * [(f * c) - 1 - p])_k = 0 \text{ for all } k \geq 1. \tag{6A.4}$$

Hence $\mathcal{A}[FC - 1 - P]$ should correspond to an anticausal function.

A straightforward calculation on the basis of (6.22), (6.24) and (6.23) verifies (6A.3), and shows that

$$\mathcal{A}[FC - 1 - P] = -\frac{1}{\gamma_0 \Gamma^*}. \tag{6A.5}$$

Now Γ corresponds to a causal function whose poles and zeros are all inside the unit circle, and this clearly also applies to $1/\Gamma$. Thus $1/\Gamma^*$ corresponds to an anticausal function. The right hand side of (6A.5) is just a scaled version of this function. This verifies (6A.4). Finally, upon using (6.22), (6.24) and (6.23) in (6A.1) we obtain the first part of (6.25) after some elementary manipulations.

APPENDIX 6B

MMSE FSE and DFE of finite length

In this appendix we derive the MMSE DFE for the situation that the feedback filter is constrained to have a finite length N_{FB}. For the sake of generality we assume that the prefilter is fractionally spaced and also has a finite length (Fig. 6B.1). The received signal $r(t)$ is first applied to a fixed filter with impulse response $w(t)$ that suppresses out-of-band noise. It is then sampled at an oversampling factor L that is ideally high enough to avoid aliasing; the forthcoming analysis, however, remains valid when aliasing occurs. The sampled sequence y_n of sampling rate L/T is applied to a fractionally-spaced prefilter whose impulse response c_n is to be determined jointly with the response p_k of the feedback filter. The prefilter output is only needed once per symbol interval T is and is therefore downsampled by a factor L before the FBF output is subtracted. For reasons of causality the overall system has a strictly positive delay, i.e. at the moment when digit a_k is transmitted, the receiver will be producing a decision $\hat{a}_{k-k_0}$ that pertains to a past digit a_{k-k_0}. The system delay of k_0 symbol intervals is a design freedom that can be used to optimize performance for given filter lengths and vice versa. Optimization of pre- and feedback filter is aimed at minimizing the mean-square error ε between the decision variable $\tilde{a}_{k-k_0}$ and the past digit a_{k-k_0}.

The system of Fig. 6B.1 degenerates into a full-response fractionally-spaced equalizer (FSE) in the event that the feedback filter degenerates ($N_{FB} = 0$). It is both instructive and convenient to analyze this situation first, and to the consider the effect of the FBF in a second step. The analysis involves matrix computations and we start by expressing ε in matrix notation. Define a vector $\underline{c} \triangleq [c_0, \ldots, c_{N-1}]^T$ that contains all N

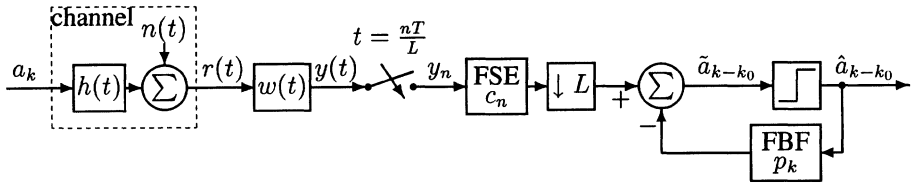


Figure 6B.1: DFE with fractionally-spaced prefilter of length N and feedback filter of length N_{FB} operating on the sampled output y_n of a receiving filter.

FSE coefficients and a vector $\underline{y}_k \triangleq [y_k, \ldots, y_{k-N+1}]^T$ that describes the tap signals of the FSE (the superscript 'T' denotes transposition). The FSE has downsampled output $\tilde{a}_{k-k_0} = \underline{y}_k^T \underline{c}$, and the MSE may be written $\varepsilon = E[(\underline{y}_k^T \underline{c} - a_{k-k_0})]^2$. The only stochastic variables in this expression are y_k and a_{k-k_0}. Upon expanding the square we find that

$$\varepsilon = \underline{c}^T \mathbf{M} \underline{c} + 2\underline{v}^T \underline{c} + \sigma_a^2 \tag{6B.1}$$

where $\mathbf{M} \triangleq E[\underline{y}_k \underline{y}_k^T]$ is an $N \times N$ autocorrelation matrix, $\underline{v} \triangleq E[\underline{y}_k a_{k-k_0}]$ is a cross-correlation vector of length N, and $\sigma_a^2 \triangleq E[a_k^2]$ is the data variance.

Exercise 6.3 Assume that data a_k is uncorrelated and has unit variance, and let $f_n \triangleq (h * w)(nT/L)$ denote the sampled system response up until the first sampler. Show that $\underline{v}$ has components $\underline{v}_i \triangleq f_{k_0 L - i}$ for $i = 0, \ldots, N-1$ and that the components of $\mathbf{M}$ are given by $\mathbf{M}_{i,j} \triangleq \sum_{k=-\infty}^{\infty} f_{kL-i} f_{kL-j} + R_{i-j}^u$ where R_i^u is the autocorrelation function of the filtered and sampled noise sequence $u_n \triangleq (n * w)(nT/L)$.

A unique minimum of ε exists when $\mathbf{M}$ is nonsingular. This is usually true, especially when L is small and noise is not altogether negligible. To minimize ε with respect to $\underline{c}$ we note that (6B.1) may be rewritten as

$$\varepsilon = (\underline{c} - \mathbf{M}^{-1})^T \mathbf{M} (\underline{c} - \mathbf{M}^{-1}\underline{v}) - \underline{v}^T \mathbf{M}^{-1} \underline{v} + \sigma_a^2 \tag{6B.2}$$

provided that $\mathbf{M}$ is indeed invertible. The only term that depends on $\underline{c}$ is non-negative and vanishes when $\underline{c} = \mathbf{M}^{-1}\underline{v}$. This is the optimum set of equalizer coefficients. The corresponding MMSE $\varepsilon_{\text{FRFSE}}$ is formed by the remaining two terms in (6B.2), i.e. $\varepsilon_{\text{FRFSE}} = \sigma_a^2 - \underline{v}^T \mathbf{M}^{-1} \underline{v}$. This completes our analysis for the FSE. We proceed to consider the DFE. For simplicity we take data a_k to be uncorrelated and we assume that error propagation is absent. Then the FBF should cancel all trailing ISI within its span. Thus, if the FBF has length N_{FB}, i.e. if $p_k = 0$ for $k \notin \{1, \ldots, N_{FB}\}$, then the decision variable $\tilde{a}_{k-k_0}$ will not be affected by any of the digits $\tilde{a}_{k-k_0-1}, \ldots, \tilde{a}_{k-k_0-N_{FB}}$. This can be accounted for by stripping the MSE of (6B.1) from any influence that stems from these N_{FB} digits. The result of this exercise is that ε still has the form (6B.1) but that $\mathbf{M}$ now has components

$$\mathbf{M}_{i,j} = \sum_{k \notin \{1, \ldots, N_{FB}\}} f_{(k+k_0)L-i} f_{(k+k_0)L-j} + R_{i-j}^u. \tag{6B.3}$$

The sum in this expression excludes indices k that cover the span of the FBF. Again the optimum prefilter has $\underline{c} = \mathbf{M}^{-1}\underline{v}$, and the MMSE amounts to $\varepsilon_{\text{FL DFE}} = \underline{v}^T\mathbf{M}^{-1}\underline{v}$. For all trailing ISI to be canceled, the impulse response of the FBF should match the tail of the system response, i.e. it can be found as $p_k = (f * c)_{kL}$ for all $k \in \{1, \ldots, N_{FB}\}$.

APPENDIX 6C

Optimum $d = 1$ DFE

The derivation in this appendix is tailored to a channel with a first-order zero at DC, but is easily amended to zeros at other frequencies as well as to a channel without zeros. If f_k has a first-order zero at DC then we may write $f_k = d_k - d_{k-1}$ where d_k is a finite-energy impulse response without spectral zeros. This leads to an equalized system response $q_k = (f * w)_k = h_k - h_{k-1}$ where $h_k \triangleq (d * w)_k$. The function q_k is constrained to be zero for $k < -N_p$ and $k > N_{FB}$ and to unity for $k = 0$. This implies that h_k must vanish for $k \notin S' \triangleq \{-N_p, \ldots, N_{FB} - 1\}$. The constraint $q_0 = 1$ is met if and only if $h_0 = 1 + h_{-1}$.

6C.1 Optimum unconstrained $d = 1$ DFE

Here the objective is to find the coefficients h_k for $k \in S'$ that minimize the noise variance at the prefilter output subject to the constraint $h_0 = 1 + h_{-1}$. Noise $(u * w)_k$ at the output of the prefilter has variance

$$\sigma^2 = \int_{-0.5}^{0.5} \mathcal{U}(e^{j2\pi\Omega})|W(e^{j2\pi\Omega})|^2 d\Omega = \int_{-0.5}^{0.5} \frac{\mathcal{U}(e^{j2\pi\Omega})}{|D(e^{j2\pi\Omega})|^2}|H(e^{j2\pi\Omega})|^2 d\Omega \qquad \text{(6C.1)}$$

where capitals denote discrete-time Fourier transforms and $\mathcal{U}(e^{j2\pi\Omega})$ denotes the PSD of u_k. In matrix notation this becomes $\sigma^2 = \underline{h}^T\mathbf{M}\underline{h}$ where $\underline{h} \triangleq [h_{-N_p}, \ldots, h_{N_{FB}-1}]^T$ is a vector of length $N_p + N_{FB}$, '^{T}' denotes transposition, and $\mathbf{M}$ is a symmetric $(N_p + N_{FB}) \times (N_p + N_{FB})$ Toeplitz matrix with coefficients $\mathbf{M}_{i,j} = x(i - j)$ where

$$x(k) \triangleq \int_{-0.5}^{0.5} \frac{\mathcal{U}(e^{j2\pi\Omega})}{|D(e^{j2\pi\Omega})|^2} \cos 2\pi\Omega k d\Omega. \qquad \text{(6C.2)}$$

In order to accommodate the constraint we define the Lagrangian $L \triangleq \underline{h}^T\mathbf{M}\underline{h} + \mu(\underline{h}^T\underline{v} - 1)$ where μ is a Lagrange multiplier and $\underline{v} = [\underline{v}_{-N_p}, \ldots, \underline{v}_{N_{FB}-1}]^T$ is a vector of length $(N_p + N_{FB})$ with components $\underline{v}_{-1} = -1$, $\underline{v}_0 = 1$ and $\underline{v}_k = 0$ else. For $N_p = 0$ the component $\underline{v}_{-1}$ does not 'fit' so that $\underline{v} = [1, 0, \ldots, 0]^T$. To minimize L we require that the gradients of L with respect to $\underline{h}$ and μ are zero. This leads to the conditions $\mathbf{M}\underline{h} = -\frac{\mu}{2}\underline{v}$ and $\underline{h}^T\underline{v} = 1$. These conditions have a unique solution provided that $\mathbf{M}$ is invertible, as we will assume. The minimizing noise power $\hat{\sigma}^2$ and the optimum

impulse response $\hat{\underline{h}}$ are given by

$$\hat{\sigma}^2 = \frac{1}{\underline{v}^T \mathbf{M}^{-1} \underline{v}} \quad \text{and} \quad \hat{\underline{h}} = \hat{\sigma}_u^2 \mathbf{M}^{-1} \underline{v}. \tag{6C.3}$$

6C.2 Optimum constrained $d = 1$ DFE

We next take $N_p = 2$ and add the constraint $q_{-2} = -q_{-1}$. Translated to the response h_k this constraint reads $h_{-1} = 0$ and since we had already required that $h_0 = 1 + h_{-1}$ it follows that h_0 must be unity. It remains to determine the coefficients h_{-2} and $h_1, \ldots, h_{N_{FB}-1}$, which we group together in a vector $\underline{h}$ of length N_{FB}. It is easy to see that

$$\sigma^2 = x(0) + 2\underline{h}^T \underline{v} + \underline{h}^T \mathbf{M} \underline{h} \tag{6C.4}$$

where $\underline{v} \triangleq [x(-2), x(1), x(2), \ldots, x(N_{FB} - 1)]^T$ is a vector of length N_{FB}, while $\mathbf{M}$ is a symmetric $N_{FB} \times N_{FB}$ matrix

$$\mathbf{M} \triangleq \begin{pmatrix} x(0) & x(-3) & x(-4) & \ldots & x(-N_{FB}-1) \\ x(3) & x(0) & x(-1) & & x(-N_{FB}+2) \\ x(4) & x(1) & x(0) & & x(-N_{FB}+3) \\ \vdots & & & & \vdots \\ x(N_{FB}+1) & x(N_{FB}-2) & x(N_{FB}-3) & \ldots & x(0) \end{pmatrix}. \tag{6C.5}$$

To minimize σ^2, we put the gradient of (6C.4) with respect to $\underline{h}$ equal to zero. This leads to the set of conditions $\mathbf{M}\underline{h} = -\underline{v}$. A unique optimizing vector $\hat{\underline{h}}$ exists when $\mathbf{M}$ is invertible and is given by $\hat{\underline{h}} = -\mathbf{M}^{-1}\underline{v}$. The corresponding minimum noise variance $\hat{\sigma}^2$ is given by $\hat{\sigma}^2 = x(0) + \hat{\underline{h}}^T \underline{v} = x(0) - \underline{v}^T \mathbf{M}^{-1} \underline{v}$.

PROBLEMS

6.1 Consider the transformation of Fig. 6.1 for the situation that g_k is an EPRIV or EEPRIV partial response ($G(\mathcal{D}) = (1-\mathcal{D})(1+\mathcal{D})^2$ or $(1-\mathcal{D})(1+\mathcal{D})^3$, respectively).

1. Determine the number of amplitude levels of b_k for EPRIV and EEPRIV in the event that a_k is binary and uncoded.
2. Redo the previous question for the event that a_k is run-length-limited with minimum run-length parameter $d = 1$ (see Section 4.8 for an explanation of this terminology).
3. Sketch the amplitude characteristics of the EPRIV and EEPRIV responses and compare them to those of Fig. 6.2. For what kind of channels can EPRIV and EEPRIV be advantageous?

6.2 Consider a discrete-time $1-\mathcal{D}$ system with white additive noise u_k of variance N_0. Data a_k is uncorrelated and binary with $a_k \in \{-1, 1\}$, and SNR = 20 dB. Determine the predetection SNRs of the Duobinary, Bipolar and Class IV ZF PREs for this channel, and compare these with the performances computed in Problems 5.8.3 and 5.9.

6.3 Redo Problem 5.1 for a $1 + \mathcal{D}$ system without excess bandwidth. Is the conclusion any different? How about $1 - \mathcal{D}$? Explain these results by reference to the frequency characteristics of the responses.

6.4 The canonical whitened matched filter according to (6.19) contains a T-spaced discrete-time filter with transfer function $C(e^{j2\pi\Omega}) \triangleq 1/[\gamma_0 \Gamma^*(e^{j2\pi\Omega})]$. Prove that this discrete-time filter is anti-causal, i.e. that its impulse response satisfies $c_k = 0$ for all $k > 0$.

6.5 Prove (6.19) and (6.20).

6.6 Reconsider the discrete-time $1 + \alpha\mathcal{D}$ channel of Problem 5.6. Assume that a discrete-time ZF DFE operates on the channel output y_k. Determine the loss in predetection signal-to-noise ratio of this DFE with respect to the matched-filter bound a) by a time-domain argument, and b) by a frequency-domain argument. Compare your results with those of Problem 5.6.3 and explain the differences qualitatively. For which values of $\alpha \in [-1, 1]$ is the difference largest? Hint for b): make use of the integral identity $\int_{-0.5}^{0.5} \ln(a + b\cos 2\pi x)dx = \ln[(a + \sqrt{a^2 - b^2})/2]$.

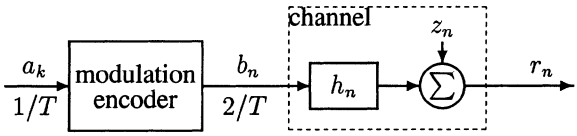


Figure 6P.1: Discrete-time system with modulation encoder having rate $R = 1/2$.

6.7 In the discrete-time system of Fig. 6P.1, binary data $a_k \in \{-1, +1\}$ of data rate $1/T$ is applied to a modulation encoder with rate $R = 1/2$. The encoded binary sequence $b_n \in \{-1, +1\}$ has data rate $2/T$ and is applied to a channel with impulse response

$$h_n = \begin{cases} \sqrt{2} & \text{for } n = 1, \\ 1 & \text{for } n = 0, 2, \text{ and} \\ 0 & \text{else.} \end{cases} \tag{6P.1}$$

The additive noise sequence z_n is white and Gaussian with variance N_0.

1. Is the matched-filter bound (MFB) attainable for this channel and arbitrary rate 1/2 modulation codes? If not, then how close can a maximum-likelihood receiver get?
2. Is zero-forcing linear equalization practicable for this channel?
3. What loss does a zero-forcing decision-feedback equalizer incur with respect to the MFB?
   Hint: First determine the phase characteristics of the channel and then use a time-domain analysis.

6.8 We reconsider the system of Problem 6.7 for a specific category of rate $R = 1/2$ codes, viz. those with a minimum run-length of 2 ($d = 1$). Examples of such codes are MFM and Miller-Squared (M^2) (see Section 4.8).

1. Is the MFB attainable for codes of this type? If so, why?
2. Show that a slicer suffices to recover b_n from the received sequence r_n as SNR $\to \infty$, i.e. that no equalization is necessary. What is the loss of this reception scheme with respect to the MFB?

A similar reasoning applies to rate 1/2 codes with $d = 2$ (such as (2,7)). These results explain in part why run-length limited codes are popular for use on low-pass channels.

Probably the simplest $d = 1$ code with rate $R = 1/2$ is the repetition code $b_{2k} = b_{2k+1} = a_k$ for all k. A demodulator for this code is shown in Fig. 6P.2.a. Successive pairs (r_{2k-1}, r_{2k}) of the received sequence r_n are combined into a sequence $y_k = r_{2k-1} + r_{2k}$ with the same data rate $1/T$ as a_k. Because encoder, channel and demodulator are all linear, y_k may be cast in the linear form $y_k = (a * f)_k + u_k$ where u_k is the noise component of y_k and f_k is an impulse response that accounts for filtering of the data a_k by encoder, channel and demodulator. The corresponding equivalent discrete-time channel is shown in Fig. 6P.2.b.

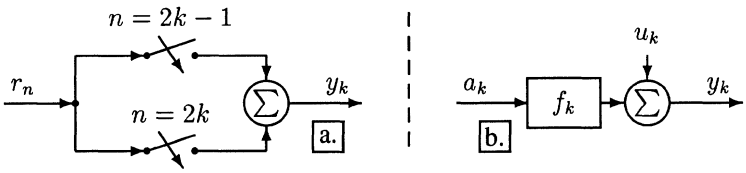


Figure 6P.2: a. Demodulator operating on the received sequence r_n of Fig. 6P.1. b. Equivalent discrete-time model for the cascade of repetition encoder, channel and demodulator.

1. Determine f_k, its Fourier transform $F(e^{j2\pi\Omega})$, the power spectral density $\mathcal{U}(e^{j2\pi\Omega})$ of u_k, the folded signal-to-noise ratio $\text{FSN}(e^{j2\pi\Omega}) = |F(e^{j2\pi\Omega})|^2/\mathcal{U}(e^{j2\pi\Omega})$, and the average signal-to-noise ratio $\text{SNR} = \int_{-0.5}^{0.5} \text{FSN}(e^{j2\pi\Omega})d\Omega$.
2. Is the MFB attainable for the channel of Fig. 6P.2.b? Explain and quantify any discrepancy with the results of Problem 6.7.1.
3. Compute the loss with respect to the MFB of zero-forcing linear and decision-feedback equalizers operating on y_k. Hint: Make use of the integral identities $\int_{-0.5}^{0.5}(a + b\cos 2\pi x)^{-2}dx = a(a^2 - b^2)^{-3/2}$ and $\int_{-0.5}^{0.5} \ln(a + b\cos 2\pi x)dx = \ln[(a + \sqrt{a^2 - b^2})/2]$.

6.9 Use (6.24) to prove the first part of (6.25).

6.10 In this problem we consider the effect of data correlation on the discrete-time MMSE DFE of Section 6.3. For notational convenience we suppress all dependencies on Ω.

1. Express the equalized system characteristic FC in terms of G, $|F|^2/\mathcal{U}$ and $\mathcal{A}$. Subject to which condition does the equalized system have minimum phase? Relate this condition to the ZF DFE. Hint: Use (6.24) and the fact that G has minimum phase.
2. Express FC in terms of Γ, γ_0 and $\mathcal{A}$. Subject to which condition does the FBF cancel all trailing ISI? Does this condition ensure that the equalized system has minimum phase?

6.11 Subject to which limiting condition does the optimum MMSE DFE approach the optimum ZF DFE? Is this condition likely to be met in practice?

6.12 Consider the systems of Figs. 6.14.a and 6.14.b. Assume that data a_k is uncorrelated and in the PAM alphabet $a_k \in \{\pm 1, \pm 3, \ldots \pm (M-1)\}$, and that the precoder in Fig. 6.14.b causes s_k to be uncorrelated and uniformly distributed across the interval $(-M, M]$. Denote the data component $(s * g)_k$ of y_k by b_k and its power $E[b_k^2]$ by P_b. Determine P_b for the configurations of Figs. 6.14.a and 6.14.b. Which of the two is largest? Subject to which condition are they equal? Is there any advantage to g_k having minimum phase?

6.13 Consider the Tomlinson/Harashima precoding technique of Section 6.4 for the situation that coefficients g_k are all integer-valued. Show that all samples s_k in Fig. 6.14.b are odd integers. What values can d_k assume? Comment on the complexity of the precoder.

6.14 Consider a discrete-time $1 + \mathcal{D}$ system with unipolar binary data $a_k \in \{0, 1\}$. Show that the Tomlinson/Harashima precoder with $M = 2$ for this channel is equivalent to a regular precoder as in Section 6.2.

6.15 Consider the detector of Fig. 6.15 for an arbitrary target response g_k. Show that the required FBF is identical to the one of a feedback detector without noise whitening as in Fig. 6.4.b, except for the first FBF tap.

6.16 Consider the detector of Fig. 6.16 for an arbitrary target response g_k. Assume that the first-order LPF is replaced by a stable N^{th} order all-pole network so as to be able perform more accurate noise whitening. Show that the required FBF is identical to the one of a feedback detector without noise whitening (Fig. 6.4.b) except for the first N FBF taps.

REFERENCES

[1] M.E. Austin, Sc.D. Dissertation, MIT, Cambridge, Massachussets, May 1967.

[2] J.R. Barry, E.A. Lee and D.G. Messerschmitt, 'Capacity Penalty due to Ideal Zero-Forcing Decision-Feedback Equalization', *IEEE Trans. Inform. Theory*, Vol. 42, No. 4, pp. 1062-1071, July 1996.

[3] P.S. Bednarz, N.P. Sands, C.S. Modlin, S.C. Lin, I. Lee, and J.M. Cioffi, 'Performance Evaluation of an Adaptive RAM-DFE Read Channel', *IEEE Trans. Magn.*, Vol. MAG-31, No. 2, pp. 1121-1127, March 1995.

[4] R.T. Behrens and A.J. Armstrong, 'An Advanced Read/Write Channel for Magnetic Disk Storage', *Proc. 26th Asilomar Conf. on Signals, Systems, and Computers*, Oct. 26-28, 1992, pp. 956-960.

[5] C.A. Belfiore and J.H. Park, Jr., 'Decision Feedback Equalization', *Proc. IEEE*, Vol. 67, No. 8, pp. 1143-1156, Aug. 1979.

[6] W.R. Bennett and J.R. Davey, *Data Transmission*. New York: McGraw-Hill, 1965.

[7] J.W.M. Bergmans, 'A Bit-By-Bit Detector for Partial-Response Reception on Digital Magnetic Recording Channels', *Proc. GLOBECOM'94*, pp. 1148-1151, San Fransisco, California, Nov. 27 - Dec. 2, 1994.

[8] J.W.M. Bergmans, 'Decision Feedback Equalization for Run-Length-Limited Modulation Codes with $d = 1$', *IEEE Trans. Magn.*, 1996.

[9] J.M. Cioffi, G.P. Dudevoir, M.V. Eyuboğlu, and G.D. Forney, Jr., 'MMSE Decision-Feedback Equalizers and Coding - Part II: Coding Results', *IEEE Trans. Commun.*, Vol. 43, No. 10, pp. 2595-2604, Oct. 1995.

[10] A.A. Clokey, 'Automatic Printing Equipment for Long Loaded Submarine Telephone Cables', *Bell Syst. Tech. J.*, Vol. 6, pp. 402-424, July 1927.

[11] J.L. Doob, *Stochastic Processes*. New York: John Wiley and Sons, 1967.

[12] A. Duel-Hallen, 'Equalizers for Multiple Input/Multiple Output Channels and PAM Systems with Cyclostationary Input Sequences', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 3, pp. 630-639, Apr. 1992.

[13] D.L. Duttweiler, J.E. Mazo, and D.G. Messerschmitt, 'An Upper Bound on the Error Probability in Decision-Feedback Equalization', *IEEE Trans. Inform. Theory*, Vol. IT-220, pp. 490-497, July 1974.

[14] M.J. Ferguson, 'Optimal Reception for Binary Partial Response Channels', *Bell Syst. Tech. J.*, Vol. 51, No. 2, pp. 493-505, Feb. 1972.

[15] R. Fischer, Using Flexible Precoding for Channels with Spectral Nulls', *Electron. Lett.*, Vol. 31, No. 5, pp. 356-358, March 2, 1995.

[16] K.D. Fisher, J.M. Cioffi, W.L. Abbott, P.S. Bednarz, and C.M. Melas, 'An Adaptive RAM-DFE for Storage Channels', *IEEE Trans. Commun.*, Vol. COM-39, No. 11, pp. 1559-1568, Nov. 1991.

[17] G.D. Forney, Jr., 'Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference', *IEEE Trans. Inform. Theory*, Vol. IT-18, No. 3, pp. 363-378, May 1972.

[18] G.D. Forney and M.V. Eyuboğlu, 'Combined Equalization and Coding Using Precoding', *IEEE Commun. Magaz.*, pp. 25-34, Dec. 1991.

[19] P.J. van Gerwen, 'On the Generation and Application of Pseudo-Ternary Codes in Pulse Transmission', *Philips Res. Rept.*, Vol. 20, pp. 469-484, Aug. 1965.

[20] P.J. van Gerwen, N.A.M. Verhoeckx, and T.A.C.M. Claasen, 'Design Considerations for a 144 kb/s Digital Transmission Unit for the Local Telephone Network', *IEEE J. Selected Areas Commun.*, Vol. SAC-2, pp. 314-323, March 1984.

[21] H. Harashima and H. Miyakawa, 'Matched-Transmission Technique for Channels with Intersymbol Interference', *IEEE Trans. Commun. Technol.*, Vol. COM-20, pp. 774-780, Aug. 1972.

[22] P. Kabal and S. Pasupathy, 'Partial-Response Signaling', *IEEE Trans. Commun.*, Vol. COM-23, No. 9, pp. 921-934, Sept. 1975.

[23] J.G. Kenney, L.R. Carley and R.W. Wood, 'Multi-Level Decision Feedback Equalization for Saturation Recording', *IEEE Trans. Magn.*, Vol. 29, No. 3, pp. 2160-2171, July 1993.

[24] H. Kobayashi, 'Application of Probabilistic Decoding to Digital Magnetic Recording Systems', *IBM J. Res. Develop.*, pp. 64-74, Jan. 1971.

[25] E.R. Kretzmer, 'Generalization of a Technique for Binary Data Transmission', *IEEE Trans. Commun. Technol.*, Vol. COM-14, pp. 67-68, Feb. 1966.

[26] R. Laroia, S.A. Tretter, and N. Farvardin, 'A Simple and Effective Precoding Scheme for Noise Whitening on Intersymbol Interference Channels', *IEEE Trans. Commun.*, Vol. COM-41, No. 10, pp. 1460-1463, Oct. 1993.

[27] A. Lender, 'The Duobinary Technique for High-Speed Data Transmission', *IEEE Trans. on Commun. Electron.*, Vol. 82, pp. 214-218, May 1963.

[28] J.-Y. Lin and C.-H. Wei, 'Adaptive Nonlinear Decision Feedback Equalization with Channel Estimation and Timing Recovery in Digital Magnetic Recording Systems', *IEEE Trans. Circuits Syst. -II*, Vol. 42, No. 3, pp. 196-206, March 1995.

[29] R.C. Mathes, U.S. Patent 1 295 553, Feb. 25, 1919.

[30] D.G. Messerschmitt, 'Design of a Finite Impulse Response for the Viterbi Algorithm and Decision-Feedback Equalizer', *Proc. ICC'74*, Minneapolis, MN, June 17-19, 1974.

[31] D.G. Messerschmitt, 'Generalized Partial Response for Equalized Channels with Rational Spectra', *IEEE Trans. Commun.*, Vol. COM-23, No. 11, pp. 1251-1258, Nov. 1975.

[32] P. Monsen, 'Feedback Equalization for Fading Dispersive Channels', *IEEE Trans. Inform. Theory*, Vol. IT-17, pp. 56-64, Jan. 1971.

[33] E.A. Newcombe and S. Pasupathy, 'Effects of Filtering Allocation on the Performance of a Modified Duobinary System', *IEEE Trans. Commun.*, Vol. COM-28, No. 5, pp. 749-752, May 1980.

[34] B.R. Petersen and D.D. Falconer, 'Minimum Mean-Square Equalization in Cyclostationary and Stationary Interference - Analysis and Subscriber Line Calculations', *IEEE J. Selected Areas Commun.*, Vol. SAC-9, No. 6, pp. 931-940, Aug. 1991.

[35] R. Price, 'Nonlinearly Feedback-Equalized PAM vs. Capacity for Noisy Filter Channels', *Proc. ICC'74*, Minneapolis, MN, June 17-19, 1974.

[36] J.J. O'Reilly and A.M. de Oliveira Duarte, 'Error Propagation in Decision Feedback Receivers', *IEE Proc.*, Vol. 132, Pt. F, No. 7, pp. 561-566, Dec. 1985.

[37] J. Salz, 'Optimum Mean-Square Decision Feedback Equalization', *Bell Syst. Tech. J.*, Vol. 52, No. 8, pp. 1341-1373, Oct. 1973.

[38] H.K. Thapar and A.M. Patel, 'A Class of Partial Response Systems for Increasing Storage Density in Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-23, No. 5, pp. 3666-3668, Sept. 1987.

[39] M. Tomlinson, 'New Automatic Equalizer Employing Modulo Arithmetic', *Electron. Lett.*, Vol. 7, Nos. 5/6, pp. 138-139, March 1971.

7

VITERBI DETECTION

7.1 INTRODUCTION

In Chapter 3 we have observed that maximum-likelihood detection of a data sequence in the presence of ISI and noise becomes enormously complicated for long messages when done in a 'brute-force' manner. In the present chapter we develop the Viterbi detector (VD), which performs maximum-likelihood detection in a much more efficient fashion. Basic to the VD is a technique called *dynamic programming* that we introduce in Section 7.2 with the aid of an example. Dynamic programming originated as a solution to the classical shortest path problem, but can also be tailored to data detection (Sections 7.3 and 7.4). The recursive nature of the technique causes total processing effort to grow only linearly with the message length, as opposed to the exponential growth for the 'brute-force' receiver of Chapter 3.

At the heart of the VD is a so-called *trellis diagram* that is determined by the modulation code and the memory length of the system (Section 7.4). The maximum-likelihood data sequence is basically a path through this diagram. To be able to identify this path, the VD contains a *path memory* in which a fixed number of candidate data sequences are kept track of, one for each *state* in the trellis diagram. The path memory is controlled by a set of coupled *add-compare-select units* which keep track of the relative likelihoods of the candidate sequences. These units, in turn, are driven by *branch metric computation units* which convert received data symbols into incremental likelihood information. Features and implementation aspects of these building blocks are discussed in Section 7.5.

Even though it is dramatically simpler than the 'brute-force' receiver of Chapter 3, the archetypal VD can still be unduly complicated, especially for data alphabets with many levels and channels with a large memory length. Unfortunately, further simplifications fundamentally cause a detraction from optimality. Even so, in the past few decades numerous approaches have been uncovered that permit substantial complexity reductions at a modest and sometimes negligible expense to performance. In view of their practical importance we devote ample space to these simplifications. Four approaches are elaborated:

1. *Channel memory truncation by means of prefiltering* (Section 7.6). At the cost of noise enhancement, this technique permits a large reduction of the number of candidate sequences that the VD must keep track of. In some cases this number can, in fact, be as small as two and this permits further simplifications (Section 7.6.1). Of particular interest is a scheme due to Ferguson that has found widespread application in digital magnetic recording (Sections 7.6.2-7.6.3).

2. *Truncation via sequence feedback* (Section 7.7). This manner to simplify the VD can be regarded as an extension of decision feedback equalization. It avoids noise enhancement yet fails to exploit all data power in the detection process. A suitable pre-equalizer will minimize this power loss.

3. *Set partitioning* (Section 7.8). This approach attacks the impact on VD complexity of the number of data levels, and forms a natural complement to sequence feedback.

4. *Restricted-delay detection* (Section 7.9). Here the detectors' capability to defer decisions is restricted to a modest but acceptable level.

These four techniques are largely complementary and tend to be quite effective. Their systematic nature facilitates a performance analysis and leads to hardware structures of considerable regularity. In these regards they are favorable to many of the *ad hoc* simplifications that have pervaded the literature (see [64, 12, 2] for a small sample).

Throughout the chapter the emphasis is on concepts rather than mathematics. For this reason some of the more theoretical subjects have been relegated to appendices. These cover the performance consequences of a finite detection delay (Appendix 7B), of the effect of noise coloration as induced by prefiltering (Appendix 7C), and of sequence feedback (Section 7D). Also relegated to appendices are two detector types that are tailored to specific combinations of modulation code and equalizer type (DC-free codes with full-response and $1 - \mathcal{D}$ reception: Appendix 7A; run-length-limited codes with minimum run-length parameter $d = 1$ and $1 - D$ reception: Appendix 7E).

7.2 DYNAMIC PROGRAMMING

At the heart of the VD is a technique that is called *dynamic programming*. We introduce this technique with an example that dates back to the cold-war era (Fig. 7.1).

A Dutch person living in the Noord-Oost Polder (NOP) wishes to visit his acquaintances in Wrocław (W), Poland. To this end, he must cross the border between Holland and West-Germany near either Enschede (E) or Venlo (V). From here on he may head to either Braunschweig (B) or Kassel (K) near East-Germany. A final choice is between Görlitz (G) and Frankfurt-am-Oder (F), both facing the Polish border. In keeping with Dutch national character, our friend wishes to take the cheapest and hence shortest route to his destination. The distances along all possible stretches being given, it would be a simple matter to compute the total length of all 8 possible routes and select the shortest

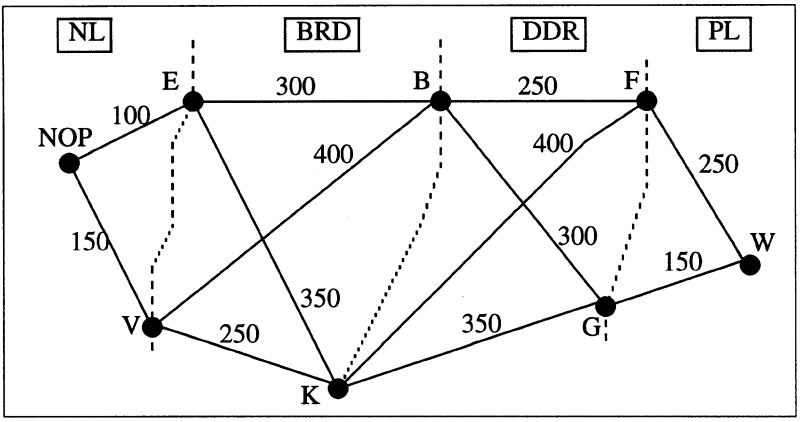


Figure 7.1: Possible routes between the Noord-Oost Polder (NOP) in Holland and Wrocław (W) in Poland.

among them. Reluctant to perform so many computations, our friend embarks on an alternative, stage-wise approach:

1. To get to the East-German border, he must reach either Braunschweig or Kassel. Among the two routes to Braunschweig, the shortest one is via Enschede (400 km), and the one via Venlo is discarded from further consideration. Similarly, the shortest of both routes to Kassel is via Venlo (400 km), and the one via Enschede is discarded (Fig. 7.2).

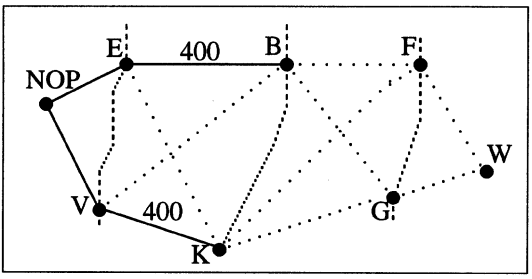


Figure 7.2: Surviving routes to Braunschweig (B) and Kassel (K).

2. To get to the Polish border at Frankfurt would take 400+250=650 km via the shortest route to Braunschweig, and 400+400=800 via the shortest route to Kassel. The route via Braunschweig survives, and the one via Kassel is discarded. Similarly, the shortest route to Görlitz via Braunschweig is 400+300=700 km versus 400+350=750 via Kassel. Again the route via Braunschweig survives, and the one via Kassel is discarded (Fig. 7.3).

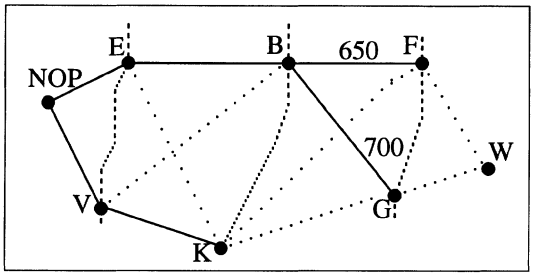


Figure 7.3: Surviving routes to Frankfurt-am-Oder (F) and Görlitz (G).

3. To reach Wrocław via Frankfurt would take 650+250=900 km along the shortest route, versus 700+150=850 via Görlitz. Hence the route via Enschede, Braunschweig and Görlitz is shortest among all routes to Wrocław (Fig. 7.4).

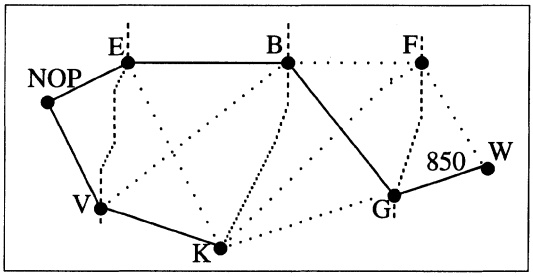


Figure 7.4: Surviving route to Wrocław (W).

This procedure is a special case of *forward dynamic programming*. By splitting the shortest path problem up into stage-wise subproblems that are easily solved, the computational burden of a straightforward 'brute-force' approach is avoided. This advantage becomes larger as the number of stages *en route* increases: For every additional stage the total number of possible paths doubles, whereas only one additional stage of computations is necessary with dynamic programming. We proceed to reformulate maximum-likelihood sequence detection in the form of a shortest-path problem to which dynamic programming is applicable.

7.3 TWO-STATE VITERBI DETECTION

Fig. 7.5 depicts a $1-\mathcal{D}$ channel with binary input $a_k \in \{-1, 1\}$. The received sequence z_k consists of a data component $x_k \triangleq a_k - a_{k-1}$ and a white noise component n_k. For the sake of simplicity we assume that a message of only $N = 3$ bits a_0, a_1, a_2 is transmitted. Digits a_k with $k \notin \{0, 1, 2\}$ are taken to be zero. Because of the memory of the channel,

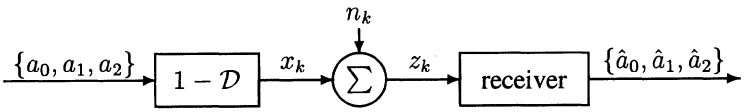


Figure 7.5: Discrete-time system model with $1 - \mathcal{D}$ partial-response channel and white noise n_k.

there are 4 noiseless channel outputs x_k that may differ from zero: x_0 and x_3 are binary ($x_0 = a_0 - a_{-1} = a_0 \in \{-1, 1\}$; similarly $x_3 = a_3 - a_2 = -a_2 \in \{-1, 1\}$), while x_1 and x_2 are ternary ($x_1, x_2 \in \{-2, 0, 2\}$). These possible outputs are depicted in the *trellis diagram* of Fig. 7.6.

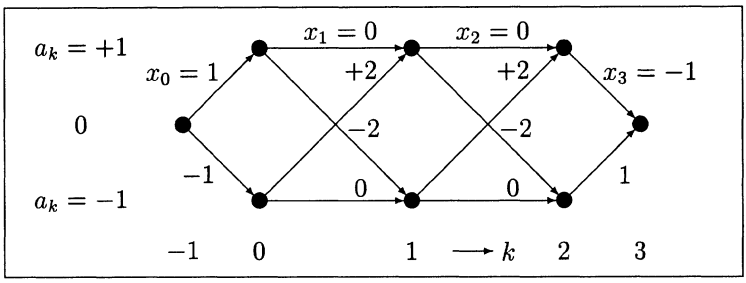


Figure 7.6: Trellis for detection of $N = 3$ binary data symbols a_0, a_1 and a_2 on a $1 - \mathcal{D}$ partial-response channel.

The nodes represent the possible digits a_k for $k = -1, \ldots, 3$. Transitions between nodes are called *branches*. Every branch uniquely specifies both a previous digit a_{k-1} and a subsequent digit a_k, and thereby a noiseless channel output $x_k = a_k - a_{k-1}$. This symbol value is noted alongside each transition. Every possible data sequence $\underline{a} \triangleq (a_0, a_1, a_2)$ corresponds to a single path through the trellis and vice versa.

It was established in Chapter 3 for white Gaussian noise n_k that maximum-likelihood sequence detection boils down to finding the admissible sequence $\underline{x} \triangleq (x_0, x_1, x_2, x_3)$ that is closest to the detector input $\underline{z} \triangleq (z_0, z_1, z_2, z_3)$ in the Euclidian sense. The Euclidian distance λ between $\underline{z}$ and $\underline{x}$ is

$$\lambda \triangleq \sum_{k=0}^{3} (z_k - x_k)^2. \tag{7.1}$$

This is a sum of 4 distance increments $\beta_k \triangleq (z_k - x_k)^2$ ($k = 0, 1, 2, 3$). These increments are commonly designated *branch metrics* because each of them is uniquely associated with a single branch in Fig. 7.6. All metrics β_k may be computed as soon as the kth detector input z_k becomes available. The receiver must determine the path through the trellis for which the sum of all branch metrics is smallest. This brings us back to the shortest path problem of Section 7.2. The topology of Fig. 7.6 is identical

to that of Fig. 7.1, each branch in Fig. 7.6 has a known distance β associated with it, and the objective is to find the shortest complete path through the trellis. The Viterbi detector (VD) does this by following the recursive procedure of Section 7.2. It progresses in a stage-wise fashion through the trellis. At each stage k, it finds the *surviving data paths* that lead to the nodes $a_k = +1$ and $a_k = -1$ as well as the *path metrics* λ_k^+ and λ_k^- that define the length (i.e. the sum of branch metrics) of both surviving paths. Main advantage of this approach is that computational effort only increases linearly with the message length. Since data symbols are normally transmitted at a fixed pace, this means that required processing effort per unit of time is fixed. This elementary condition of practicality is not met by the 'brute force' search of Chapter 3, for which computational effort increases exponentially with message length.

The trellis diagram of Fig. 7.6 owes its simplicity to the fact that data a_k is binary and that the channel has a memory length of only one symbol interval. We proceed to develop more general trellis diagrams and Viterbi detectors.

7.4 EFFECT OF TRANSMISSION CODE AND CHANNEL ON TRELLIS DIAGRAM

In Fig. 7.7.a, a data sequence a_k is applied to a continuous-time noisy dispersive channel. The received signal $r(t)$ is applied to a prefilter and sampled in synchronism with the data sequence. The prefilter is ideally a whitened matched filter (see Section 3.10.2). The sampler output z_k serves as detector input and may be modeled as in Fig. 7.7.b. The data component component x_k of z_k is a filtered version of a_k, and the noise component n_k is ideally white.

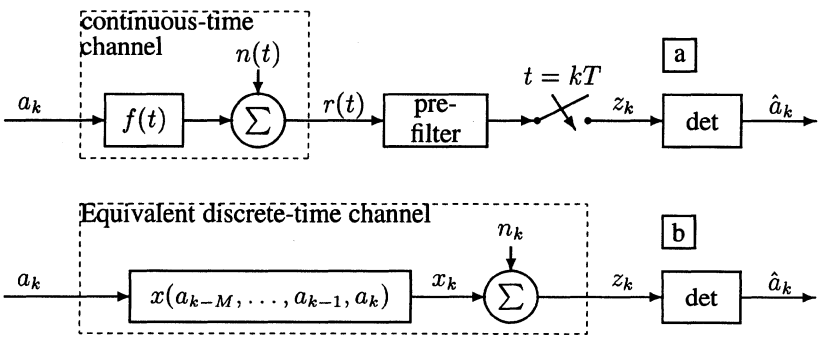


Figure 7.7: a. Transmission system with continuous-time channel, prefilter, sampler and detector. b. Equivalent discrete-time model.

Viterbi detection is possible whenever the equivalent channel of Fig. 7.7.b has a *finite* memory length M. Apart from a possible delay, this means that the output z_k depends only on the current digit a_k and the M most recent digits $a_{k-M}, \ldots, a_{k-1}$. To

stress this fact we write

$$z_k = x(\mathbf{s}_k, a_k) + n_k \tag{7.2}$$

where the *state*

$$\mathbf{s}_k \triangleq [a_{k-M}, \ldots, a_{k-1}] \tag{7.3}$$

accounts for the memory of the channel. Note that the formulation (7.2) also applies to nonlinear input-output relations of memory length M. In the linear case we have

$$x(\mathbf{s}_k, a_k) = \sum_{i=0}^{M} h_i a_{k-i}, \tag{7.4}$$

where h_k is the symbol response of the equivalent channel. The set S of possible states $\mathbf{s}_k$ depends on the data alphabet and the applied transmission code as well as on M. The cardinality $|S|$ of S may be large, but it is fixed and finite. This is essential for applicability of the VD. We first consider uncoded transmission in more detail.

Uncoded transmission Let data a_k be uncoded with L possible amplitude levels. Then there are $|S| = L^M$ possible states. Thus $|S|$ depends heavily on both L and M.

Example 7.1 The system of Section 7.3 has $L = 2$ and $M = 1$, whence $|S| = 2$.

Digital recording systems normally use binary data ($L = 2$), and ISI may well extend across 10 symbol intervals. If the prefilter does not suppress ISI, then there are $|S| \simeq 2^{10} \simeq 10^3$ states.

Various ISDN transmission systems use uncoded 4-level data. The channel can easily have a memory length of 20 symbol intervals; typical ISDN prefilters do not suppress ISI significantly. In that case $|S|$ can be in the order of $4^{20} \simeq 10^{12}$.

From (7.2) we note that the noiseless detector input x is determined by the current state $\mathbf{s}_k$ and the current digit a_k. This digit, in turn, is fully determined by the next state $\mathbf{s}_{k+1}$ (see (7.3)). This means that x is fully determined by the pair $(\mathbf{s}_k, \mathbf{s}_{k+1})$, i.e. by a *transition* between states. Once the kth sampler output z_k has been received, this also determines the branch metric $\beta_k = (z_k - x_k)^2$ fully. Thus we may take the succession of all possible states $\mathbf{s}_k$ as a basis of the trellis diagram.

Example 7.2 Let us consider a system with uncoded quaternary data $a_k \in \{-3, -1, 1, 3\}$ ($L = 4$) and a channel with memory length $M = 1$. Here $\mathbf{s}_k = a_{k-1}$, and there are $L^M = 4$ possible states $\mathbf{s}^0, \ldots, \mathbf{s}^3$ (corresponding to $a_{k-1} = -3, -1, 1$ and 3, respectively). We assume that a total of $N = 5$ digits $a_0, \ldots, a_4$ is transmitted. The initial state $\mathbf{s}_0$ (a_{-1}) and final state $\mathbf{s}_{N+1}$ (a_N) are both assumed to be known. In the trellis diagram of Fig. 7.8, states are represented as nodes and time progresses step-wise from left to right.

Since transmission is uncoded, any succession of digits $a_0, \ldots, a_{N-1}$ is possible. Correspondingly, any succession of states $\mathbf{s}_1, \ldots, \mathbf{s}_N$ is possible. Transitions $(\mathbf{s}_k, \mathbf{s}_{k+1})$ uniquely determine a noiseless detector input $x(a_{k-1}, a_k)$ and thereby a branch metric $\beta = [z_k - x(a_{k-1}, a_k)]^2$. These outputs could be noted alongside transitions as in Fig. 7.6, but that would render the picture unduly complicated.

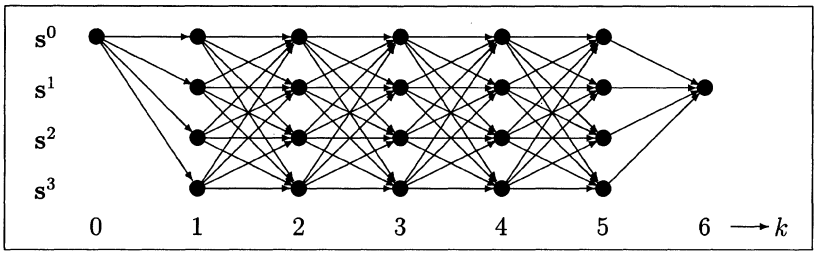


Figure 7.8: Trellis diagram for uncoded transmission of a message of $N = 5$ quaternary digits across a channel of memory length $M = 1$.

$\mathbf{s}_k$	a_{k-2}	a_{k-1}	a_k	$\mathbf{s}_{k+1}$
$\mathbf{s}^0$	$-$	$-$	$-$	$\mathbf{s}^0$
			$+$	$\mathbf{s}^1$
$\mathbf{s}^1$	$-$	$+$	$-$	$\mathbf{s}^2$
			$+$	$\mathbf{s}^3$
$\mathbf{s}^2$	$+$	$-$	$-$	$\mathbf{s}^0$
			$+$	$\mathbf{s}^1$
$\mathbf{s}^3$	$+$	$+$	$-$	$\mathbf{s}^2$
			$+$	$\mathbf{s}^3$

Table 7.1: Possible states and their successors for uncoded binary signaling across a channel with memory length $M = 2$.

Example 7.3 Consider transmission of an uncoded message of $N = 5$ binary digits $a_0, \ldots, a_4 \in \{-1, 1\}$ ($L = 2$) across a channel with memory length $M = 2$. There are $|S| = L^M = 4$ states $\mathbf{s}_k = [a_{k-2}, a_{k-1}]$, just as many as in the previous example. We label these states as in Table 7.1. For each state $\mathbf{s}_k$ the table also summarizes the new state $\mathbf{s}_{k+1} = [a_{k-1}, a_k]$ that arises for both possible choices of the next digit a_k. Old and new state together determine the triplet a_{k-2}, a_{k-1}, a_k, which in turn determines the noiseless channel output $x(a_{k-2}, a_{k-1}, a_k)$. Unlike the previous example, only two new states can be reached from every old state. This leads to a sparser trellis diagram (Fig. 7.9).

The initial state $\mathbf{s}_0 = [a_{-2}, a_{-1}]$ is assumed known. This implies that a_{-1} is known, i.e. that there are only two possible states $\mathbf{s}_1 = [a_{-1}, a_0]$. In the same way there are only two possible states $\mathbf{s}_6 = [a_4, a_5]$, and only a single terminal state $\mathbf{s}_7 = [a_5, a_6]$. Just as in previous examples, a unique noiseless detector input x_k is associated to every branch in the trellis, and a corresponding branch metric $\beta_k = (z_k - x_k)^2$ can be determined as soon as the kth detector input z_k becomes available.

The trellis diagrams of Figs. 7.8 and 7.9 consist of a 'stationary' part with $|S|$ states, and 'heading' and 'trailing' parts with less than $|S|$ states. The duration of the latter parts grows linearly with M and is independent of N (see Problem 7.6). Conversely, the duration of the 'stationary' part depends only on the message length N. This part consists of identical stages, one for each symbol interval k. Each stage involves a fixed

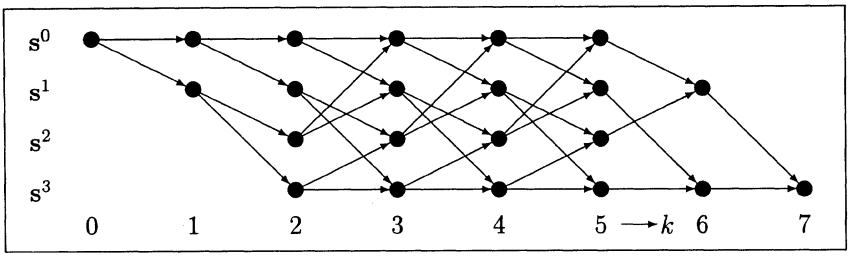


Figure 7.9: Trellis diagram for uncoded transmission of a message of $N = 5$ binary digits across a channel of memory length $M = 2$.

set of operations (computation of branch metrics, addition of branch metrics to path metrics, metric comparison, selection of surviving paths and path metrics). Thus the complexity of the detection process grows linearly with N, as opposed to the exponential growth for the 'brute-force' MLSD of Chapter 3. Since digits are normally transmitted at a regular pace, a *fixed* processing power will do to carry out the detection process, irrespective of the message length N. This power is determined by a single trellis segment.

Effect of coding Coding may simplify the trellis but may also make it more complicated. We provide two illustrations.

Example 7.4 In digital recording systems, binary modulation codes often introduce a so-called d-constraint that characterizes the smallest allowed number $d+1$ of consecutive identical bits a_k (see Section 4.8 for an introduction to run-length-limited codes). In this example we take $d = 1$, so that strings like '$\ldots + + - + + \ldots$' and '$\ldots - - + - - - \ldots$' are forbidden while '$\ldots + + - - + + \ldots$' and '$\ldots - - + + - - - \ldots$' are allowed. We first consider a channel memory length $M = 2$. Here the state $\mathbf{s}_k = [a_{k-2}, a_{k-1}]$ is a string of 2 consecutive bits. The four possible strings $--$, $-+$, $+-$ and $++$ are all compatible with the $d = 1$ constraint, i.e. there are four possible states, just as in the uncoded case. Transitions between states represent triplets a_{k-2}, a_{k-1}, a_k. The triplets $-+-$ and $+-+$ violate the $d = 1$ constraint and are hence forbidden. As a result, the stationary part of the trellis has only 6 branches per stage (Fig. 7.10, left) as opposed to 8 branches in the uncoded case (Fig. 7.9). For the sake of clarity, states in Fig. 7.10 are represented as strings, and only a single stage of the trellis diagram is shown.

We next consider a channel memory length $M = 3$. Here the state $\mathbf{s}_k = [a_{k-3}, a_{k-2}, a_{k-1}]$ is a string of 3 consecutive bits. The strings '$-+-$' and '$+-+$' violate the $d = 1$ constraint and are hence forbidden. What remains is a set S of $|S| = 6$ possible states (Fig. 7.10, right). Some transitions between states violate the d-constraint and are also forbidden. Extension of the old state $\mathbf{s}_k =$'$--+$' by $a_k = +1$ leads to the new state $\mathbf{s}_{k+1} =$'$-++$'. The alternative extension would lead to the forbidden state '$-+-$' and is therefore forbidden. For the same reason the state $\mathbf{s}_k =$'$++-$' also has only one allowed extension. Both possible extensions of the remaining states are allowed. The total number of branches B in the trellis segment amounts to 10. By comparison, $|S| = 8$ and $B = 16$ are needed for uncoded transmission. Thus the d-constraint indeed simplifies the trellis. This simplification is used, for example, in the VD of [49], which is meant for an EPRIV $(1 + \mathcal{D} - \mathcal{D}^2 - \mathcal{D}^3)$ channel.

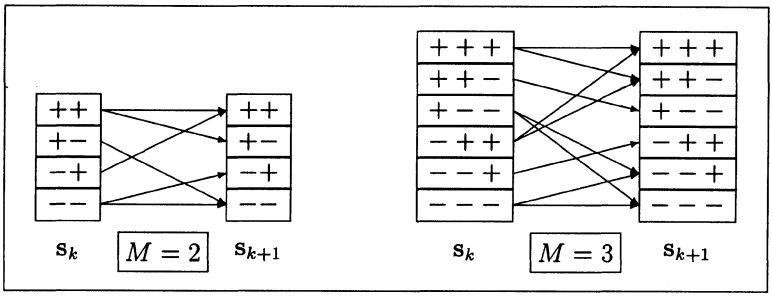


Figure 7.10: Trellis segments for binary modulation code with $d = 1$ and two channel memory lengths M.

Example 7.5 In Section 4.9 we have seen that the use of a charge-constrained modulation code can be advantageous for channels with a poor response at low frequencies. Charge-constrained codes create a spectral zero at DC by ensuring that the *charge* $c_k \triangleq \sum_{i=0}^{k} a_k$ (also called *running digital sum*) of the coded data symbols a_k is bounded between finite lower and upper limits C_l and C_u. The effect of the DC constraint can be visualized in an *unbalance trellis diagram* that depicts the possible evolutions of c_k over time. Fig. 7.11 provides an illustration for Miller-squared, which has $C_l = C_u = 3$ (see Example 4.39).

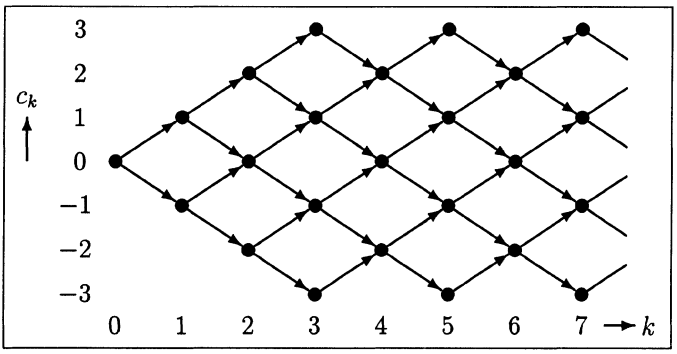


Figure 7.11: Unbalance trellis diagram of Miller-squared. The initial charge c_0 is assumed to be zero.

Every allowable DC-constrained sequence a_k corresponds to a path in the trellis of Fig. 7.11 and vice versa. Specifically, upgoing transitions in Fig. 7.11 have $a_k = +1$, while downgoing transitions have $a_k = -1$. Conversely, a_k is just the difference of successive charges ($a_k = c_k - c_{k-1}$). Charge c_k is even at all even instants k, and odd at all odd instants. The DC constraint causes the height of the trellis to be finite ($C_l \leq c_k \leq C_u$ for all k). The detector attempts to track the charge sequence c_k and derives bit decisions $\hat{a}_k = \hat{c}_k - \hat{c}_{k-1}$ from the estimated charge sequence $\hat{c}_k$. Due to noise, $\hat{c}_k$ may deviate for a while from c_k, but after some time it will bump against the 'ceiling' or 'floor' of the trellis and become realigned with c_k. The corresponding bit decisions $\hat{a}_k$ will differ in at least two positions from a_k, i.e. single bit-errors cannot occur.

This fact can be exploited in a full-response system for a performance improvement of 3 dB over bit-by detection of a_k (see Appendix 7A.1 for a more detailed analysis).

In a full-response system there is no ISI ($M = 0$), and the detector input z_k is just a noisy version of a_k. The trellis of Fig. 7.11 can serve as a basis for Viterbi detection because every branch of it uniquely determines a noiseless channel output a_k. The detector will need 3 states at even instants and 4 states at odd ones. It will not make single bit-errors as a bit-by-bit detector would. It will, however, occasionally suffer from double bit-errors that arise when a valid sequence a_k is confused with its closest valid 'neighbor' a'_k. Since a_k and a'_k must pass through adjacent nodes in Fig. 7.11, only two out of all possible states are of interest to the detection process at any given point in time. This paves the way to a simplified two-state detector that is due to Wood [72]. This detector is described in Appendix 7A.2.

Even with two states, detection is still more complicated than for uncoded data, where a simple bit-by-bit detector will do. For channel memory lengths $M \geq 1$, the trellis for the DC-free code remains more complicated than that for uncoded transmission; Appendix 7A.3 is concerned with the case $M = 1$.

7.5 CHARACTERISTICS OF VITERBI DETECTION

In the previous sections we have reformulated the detection of a transmitted data sequence a_k in Fig. 7.7 as the search of the shortest path through a trellis of states. This trellis is determined by the the transmission code and the memory length of the system up until the detector. Each branch in the trellis determines a noiseless detector input x_k along with a branch metric $\beta_k = (z_k - x_k)^2$ that can be interpreted as the 'length' of the branch. The total length of a path through the trellis is the sum of all branch metrics along that path. It should be noted that minimization of this path length is equivalent to maximum-likelihood detection if and only if noise n_k at the detector input is Gaussian and white (see Chapter 3). In other cases the approach may still work very well, but it is no longer entirely optimum. The effect of colored noise is reviewed in Appendix 3D (see also Appendix 7C).

The Viterbi detector searches the shortest path through the trellis via forward dynamic programming. To this end, it keeps track for any of the $|S|$ possible states $\mathbf{s}_k$ of a *surviving path* $\underline{p}(\mathbf{s}_k)$ that characterizes the most likely data sequence leading to this state. For each state it also keeps track of a *path metric* $\lambda(\mathbf{s}_k)$ that characterizes the length of the surviving path, i.e. the sum of all branch metrics $\beta_0, \ldots, \beta_k$ for this path. For any stage of the trellis (i.e. for any symbol interval k), a branch metric $\beta(\mathbf{s}_k, \mathbf{s}_{k+1})$ is computed for all B branches that connect the current and next states $\mathbf{s}_k$ and $\mathbf{s}_{k+1}$. Addition of $\beta(\mathbf{s}_k, \mathbf{s}_{k+1})$ to $\lambda(\mathbf{s}_k)$ yields the metric of the extended path that leads to $\mathbf{s}_{k+1}$ via $\mathbf{s}_k$. Of all extended paths that lead to $\mathbf{s}_{k+1}$, the one with smallest metric survives, and the other ones are discarded.

7.5.1 Path truncation

In principle, surviving paths should be retained from the beginning of transmission. In practice message lengths N are not strictly bounded, and the length of surviving paths would rapidly become unmanageable. For this reason, paths are normally truncated at a finite *truncation depth* D, i.e. surviving paths $\underline{p}(\mathbf{s}_k)$ are of the form $\underline{p}(\mathbf{s}_k) = [a_{k-\mathsf{D}}, \ldots, a_{k-M-1}]$. This yields a fixed detection delay of D symbol intervals. Truncation is allowable because surviving paths tend to merge somewhere in the near past (compare Fig. 7.3, where the surviving paths to Frankfurt and Görlitz merge in Braunschweig). When D is large enough, the $|S|$ digits $a_{k-\mathsf{D}}$ of the $|S|$ surviving paths $\underline{p}(\mathbf{s}_k)$ will most probably coincide, and any of them can be used as a near-maximum-likelihood estimate of $a_{k-\mathsf{D}}$. For small values of D, it becomes important to select the digit $a_{k-\mathsf{D}}$ from the surviving path with smallest path metric since this is the path with greatest current likelihood. Conceptually, the VD acts as a window of width D through which a portion of the trellis may be viewed. Every symbol interval k, the window slides one position to the left. New nodes appear on the right, and some paths are extended to them while other paths disappear. At the same time an old column of nodes on the left is shifted out of sight to serve as VD output.

Performance consequences of a finite detection delay D are assessed in Section 7.9 and Appendix 7B (see also [8]). Required values of D tend to grow with the amplitude distortion of the channel, and may be as large as 10 to 20 symbol intervals for uncoded transmission. Coding may reduce these figures but may also enlarge them.

Example 7.6 Fig. 7.12 depicts a transmission system with uncoded binary data a_k, a $1 + \alpha\mathcal{D}$ channel, and a Viterbi detector. For this system, the effect of a restricted detection delay D on bit error rates is analyzed in Appendix 7B.1.

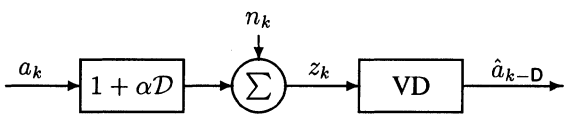


Figure 7.12: Transmission system with uncoded binary data a_k, $1+\alpha\mathcal{D}$ channel with memory length $M = 1$, and Viterbi detector with detection delay D.

The parameter α determines the amount of ISI as well as the amplitude distortion introduced by the channel. Both are small when $|\alpha|$ is small, and so is the required detection delay D. Conversely, a delay of some 15 bit intervals is needed as $|\alpha| \to 1$; this will ensure that losses with respect to the matched-filter bound remain negligible down to bit error rates of around 10^{-10}.

Example 7.7 Let us reconsider the channel of Example 7.6 for the situation that data a_k is run-length-limited with minimum-run-length parameter $d = 1$ (see Example 7.4 for an explanation of d). In Problem 7.10 it is shown that the simple receiver of Fig. 7.13 is optimum in this case whenever $\alpha \geq 0$ (see also Section 5.5).

The receiver is just a matched filter followed by a slicer. The detection delay D is only one symbol interval and is due to the matched filter. This illustrates that coding can induce substantial reductions of both detection delay and receiver complexity.

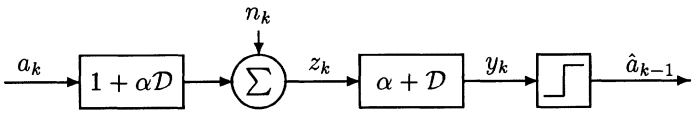


Figure 7.13: Optimum reception of binary data a_k with minimum-run-length parameter $d = 1$ across a $1 + \alpha\mathcal{D}$ channel with white Gaussian noise n_k is possible for $\alpha \geq 0$ with a very simple receiver.

Example 7.8 On top (or instead) of the d-constraint, run-length-limited codes often introduce a so-called k-constraint that characterizes the largest allowed separation between data transitions.[1] Runs of identical bits a_k may be at most $k + 1$ symbol intervals long. On a $1 - \mathcal{D}$ channel, the noiseless channel output $x_k = a_k - a_{k-1}$ becomes nonzero as a result of transitions, i.e. whenever $a_k \neq a_{k-1}$. The polarity of x_k then uniquely determines both a_k and a_{k-1}. Conversely, outputs $x_k = 0$ occur throughout a run of identical bits, irrespective of the bit polarity. Optimum decisions with respect to this polarity cannot be made before the run has ended with a transition. In uncoded transmission, this means that decisions may have to be deferred for 15 symbol intervals or more (see Example 7.6) so as to ensure with a sufficiently high probability that a transition has occurred. The use of a k-constraint, by contrast, ascertains that transitions occur within $k + 1$ symbol intervals from each other, and a detection delay $\mathsf{D} \simeq k$ suffices to ensure that all relevant information is exploited in the detection process. This insight is elaborated in [54] to develop a particularly simple $1 - \mathcal{D}$ detector for codes like MFM with $k = 3$. A more powerful yet somewhat more complicated detector that exploits both d and k constraints is developed in Appendix 7E (see also Section 7.7.2).

Example 7.9 In order to exploit charge constraints as in Example 7.5, the VD must trace the estimated charge sequence until it bumps into the 'ceiling' or 'floor' of the unbalance trellis. This may take quite a while, especially when the trellis has a large height. The detection delay should be large enough to permit the VD this degree of patience in making its final selection. Required delays may be in the order of hundreds of symbol intervals (see Section 7A.3, Fig. 7A.4). Substantial reductions of this delay are possible via codes in which the unbalance trellis diagram slims down periodically to one or only a few charge states (see Section 4.9.6).

7.5.2 VD implementation and complexity

VD implementation requires branch metric computation (BMC) units, add-compare-select (ACS) units which serve to compute and compare metrics of extended paths and perform a path selection on the basis of the outcome of the comparison, and a path memory which serves to store the surviving data paths.

Branch metric computation units These units transform detector inputs into branch metrics, based on knowledge of the system response. This knowledge can either be present *a priori* or acquired adaptively; the latter taste will be illustrated in Example 7.12 (see also Section 8.8). Since branch metric computation is an open-loop

[1]This k should not be confused with the time index, for which we use the subscript $_k$.

operation, a fixed *latency* (i.e. a fixed pure delay) is allowable. This permits pipelining and thereby facilitates high-speed implementations.

Add-compare-select units In essence, the ACS unit keeps track of the path metrics. This is a closed-loop process, and latencies must in principle be smaller than one symbol interval. It is this requirement that restricts attainable data rates. Techniques to break this bottleneck are described, for example, in [19], [60] and [37]. These are all linear-scale solutions, i.e. for a fixed hardware processing speed the attainable data rate will increase linearly with hardware complexity. Conversely, in cases where the throughput of a hardware ACS is high compared to the symbol rate, it is possible to multiplex the ACS across several trellis states for improved area-time efficiency [56, 57].

It should be noted that branch metrics β_k are all non-negative. A direct consequence is that path metrics λ_k are a non-increasing function of k, and may grow without bound (see [58] and [31] for numerical analyses). This is clearly impractical, both for analog and digital metric representations. There are several ways to overcome this problem [55]:

Periodic reset: At periodic intervals a sequence of *a priori* known digits a_k is inserted into the data stream so as to force the trellis to contract into a single state. Then all survivoring paths will merge, and all path metrics can be reset.

Variable shift: After some fixed number of iterations, the minimum survivor metric is subtacted from all metrics [62]. This is possible because of the fact that path selection is based on the minimum of all path metrics. Reduction of all path metrics by the same amount does not affect the outcome of the selection.

Difference metrics: The detection process is reformulated in terms of differences of path metrics, which vary in a well-defined range around zero. This approach is particularly simple and elegant for two-state Viterbi detectors (see Section 7.6.1), but may also work well with more states (see Appendix 7E).

Modulo normalization: Here use is made of the overflow properties of two's complement arithmetic in order to avoid any explicit metric modifications while retaining the possibility to identify the smallest metric at all times [30, 55].

Path memory Surviving paths are often stored in a set of cross-coupled shift registers, one for each path. Update of a path register requires a shift operation whenever a new state s_{k+1} coincides with the old state s_k; in all other cases a parallel load operation is needed. We will later describe this *register-exchange* process in more detail. In certain cases a random access memory (RAM) is used instead (see Section 7.6.3). Another possibility involves the use of a *traceback memory* in which the last $\mathsf{D} - M$ predecessors of every state are memorized. If D is sufficiently large then all state trajectories will merge within the span of the memory, and an estimate $\hat{s}_{k-\mathsf{D}+M}$ of the state at moment $k - \mathsf{D} + M$ can be formed by tracing the list of predecessors back in time, starting from an arbitrary state at moment k. The digit $a_{k-\mathsf{D}}$ corresponding to $\hat{s}_{k-\mathsf{D}+M}$ serves as the final decision of the VD. In comparison to the register-exchange path memory,

the traceback memory has smaller complexity at the cost of increased latency. An elegant mathematical description that encompasses both memories as well as attractive intermediate forms may be found in [20]. Further path register organizations are possible and may be advantageous, depending on the data rate and the number of states [52], [78], [21].

VD complexity The number of ACS units and path registers in the VD is determined by the number of states $|S|$ and may become very large (see Example 7.1). A similar remark applies to the number of branch metrics B that must be computed per symbol interval. Even though coding may temper the exponential growth of $|S|$ and B with L and M, for practical values $L \simeq 2 - 8$ and $M \simeq 5 - 20$, both $|S|$ and B tend to become very large. Thus, notwithstanding the fact that the VD is exponentially simpler than the 'brute-force' MLSD of Chapter 3, Viterbi detection may still be unduly complicated. This explains why much research effort has been directed in the past few decades towards simplifications. These normally lead to a deviation from maximum-likelihood detection, but in many instances substantial complexity reductions are possible at a minor expense to performance. Many of these simplifications are inspired heuristically (see [64, 12, 2] for a small sample). It is often hard to predict their performance consequences theoretically, and one must resort to simulations for a quantitative assessment. This leaves an uncertainty as to their merits in practical situations, where channels are likely to deviate from those that were covered by simulation. There are, fortunately, some systematic alternatives to these *ad hoc* simplifications whose performance can be assessed analytically. In the remainder of this chapter we elaborate four of these techniques. The first two, *channel memory truncation by means of prefiltering* [17] and *sequence feedback* [14], [16], [5], [11], are aimed at tempering the dependence of $|S|$ and B on M. The third, *set partitioning* [16], attacks the depence on L. The fourth technique, *restricted-delay detection*, restricts the detection delay D to a small but acceptable value.

7.6 CHANNEL MEMORY TRUNCATION BY MEANS OF PREFILTERING

Here the whitened matched filter which precedes the VD is replaced by a prefilter that curtails the memory length of the system into a manageable value M, say $M = 1$ or $M = 2$ [17]. This replacement inevitably causes the noise in the VD input signal to grow in power and to become colored. The first effect always degrades performance. The second one often does, but may also lead to an improvement. An example, originating in high-density magnetic recording [73], is shown in Fig. 7.14.

The symbol response $h(t)$ of the recording channel extends across many symbol intervals, and use of a whitened matched filter would necessitate a VD with prohibitively many states. Instead, a prefilter is used that transforms $h(t)$ into a *desired response* or *target response* whose sampled version is Class IV $(1 - \mathcal{D}^2)$. This is a proper choice:

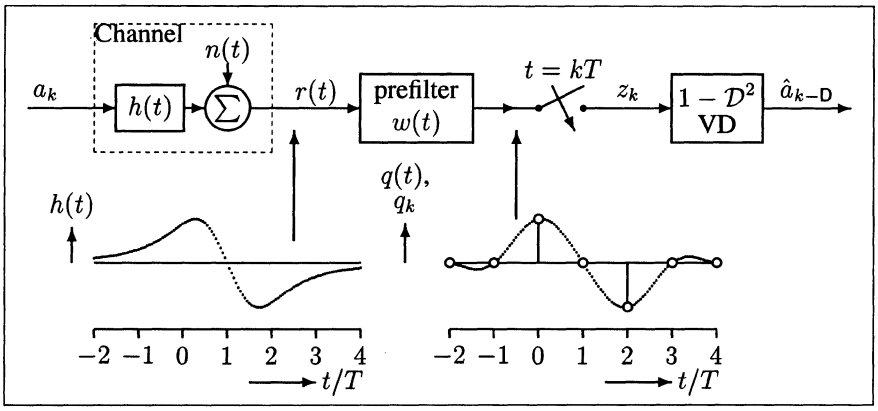


Figure 7.14: Partial-response maximum-likelihood (PRML) reception.

$h(t)$ resembles the desired response quite well, i.e. little equalization effort is required, and noise enhancement and coloration remain small. In return for this modest degradation, the equalized system response has a memory length M of only 2 symbol intervals. Thus a 4-state VD will do. This number can, in fact, be halved by exploiting the fact that the $1 - \mathcal{D}^2$ response contains no odd coefficients of $\mathcal{D}$.

Interleaved detection of Class IV In Fig. 7.15, the prefiltered and sampled sequence z_k of Fig. 7.14 is split into even and odd components $z_m^e \triangleq z_{2m}$ and $z_m^o \triangleq z_{2m+1}$ of data rate $1/(2T)$. Because of its Class IV structure, z_k is of the form $z_k = a_k - a_{k-2} + n_k$, where n_k is the noise component of z_k. Hence

$$z_m^e = a_{2m} - a_{2m-2} + n_{2m} = a_m^e - a_{m-1}^e + n_{2m}, \tag{7.5}$$

where $a_m^e \triangleq a_{2m}$ is a sequence of data rate $1/(2T)$ that contains the even bits of a_k. This shows that z_{2m} is a Bipolar $(1 - \mathcal{D})$ version of a_m^e and that it does not depend on the odd bits. A 2-state Bipolar VD operating on z_{2m} suffices for detection of a_m^e. A similar VD is needed for detection of the odd bits $a_m^o \triangleq a_{2m+1}$ from z_{2k+1}. VD outputs are interleaved into an estimate of a_k. Both detectors run at $1/(2T)$, i.e. half of the original data rate. This facilitates implementation at high data rates.

Partial-response Viterbi detection It is no coincidence that a partial response serves as target response in the above example and, in fact, in most applications of the prefiltering approach under consideration:

1. In Chapter 6 we have seen that partial-response equalization is well suited for use on channels with severe amplitude distortion, provided that a partial response is selected whose amplitude characteristics resemble those of the channel. Then noise will be hardly enhanced or colored, and performance may approach that

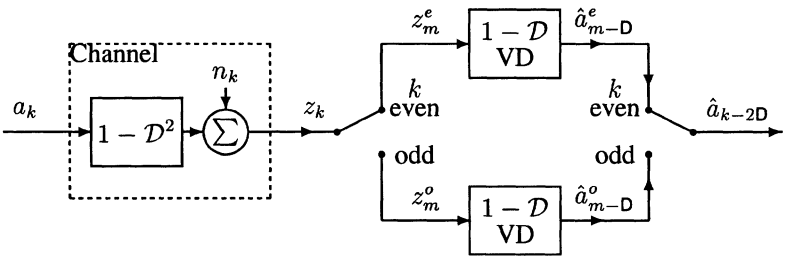


Figure 7.15: Interleaved detection of Class IV by means of two 2-state Viterbi detectors that run at half the original data rate.

of a full-fledged VD with whitened matched filter. It can be shown that partial responses are appropriate if not optimum target responses for this purpose [17, 25, 3]. To a first-order approximation, performances are around 3 dB better than those derived in Chapter 6 for partial-response equalization with bit-by-bit detection. The reason is that in the latter case trailing ISI is in effect canceled, i.e. half of all received data energy is neglected. The VD, on the other hand, exploits all received data energy. A more accurate performance analysis that accounts for the effect of colored noise is given in Appendix 7C. A noteworthy outcome of the analysis is that the Class IV VD is hardly sensitive to impairments near DC and the Nyquist frequency. This also applies to prefilter inaccuracies near these two band edges.

2. Partial responses all have a small memory length M. Typically only 2 VD states are required for binary data. Any 2-state VD may be simplified by using *difference metrics* (Section 7.6.1).

3. The two most important responses, viz. $1+\mathcal{D}$ and $1-\mathcal{D}$, have a very simple structure that can be exploited to simplify the VD further (Section 7.6.2).

7.6.1 Difference metrics

Let a_k be binary with $a_k \in \{-1, 1\}$ and let $M = 1$. Then the VD has two states $\mathbf{s}^-$ ($a_{k-1} = -1$) and $\mathbf{s}^+$ ($a_{k-1} = +1$) with associated path metrics λ_k^+ and λ_k^-, respectively. The corresponding trellis diagram is shown in Fig. 7.16. There are 4 branches $- \to -, \ldots, + \to +$, with metrics $\beta_k^{--}, \ldots, \beta_k^{++}$. The surviving path to $\mathbf{s}_{k+1}^-$ is determined by comparing the metrics of the competing paths via $\mathbf{s}_k^-$ and $\mathbf{s}_k^+$, i.e.

$$\lambda_{k+1}^- = \min(\lambda_k^- + \beta_k^{--}, \lambda_k^+ + \beta_k^{+-}). \tag{7.6}$$

A similar comparison occurs for state $\mathbf{s}_{k+1}^+$:

$$\lambda_{k+1}^+ = \min(\lambda_k^- + \beta_k^{-+}, \lambda_k^+ + \beta_k^{++}). \tag{7.7}$$

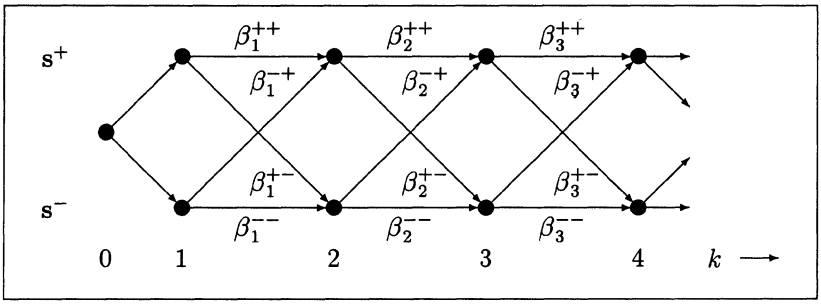


Figure 7.16: Two-state trellis.

What matters in these comparisons is which metric is largest. Thus only the *difference* between metrics is of concern. To exploit this fact we define the *difference path metric* $\Delta_k \triangleq \lambda_k^- - \lambda_k^+$, along with metric increments $Q_{k+1}^- \triangleq \lambda_{k+1}^- - \lambda_k^+$ and $Q_{k+1}^+ \triangleq \lambda_{k+1}^+ - \lambda_k^+$. It is easy to express (7.6) and (7.7) in terms of these quantities upon subtracting λ_k^+ from the left and right hand sides. The result is

$$Q_{k+1}^- = \min(\Delta_k + \beta_k^{--}, \beta_k^{+-}) \tag{7.8}$$

and

$$Q_{k+1}^+ = \min(\Delta_k + \beta_k^{-+}, \beta_k^{++}). \tag{7.9}$$

As expected, the absolute values of λ_k^+ and λ_k^- do not come into play. Furthermore, subtraction of both minima yields the new difference path metric:

$$Q_{k+1}^- - Q_{k+1}^+ = (\lambda_{k+1}^- - \lambda_k^+) - (\lambda_{k+1}^+ - \lambda_k^+) = \Delta_{k+1}. \tag{7.10}$$

In summary, based on the old difference path metric Δ_k and branch metrics $\beta_k^{--}, \ldots, \beta_k^{++}$, we determine survivors for time $k+1$ according to (7.8) and (7.9). These comparisons further yield minima Q_{k+1}^- and Q_{k+1}^+, whose difference determines the new difference path metric Δ_{k+1}. Thus the entire detection process is cast in terms of a single difference metric, as opposed to two metrics λ_k^+ and λ_k^+ in the standard VD. This simplifies the VD because less quantities need to be stored and updated. A further advantage is that Δ_k varies in a bounded range around 0. This avoids the complications that were mentioned in Section 7.5.2 for metrics such as λ_k^+ and λ_k^- that may grow without bound.

Exercise 7.1 Assume that $\Delta_k < 0$. Which of both states $\mathbf{s}_k$ is most likely? What is the bit a_{k-1} corresponding to this state? Show that a preliminary decision $\hat{a}_{k-1}$ with respect to a_{k-1} can be derived from Δ_k in a very simple manner.

7.6.2 Ferguson's algorithm

Our derivation so far has only exploited the fact that the VD has two states. Consequently, difference path metrics can be used for any two-state VD. A further simplific-

ation is possible for first-order partial-response channels by exploiting the simple structure of partial-response branch metrics. This leads us to Ferguson's algorithm [18, 73].

For the sake of simplicity we restrict attention to $1-\mathcal{D}$ (the $1+\mathcal{D}$ case is completely analogous). Then the VD has input $z_k = x_k + n_k$, where $x_k \triangleq a_k - a_{k-1}$ and n_k is additive noise. In the trellis diagram of Fig. 7.17, transitions between states are noted with the corresponding values of the noiseless channel output x.

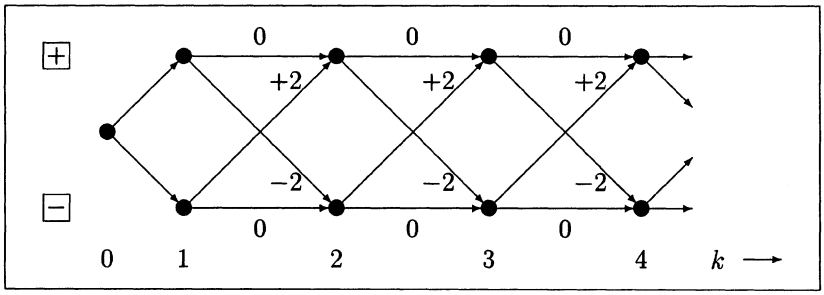


Figure 7.17: Bipolar $(1-\mathcal{D})$ trellis.

For the horizontal branches ($+ \rightarrow +$ and $- \rightarrow -$) a_{k-1} and a_k are equal, and $x = 0$. This yields branch metrics $\beta_k^{++} = \beta_k^{--} = (z_k - 0)^2 = z_k^2$. For the cross-over branches ($+ \rightarrow -$ and $- \rightarrow +$) we have $x = -2$ and $x = +2$, and branch metrics are $\beta_k^{+-} = (z_k + 2)^2$ and $\beta_k^{-+} = (z_k - 2)^2$. Thus (7.8) and (7.9) become

$$Q_{k+1}^{-} = \min(\Delta_k + z_k^2, (z_k + 2)^2) \tag{7.11}$$

and

$$Q_{k+1}^{+} = \min(\Delta_k + (z_k - 2)^2, z_k^2). \tag{7.12}$$

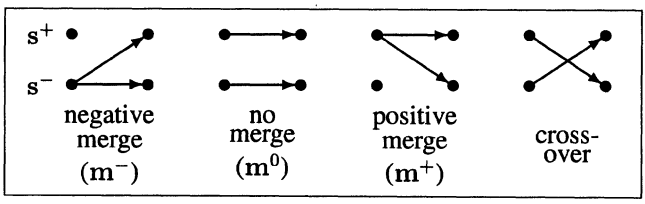


Figure 7.18: Path extensions come in 4 types, of which the cross-over is impossible for $1-\mathcal{D}$ channels.

We can distinguish four types of path extension (Fig. 7.18):

Negative merge (m^-): Here the $- \rightarrow -$ and $- \rightarrow +$ branches survive. From (7.11) and (7.12), this occurs when $\Delta_k + z_k^2 < (z_k + 2)^2$ and $\Delta_k + (z_k - 2)^2 < z_k^2$, i.e. when $\Delta_k < (z_k + 2)^2 - z_k^2 = 4z_k + 4$ and $\Delta_k < z_k^2 - (z_k - 2)^2 = 4z_k - 4$. The

latter condition is strongest and the former one does not come into play. The new difference metric amounts to

$$\Delta_{k+1} = Q^-_{k+1} - Q^+_{k+1} = (\Delta_k + z_k^2) - (\Delta_k + (z_k - 2)^2) = 4z_k - 4. \quad (7.13)$$

No merge ($\mathbf{m}^0$): Both horizontal branches survive when $\Delta_k + z_k^2 < (z_k + 2)^2$ and $z_k^2 < \Delta_k + (z_k - 2)^2$, i.e. when $\Delta_k < 4z_k + 4$ and $\Delta_k > 4z_k - 4$. The new difference metric is $\Delta_{k+1} = Q^-_{k+1} - Q^+_{k+1} = (\Delta_k + z_k^2) - (z_k^2) = \Delta_k$, i.e. Δ does not change.

Positive merge ($\mathbf{m}^+$:) By analogy to the negative merge, the $+ \rightarrow +$ and $+ \rightarrow -$ branches survive when $\Delta_k > 4z_k + 4$, and the new difference metric is $\Delta_{k+1} = 4z_k + 4$.

Cross-over: The regions of Δ_k for which $\mathbf{m}^-$, $\mathbf{m}^0$ and $\mathbf{m}^+$ occur together cover all possible values of Δ_k. Thus a cross-over cannot occur for the channel at hand.

It is obviously convenient to record and update $\Delta'_k \triangleq \Delta_k/4$ instead of Δ_k. The required actions are summarized in Table 7.2. An implementation is shown in Fig. 7.19.

Δ'_k	Δ'_{k+1}	Survivor update
$> z_k + 1$	$z_k + 1$	$\mathbf{m}^+$
$\in [z_k - 1, z_k + 1]$	Δ'_k	$\mathbf{m}^0$
$< z_k - 1$	$z_k - 1$	$\mathbf{m}^-$

Table 7.2: Ferguson's algorithm. Positive merge, no merge and negative merge are indicated by $\mathbf{m}^+$, $\mathbf{m}^0$ and $\mathbf{m}^-$, respectively.

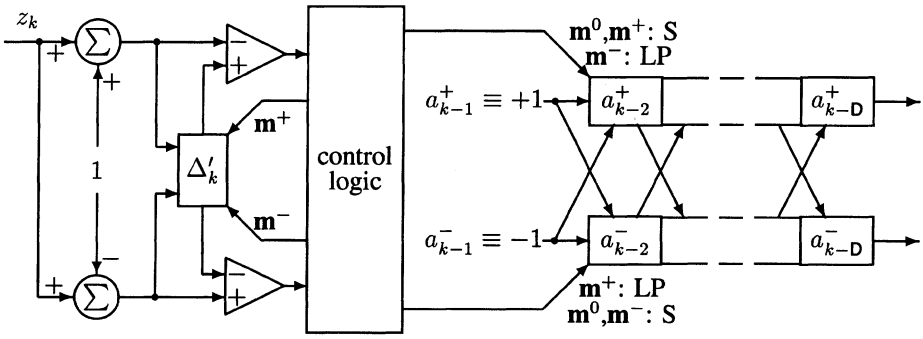


Figure 7.19: Possible implementation of $1 - \mathcal{D}$ Viterbi detector according to Ferguson's algorithm.

The path memory takes the form of cross-coupled shift registers which keep track of the surviving paths $\underline{p}^-_k = [a^-_{k-D}, \ldots, a^-_{k-2}]$ and $\underline{p}^+_k = [a^+_{k-D}, \ldots, a^+_{k-2}]$. The bits

$a_{k-1}^- = -1$ and $a_{k-1}^+ = +1$ that correspond to states $\mathbf{s}^-$ and $\mathbf{s}^+$ are hardwired. At any instant k, both registers perform either a shift (S) or Load Parallel (LP) operation under the control of the sequences $\mathbf{m}^+$, $\mathbf{m}^0$ and $\mathbf{m}^-$ that are formed by control logic. This logic distinguishes which of the three regions of Table 7.2 z_k is in. To this end, Δ'_k is compared with $z_k - 1$ and $z_k + 1$. In the event of a merge $(\mathbf{m}^+/\mathbf{m}^-)$, the new value Δ'_{k+1} is saturated at the lower or upper comparison value. Otherwise it remains unchanged. Provided that the detection delay D is large enough (e.g. D $\simeq$ 20), the oldest digits of both shift registers will virtually always coincide, and any of them can be used as a near-maximum-likelihood estimate $\hat{a}_{k-\mathsf{D}}$ of $a_{k-\mathsf{D}}$.

Exercise 7.2 Verify that the cross-coupled shift register of Fig. 7.19 will track the surviving paths $\underline{p}_k^-$ and $\underline{p}_k^+$ when controlled in the indicated manner by $\mathbf{m}^-$, $\mathbf{m}^0$ and $\mathbf{m}^+$.

Example 7.10 Figure 7.20 illustrates the operation of the Ferguson VD for an example input z. The actual transmitted data sequence is $a_k = \ldots + + - + + + - \ldots$; corresponding noiseless channel outputs x_k are indicated with closed circles. The detector input z_k is a noisy version of x_k ($z_k = x_k + n_k$); samples of z_k are indicated by open circles. These samples define an uncertainty band $[z-1, z+1]$ whose upper and lower edges are depicted in the form of continuous-time waveforms (dashed). The difference metric Δ' is constrained to remain within this band. In principle Δ' is not changed, i.e. the graph of Δ' (solid) is a horizontal line. However, whenever Δ' bumps into a band edge it must trace that edge until the edge changes direction. In that case there is a positive or negative merge $(\mathbf{m}^+/\mathbf{m}^-)$, depending on which edge Δ' has bumped into. Conversely, there is no merge $(\mathbf{m}^0)$ when Δ' moves horizontally toward the end of a symbol interval. Except for a rotation of 90 degrees, this process is like water (Δ') falling through a pipe that is bended in the shape of the received signal.

The bottom of the picture depicts the paths through the trellis that correspond to the sequence of merges at hand. Two paths are extended in parallel as long as there is no merge $(\mathbf{m}^0)$. A positive merge causes the path via $\mathbf{s}_k^+$ to survive to both $\mathbf{s}_{k+1}^+$ and $\mathbf{s}_{k+1}^-$. This means that all bits since the previous negative merge up until a_{k-1} are judged to be $+1$; the most recent bit a_k remains ambiguous until the next negative merge. A similar interpretation can be given for a negative merge. All bits are detected correctly in spite of the fact that noise n_k is so large that x_k falls outside the band on two occasions. A bit-by-bit detector would have been in error here.

7.6.3 Variant of Ferguson detector for precoded data

Partial-response systems often employ a precoder at the transmitting end of the system. This precoder converts the original bit sequence d_k into a binary data sequence a_k that is recorded or transmitted (see Section 6.2). For $1-\mathcal{D}$ the precoder is just a modulo-2 integrator, i.e. $d_k = 1$ indicates a transition of a_k ($a_k = -a_{k-1}$), while $d_k = -1$ indicates no transition ($a_k = a_{k-1}$). The values d_k may be determined directly from the sequence of merges in the VD. Specifically, no merge $(\mathbf{m}^0)$ means that both surviving paths are extended 'horizontally', i.e. without transition (see Fig. 7.20). Thus we can safely decide that $d_k = -1$, even though the polarity of the ongoing run of a_k's has not yet been decided upon. A positive merge $(\mathbf{m}^+)$ indicates that a sequence of no merges has ended. In this event the previous merge of opposite polarity $(\mathbf{m}^-)$ must have been a transition (compare Fig. 7.20). This means that $d_p = +1$, where p indicates the instant

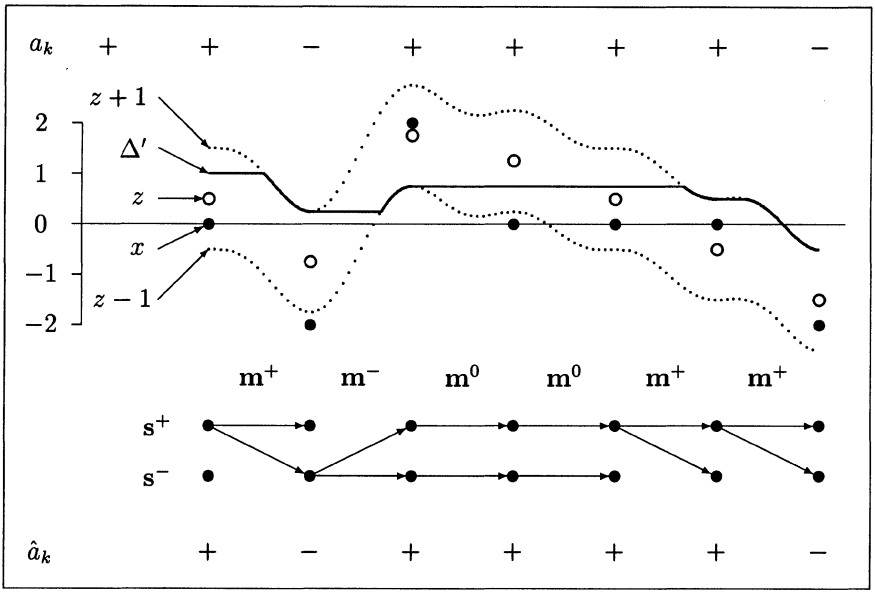


Figure 7.20: Illustration of the operation of Ferguson's Viterbi detector.

at which this merge has occurred. In the event of a negative merge, we can similarly decide that $d = -1$ at the instant p at which the previous positive merge occurred.

Exercise 7.3 Argue that the use of a precoder lowers bit error rates somewhat for small detection delays D. Does performance improve for large D?

The foregoing observations pave the way to an alternative version of Ferguson's detector [73] that is particularly suited for a digital implementation. Let $p \leq k-1$ be the previous instant at which there was a merge, and denote the polarity of this merge by $\varepsilon_p \in \{-1, 1\}$. Then by Table 7.2, $\Delta'_{p+1} = z_p + \varepsilon_p$. Between the instants p and $k-1$ there are no merges, i.e.

$$\Delta'_i = z_p + \varepsilon_p \text{ for } i \in \{p+1, \ldots, k\}. \tag{7.14}$$

Rather than keeping track of Δ'_k, we may store and update z_p and ε_p. From Table 7.2, a positive merge occurs when $\Delta'_k = z_k + \varepsilon_p > z_k + 1$, i.e. when $s_k \triangleq z_p - z_k > 1 - \varepsilon_p$. Apart from the current input z_k, this comparison indeed involves only z_p and ε_p. Because of the merge, these variables must be updated according to $z_p := z_k$ and $\varepsilon_p := 1$. The remaining two regions of Table 7.2 may be recast in a similar way. The resulting update algorithm is summarized in Table 7.3. In the implementation of Fig. 7.21, the sequence s_k is formed and compared with thresholds 0 and $-2\varepsilon_p \in \{-2, 2\}$. Multiplication of both comparator outputs yields a merge indicator ($\mathbf{m}$). In the event of a merge, the output ε_k of the left comparator is an indicator of the polarity of the merge, and is

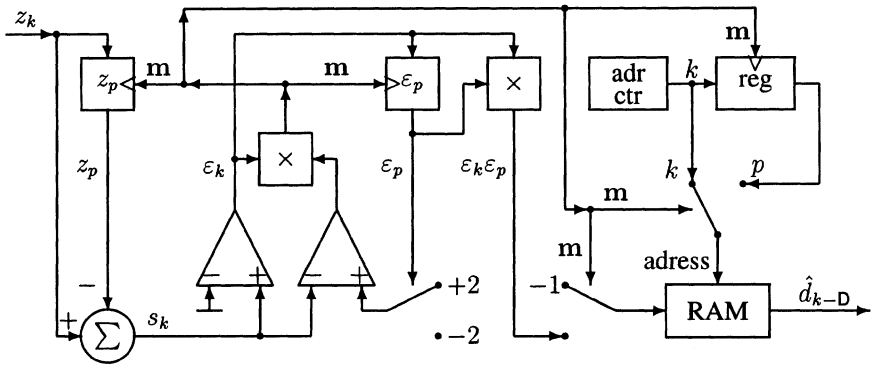


Figure 7.21: Alternative version of Ferguson detector according to [73] for detection of precoded data d_k on a $1 - \mathcal{D}$ channel.

latched into the ε_p register. At the same time z_k is latched into the z_p-register. The path memory now takes the form of a random access memory. The locations k and p of this register are updated in accordance with Table 7.3.

$s_k = z_p - z_k$	Trellis extension	VD output	Updates z_p	ε_p	p
$> 1 - \varepsilon_p$	m^+	$\hat{d}_p := -\varepsilon_p$	z_k	1	k
$\in [-1 - \varepsilon_p, 1 - \varepsilon_p]$	m^0	$\hat{d}_k = -1$	z_p	ε_p	p
$< -1 - \varepsilon_p$	m^-	$\hat{d}_p := \varepsilon_p$	z_k	-1	k

Table 7.3: Alternative version of Ferguson's algorithm according to [73] for detection of precoded data d_k on a $1 - \mathcal{D}$ channel.

Example 7.11 Fig. 7.22 illustrates the operation of the detector of Fig. 7.21 for the same example input z as in Fig. 7.20. The detector now keeps track of two decision thresholds that are at a fixed distance of each other, and are 'bended' around the actual received waveform. Positive and negative merges occur whenever the waveform 'pushes' the lower or upper threshold, respectively. Not surprisingly, the sequence of merges (and thereby the detected data sequence) is the same as in Fig. 7.20.

7.6.4 Extensions of the Ferguson detector

The simplicity of Ferguson's detector makes it attractive for applications at high data rates. A disadvantage is that only a single system response, namely $1 - \mathcal{D}$, can be handled. In practice this response may fluctuate somewhat. Even small mismatches between the actual response and the $1 - \mathcal{D}$ response assumed by the detector may lead to significant performance degradations. Gain variations, which are omnipresent in digital recording systems, are particularly cumbersome and must be accurately dealt with.

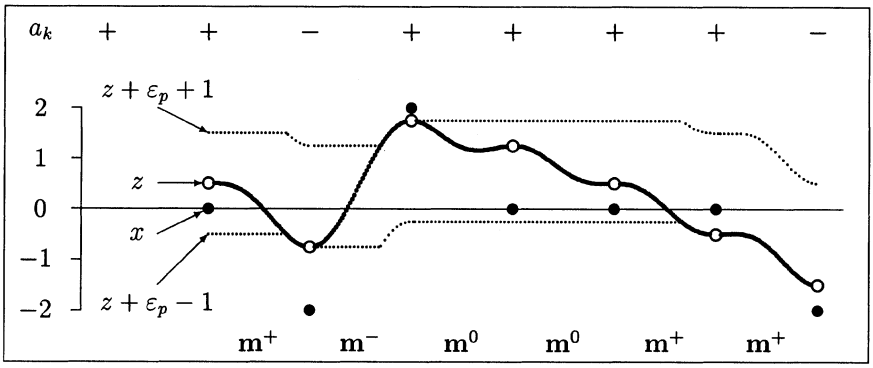


Figure 7.22: Illustration of the operation of alternative version of Ferguson detector according to [73] for detection of precoded data d_k on a $1 - \mathcal{D}$ channel.

This is usually done by means of a circuit for automatic gain control (AGC) that precedes the detector; possible AGC topologies are treated in Section 8.3. The alternative approach is to adapt the detection thresholds. To this end we must first generalize the detector to an $\alpha(1 - \mathcal{D})$ channel where α represents the actual gain of the channel and may deviate from the nominal value $\alpha = 1$. This generalization is straightforward and is left as an exercise for the reader. All that changes in Fig. 7.19 is that the constant 1 which is added to and subtracted from z_k is replaced by α, while in Fig. 7.21 the thresholds $+2$ and -2 are replaced by $+2\alpha$ and -2α, respectively. Problem 7.15 and Section 8.8.1 are concerned with adaptation of the estimate of α that the detector keeps track of. Problem 7.16 develops extensions of the Ferguson detector to arbitrary channels of memory length $M = 1$.

Apart from the detectors derived here, some other VD's exist that have been simplified specifically for partial-response channels. A notable and instructive example is the detector due to Forney [23, pp. 373-375]. None of these alternatives is as simple as Ferguson's detector.

7.7 SEQUENCE FEEDBACK

This technique involves the use of preliminary decisions to cancel the tail of the system response q_k up until the detector. What remains is a much shorter truncated response q_k^t that can be handled with a modest number of VD states. Fig. 7.23 illustrates the concept. Unlike the previous approach, the prefilter is not meant for pulse shortening. Instead, it performs phase equalization in order to concentrate pulse energy as much as possible near the beginning of the pulse. Ideally, only a small fraction of the total energy is left in the tail. As a result, tail cancellation causes only a small loss of data power and performance. In Fig. 7.23, the truncated system response q_k^t (indicated by open circles) has memory length $M' = 2$. Cancellation of the tail $q_{M'+1}, \ldots, q_M$ involves a feedback

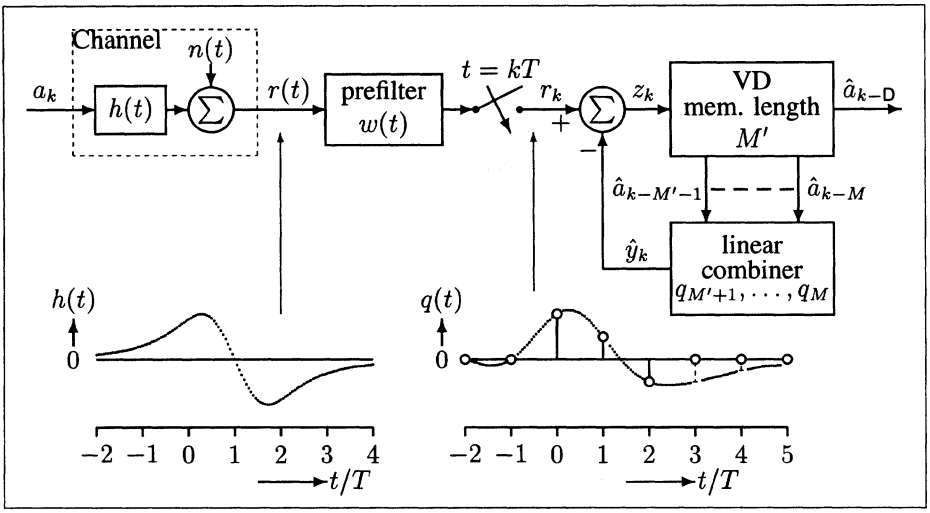


Figure 7.23: Simplification of the VD by means of sequence feedback.

operation akin to that in decision feedback equalization. A compensation sequence

$$\hat{y}_k \triangleq \sum_{i=M'+1}^{M} \hat{a}_{k-i} q_i \tag{7.15}$$

is formed on the basis of decisions $\hat{a}_{k-M'-1} \ldots \hat{a}_{k-M}$ produced by the VD. These decisions are applied to a linear combiner with weights $q_{M'+1}, \ldots, q_M$, whose output $\hat{y}_k$ ideally cancels the tail-component $y_k = \sum_{i=M'+1}^{M} a_{k-i} q_i$ of the sampler output r_k. The decisions entering the linear combiner cannot be derived from the VD output $\hat{a}_{k-\mathsf{D}}$ because the detection delay D will normally exceed $M' + 1$. This would necessitate an unphysical *negative* delay to form $\hat{y}_k$. To overcome this problem, *preliminary* decisions with a sufficiently small detection delay may be extracted from within the VD. These are taken from the surviving data path $\underline{p}(\mathrm{s}_k)$ with greatest likelihood (i.e. with smallest smallest path metric $\lambda(\mathrm{s}_k)$ among all states $\mathrm{s}_k \in S$).

By varying M' between 0 and M, performance can be traded against complexity. For $M' = M$ the feedback operation vanishes and the full-fledged VD emerges. Conversely, for $M' = 0$ only a single state remains and all trailing ISI is canceled, as in the the decision feedback equalizer (DFE) of Chapter 6. Thus the DFE is in fact a "poor man's VD".

Replacement of the linear combiner in Fig. 7.23 by a look-up table (ROM or RAM) enables nonlinearities to be handled. In Example 7.12 we illustrate an adaptive branch metric computation unit that exploits this possibility.

Relative to its full-fledged counterpart, the VD incurs performance degradations because it cancels some of the incoming data power and because of possible error propagation. Both degradations tend to be small.

Loss of data power Some of the received data power is canceled and hence not exploited in the detection process. To minimize this loss, the system response q_k should have minimum phase. This will ensure for any value of M' that pulse energy is concentrated as much as possible near the beginning of the pulse (see Section 1.6). Required values of M' tend to be small when q_k has minimum phase (see Section 7.7.3).

In Section 3.10.2 we have seen that a prefilter in the form of a canonical whitened matched filter (WMF) causes q_k to have minimum phase. This filter further causes noise n_k at the VD input to be white, as assumed in the derivation of the VD. We infer that the canonical WMF is the optimum prefilter irrespective of the truncation length M'. A proof of this fact may be found in [5].

Error propagation Erroneous decisions $\hat{a}_{k-M'-1} \ldots \hat{a}_{k-M}$ propagate via the linear combiner and may induce further decision errors. Error propagation decreases when q_k gets closer to minimum phase and when less feedback is used, i.e. when M' grows. Even for $M' = 0$ (i.e. for the DFE) it is typically not a major problem. Nevertheless, an improvement is possible by resorting to *local feedback.*

7.7.1 Local versus global feedback

In the receiver of Fig. 7.23, a single compensation sequence $\hat{y}_k$ is fed back and affects the updates of *all* state variables within the VD. Erroneous compensation causes all of these updates to become incorrect. This may be avoided by establishing a *local* feedback path for each state [33, 14, 16, 5, 11]. To this end, a compensation sequence

$$y_k^l \triangleq \sum_{i=M'+1}^{M} a_{k-i}^l q_i \tag{7.16}$$

is formed for each state $\mathbf{s}_k^l$, $l \in \{0, 1, \ldots, |S| - 1\}$, on the basis of the local data path $\underline{p}(\mathbf{s}_k^l) = [a_{k-\mathsf{D}}^l, \ldots, a_{k-M'-1}^l]$. This compensation sequence is subtracted from r_k to obtain a local version $z_k^l \triangleq r_k - y_k^l$ of z_k that underlies the branch metrics of all branches that emanate from state $\mathbf{s}_k^l$. Thus errors no longer automatically propagate to other states. This advantage becomes significant when the channel has severe amplitude distortion and M' is small [5]. The VD with local sequence feedback is known in the literature under various other names, including *decision feedback sequence estimator* (DFSE) [16] and *delayed decision feedback sequence estimator* (DDFSE) [14].

Instead of a single weighing operation as in Fig. 7.23, local feedback involves multiple weighing operations, viz. one for each state. In return for this modest overhead, local feedback yields a more homogeneous detector structure and goes without the need of first making a decision as to which survivor is most likely before the precise feedback sequence can be determined. This reduces latency, increases parallelism, and facilitates high-speed implementations.

Branch metric computation By way of illustration we consider local feedback for the situation of Fig. 7.23 and uncoded transmission. Here $M' = 2$, i.e. the trellis diagram has the structure of Fig. 7.9. There are four states $\mathrm{s}^0, \ldots, \mathrm{s}^3$, with two branches emanating from every state. Each state s_k^l ($l \in \{0, \ldots, 3\}$) uniquely determines a pair of bits (a_{k-1}^l, a_{k-2}^l); every branch determines a triplet $(a_k^i, a_{k-1}^l, a_{k-2}^l)$, where $i \in \{0, 1\}$ is a binary index that defines the newly appended bit. This triplet, in turn, determines a noiseless output $x_k^t = a_k^i q_0 + a_{k-1}^l q_1 + a_{k-2}^l q_2$ of the channel with truncated symbol response q_k^t. Branch metrics are of the form

$$\beta_k = (z_k - x_k^t)^2 = (r_k - \hat{y}_k^l - x_k^t)^2 = (r_k - x_k)^2 \tag{7.17}$$

where $x_k \triangleq x_k^t + \hat{y}_k^l$. Upon invoking (7.16) we find that

$$x_k = a_k^i q_0 + \sum_{i=1}^{M} a_{k-i}^l q_i. \tag{7.18}$$

We may apparently view x_k as a noiseless channel output for the original channel with non-truncated symbol response q_k. This output is determined in part by the newly appended bit a_k^i, for another part by the state s_k^l (a_{k-1}^l, a_{k-2}^l), and for the remainder by local feedback from the surviving path $\underline{p}(\mathrm{s}_k^l)$ $(a_{k-M'+1}^l, \ldots, a_{k-M}^l)$. Just as in the VD without feedback, the branch metric β_k is a measure of the difference of the actual channel output r_k and this noiseless one. Fig. 7.24 shows how a single branch metric may be computed.

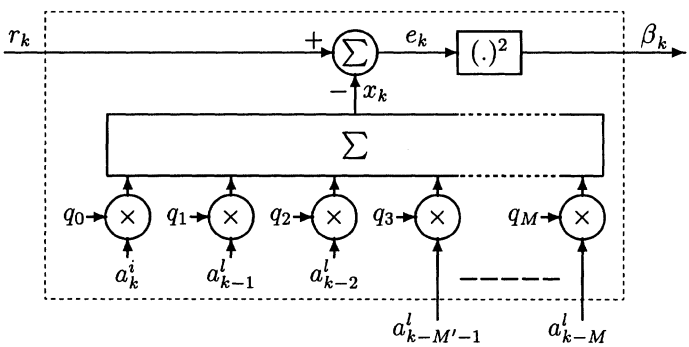


Figure 7.24: Branch metric computation unit for VD with local sequence feedback.

There are 8 branch metric computation units, one for each triplet $(a_k^i, a_{k-1}^l, a_{k-2}^l)$. These three bits are hardwired. The remaining bits $a_{k-M'-1}^l, \ldots, a_{k-M}^l$ that are needed to compute a noiseless channel output x_k are fed back from the local surviving path $\underline{p}(\mathrm{s}_k^l)$. The coefficients $q_0 \ldots q_M$ may be determined adaptively along the lines of Section 8.8. Further considerations in this context may be found in [36]. Adaptivity comes particularly easily if the linear weighing network in Fig. 7.24 is replaced by a look-up table. The following example, based on [6], serves to illustrate this.

Example 7.12 At high information densities, digital recording systems are often hampered by nonlinear ISI. This disturbance comes on top of linear ISI and noise, and causes the prefilter output r_k to be of the form $r_k = x(a_k, \ldots, a_{k-M}) + n_k$ where n_k is noise and x is a deterministic nonlinear function of $(a_k, \ldots, a_{k-M})$. This function is typically *a priori* unknown as a result of parameter variations. An estimate $\tilde{x}$ of x may be formed adaptively as shown in Fig. 7.25. The function $\tilde{x}$ is stored in a table (RAM) that is addressed by the bits $a^l_{k-M'-1}, \ldots, a^l_{k-M}$. The first $M'+1$ bits $(a^i_k, a^l_{k-1}, \ldots, a^l_{k-M'})$ are fixed for each computation unit and are hence omitted from the address space.

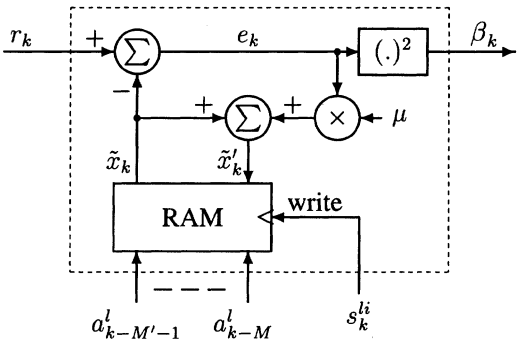


Figure 7.25: Adaptive branch metric computation unit for VD with local sequence feedback and ability to handle nonlinear ISI.

There are $2^{M'+1}$ units, one for each possible sequence $(a^i_k, a^l_{k-1}, \ldots, a^l_{k-M'})$, i.e. one for each branch in the trellis. Each branch defines an extended path of which the VD determines the metric $\lambda_k + \beta_k$. The smallest of these $2^{M'+1}$ metrics defines the triplet with greatest current likelihood. The corresponding indicator sequence s^{li}_k in Fig. 7.25 is made active whenever the triplet at hand is most likely. In that case all bits that determine $\tilde{x}$ are expected to be correct, and the error $e = r - \tilde{x} = x - \tilde{x} + n$ should consist only of noise n_k if $\tilde{x} = x$. A positive error suggests that $\tilde{x} < x$, i.e. that $\tilde{x}$ should be increased. To this end, e is scaled by a small positive factor μ and added to $\tilde{x}$. The update $\tilde{x}'$ is stored instead of $\tilde{x}$ in the RAM. One easily verifies that this recipe remains valid when $e < 0$. The adaptive look-up table (LUT) thus constructed will gradually evolve to contain a good estimate of the channel input-output function x. A more detailed discussion of LUT adaptation is given in Section 8.8.3. Problems 7.19 and 7.20 dwell further on adaptive branch metric computation.

7.7.2 Sequence feedback for coded transmission

Sequence feedback is equally applicable to coded data as it is to uncoded data. By way of example we consider MFM transmission across a $1-\mathcal{D}$ channel. MFM is a $(1,3)$ run-length-limited code, i.e. data transitions are separated by at least 2 and at most 4 identical bits. At any stage k, there is either a transition $(a_k \neq a_{k-1})$ or the length of an ongoing run increases by one $(a_k = a_{k-1})$. This is reflected in the *run-length trellis diagram* of Fig. 7.26.

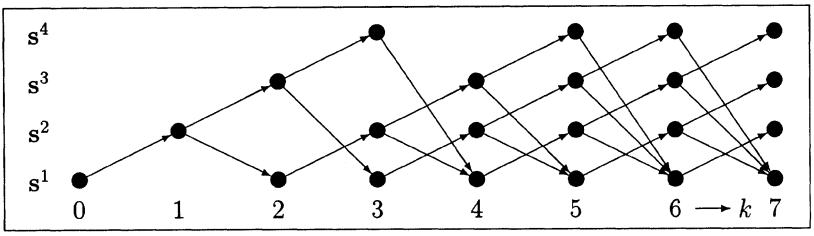


Figure 7.26: Run-length trellis diagram of MFM.

The diagram has four states $s^1, \ldots, s^4$ whose labels indicate the length of an ongoing run. Upgoing branches correspond to extension of an ongoing run by one bit, while downgoing branches correspond to a transition. For a $1 - \mathcal{D}$ channel, this means that noiseless channel outputs $x_k = a_k - a_{k-1}$ are zero for upgoing branches and $-2a_{k-1} \in \{-2, 2\}$ for downgoing ones. Unlike uncoded transmission, the latter outputs are not fixed *a priori* for each branch. Instead, they can assume one of two values, depending on a_{k-1}. To be able to determine x_k and the metric $\beta = (z_k - x_k)^2$ for the downgoing branch that emanates from state s_k^i, it is necessary to extract the most recent bit a_{k-1}^i of the surviving path $\underline{p}_k^i$ for this state. This is a form of local sequence feedback (see Section 7.7.1).

We have seen in Section 7.6.2 that the main problem in $1 - \mathcal{D}$ detection is one of pinpointing the *location* of transitions. In the situation at hand, this is done in a careful way: all possible locations are considered ($s^1, \ldots, s^4$), and path metrics keep accurate track of their likelihoods. Conversely, it is unlikely that confusion will arise with respect to the *direction* of a transition. This means for any possible location of a transition (i.e. for any state $s^1, \ldots, s^4$) that the digit a_{k-1}^i will be a good indicator of the direction of that transition. As a result, performance is essentially as good as that of a detector that keeps track of *both* possible directions of any transition. Such a detector would be based on a more complicated trellis with twice as many states.

Detector implementation is possible with 4 cross-coupled shift registers of length D, 4 path metric registers, and metric renormalization hardware that serves to prevent path metric overflow. In Appendix 7E it is shown that substantial simplifications are possible by exploiting the special structure of the trellis of Fig. 7.26. A related detector for $1 - \mathcal{D}$ channels with a delay of 3 symbol intervals is described in [54]. This detector is even simpler than the one of Appendix 7E. Performance for MFM is somewhat poorer because only the maximum run-length constraint is exploited (see Fig. 7E.3). Problem 7.32 is concerned with extensions of these approaches to $1 - \mathcal{D}^2$ channels.

7.7.3 Performance of the VD with sequence feedback

The possible occurrence of error propagation precludes an exact performance analysis of the VD with sequence feedback. Fortunately, error propagation is often negligible, particularly when local feedback is used. This justifies an approximate analysis

which neglects this phenomenon. For simplicity we restrict attention to the event that noise n_k at the VD input is Gaussian and white. The variance of n_k is denoted σ^2.

Let $\underline{a}^1$ and $\underline{a}^2$ be two admissible data sequences. Then the difference $\underline{e} \triangleq (\underline{a}^1 - \underline{a}^2)/2$ is said to be an *admissible error sequence*. Let E^N be the set of all admissible error sequences that start at $k = 0$ and have length N. The *partial Euclidian weight* $d^2(\underline{e})$ of any sequence $\underline{e} \in E^N$ is defined as

$$d^2(\underline{e}) \triangleq \sum_{k=0}^{N+M'-1} (e * q)_k^2. \tag{7.19}$$

Since q_k has memory length M, $(e * q)_k$ extends from $k = 0$ up until $k = N + M - 1$. Thus $d^2(\underline{e})$ amounts to the full Euclidian weight of $\underline{e}$ only when $M' = M$, i.e. for the full-fledged VD. The smallest of all possible partial weights is denoted $d_{M'}^2$, i.e.

$$d_{M'}^2 = \min_{N=1,2,\ldots} \min_{E^N} d^2(\underline{e}). \tag{7.20}$$

Clearly $d_M^2 = d_{\min}^2$, where $d_{\min}^2$ characterizes the performance of the full-fledged maximum-likelihood sequence detector (see Section 3.11). Also $d_{M'}^2 \leq d_M^2$ for all $M' \leq M$. Not surprisingly, $d_{M'}^2$ characterizes the performance of the VD with sequence feedback. More specifically, for sufficiently high signal-to-noise ratios it is demonstrated in Appendix 7D that this VD achieves a predetection signal-to-noise ratio $\rho_{\text{DFSE}} = d_{M'}^2/\sigma^2$.

The limiting case $M' = 0$ corresponds to the DFE. Here $d^2(\underline{e}) = q_0^2$ for the single bit-error $e_k = \delta_k$. By comparison, error sequences $\underline{e}$ of length $N \geq 2$ have weight

$$d^2(\underline{e}) = \sum_{k=0}^{N-1} (e * q)_k^2 \geq q_0^2 + \sum_{k=1}^{N-1} (e * q)_k^2 \geq q_0^2 \tag{7.21}$$

It follows that single bit-errors prevail, and that $d_0^2 = q_0^2$ irrespective of $q_1, q_2, \ldots, q_M$.

For $M' > 0$, ρ_{DFSE} continues to be governed by single bit-errors provided that ISI is not too severe. In that case

$$d_{M'}^2 = \sum_{k=0}^{M'} q_k^2, \tag{7.22}$$

i.e. the VD indeed exploits all energy of the truncated system response q_k^t. When q_k has minimum phase, performances will approach those of the full-fledged VD even when $M' \ll M$.

Example 7.13 Let $q_k = 2^{-k}$ for $k \geq 0$ and 0 for $k < 0$. For this infinitely long minimum-phase response, single bit-errors prevail for any $M' \geq 0$. The full-fledged VD exploits an energy

$$d_\infty^2 \triangleq \sum_{k=0}^{\infty} q_k^2 = \sum_{k=0}^{\infty} 4^{-k} = \frac{4}{3} \simeq 1.33. \tag{7.23}$$

For $M' = 0$ and 1 the truncated versions of q_k have energy 1 and 1.25, respectively. This corresponds to losses of 1.23 and 0.23 dB with respect to d_∞^2. Beyond $M' = 1$ losses are altogether negligible. Even the DFE ($M' = 0$) does well.

Effect of phase characteristics Chapter 3 has established that the performance of the full-fledged VD is independent of the phase characteristics of the system response. This is no longer true for the VD with sequence feedback.

Example 7.14 The systems with $(q_0, q_1, q_2) = (4,4,1)$, $(2,5,2)$ and $(1,4,4)$ have memory length $M = 2$ and minimum, linear and maximum phase, respectively (Fig. 7.27). Their amplitude characteristics $|Q(e^{j2\pi\Omega})| = 5 + 4\cos 2\pi\Omega$ are identical and reflect rather severe ISI (Fig. 7.28). (We recall that Ω is a normalized measure of frequency, with $\Omega = 1$ corresponding to the symbol rate $1/T$.)

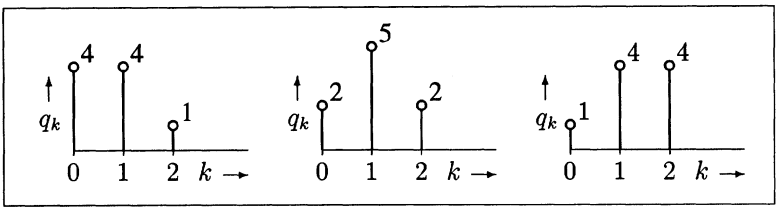


Figure 7.27: Three system responses with memory length $M = 2$ and identical amplitude characteristics.

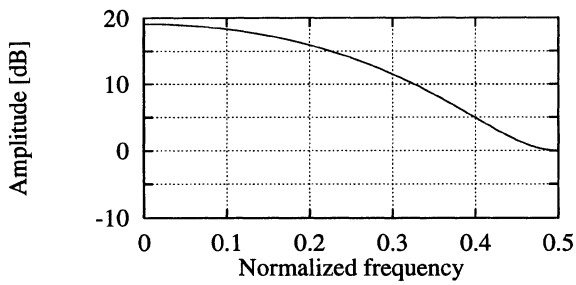


Figure 7.28: Amplitude characteristics of the channels of Fig. 7.27.

For the linear-phase channel and $M' = 1$, the single bit-error $e_k = \delta_k$ has partial Euclidian weight $d^2(\underline{e}) = (q_0^2 + q_1^2) = 29$. The double bit-error $e_k = \delta_k - \delta_{k-1}$ is also admissible, has weight $d^2(\underline{e}) = \sum_{k=0}^{2}(q_k - q_{k-1})^2 = 23$, and generates d_1^2 because no other admissible error sequence has smaller weight. The other minimum weights $d_{M'}^2$ may be determined analogously. For $M' = 1$ and 2 all are generated by $e_k = \delta_k - \delta_{k-1}$, while single bit-errors prevail for $M' = 0$. For $M' = 0$ and 1, minimum phase is clearly preferable (Table 7.4). For the full-fledged VD (i.e. $M' = M = 2$) the phase characteristics are indeed of no concern.

7.8 SET PARTITIONING

Prefiltering and sequence feedback are effective methods to temper or halt the rapid growth of the number of states $|S|$ as a function of the channel memory length M. They

q_0, q_1, q_2	d_0^2	d_1^2	$d_2^2 = d_{\min}^2$	d_{MFB}^2
$4, 4, 1$	16	25	26	33
$2, 5, 2$	4	23	26	33
$1, 4, 4$	1	10	26	33

Table 7.4: Performance comparison for the channels of Fig. 7.27.

cannot, however, control the explosive growth of $|S|$ with the number of data levels L. In this section we present a systematic approach to temper the latter growth. It is based on concepts of *set partitioning* that are best explained with the help of an example. To this end we return to the 4-level uncoded system with $M = 1$ of Example 7.2. Here we have 4 states $\mathrm{s}_k^0, \ldots, \mathrm{s}_k^3$ ($a_{k-1} = -3, \ldots, 3$), and 16 branches $(\mathrm{s}_k^l, \mathrm{s}_{k+1}^m)$ with branch metrics $\beta(\mathrm{s}_k^l, \mathrm{s}_{k+1}^m)$, $l, m \in \{0, 1, 2, 3\}$.

Let us consider the 4 extended paths that lead to state s_{k+1}^m. Competition among them is based on their metrics $\lambda_k^0 + \beta(\mathrm{s}_k^0, \mathrm{s}_{k+1}^m), \ldots, \lambda_k^3 + \beta(\mathrm{s}_k^3, \mathrm{s}_{k+1}^m)$. The smallest of these determines which extended path survives. Noise n_k in the VD input sequence r_k affects the branch metrics and may cause an erroneous selection. Two cases may be distinguished:

1. The selected path has a digit a_{k-1} that is adjacent to the one that should have been selected, e.g. -1 has been confused with either -3 or $+1$. This is rather likely because neighboring data levels induce branch metrics that will not differ very much, and a relatively small amount of noise may affect these metrics enough to cause confusion.

2. The digit a_{k-1} is further apart from the one that should have been selected, e.g. -1 has been confused with $+3$. This is quite unlikely because the corresponding branch metrics will differ significantly, even when distorted by noise.

Thus it would seem unnecessary and in fact overdone to spend the power and complexity of Viterbi detection to discriminations within the second category. This heuristic reasoning, which can be recast in a more rigorous form, motivates a partitioning of the data alphabet into 2 subsets $\Phi \triangleq \{-3, +1\}$ and $\Psi \triangleq \{-1, +3\}$. Discrimination within each subset is easy because data levels are spaced far apart, and it suffices to take instantaneous decisions. Conversely, discrimination between subsets is difficult because the intraset distance is only 2, and for this task Viterbi detection remains needed. This leads to the reduced trellis diagram of Fig. 7.29. This diagram has two states s^ϕ ($\Leftrightarrow a_{k-1} \in \Phi$) and s^ψ ($\Leftrightarrow a_{k-1} \in \Psi$) and corresponding path metrics λ_k^ϕ and λ_k^ψ. To get to s_{k+1}^ψ from s_k^ψ there are two possibilities, viz. extension by either of the symbols $a_k = -1$ and $a_k = 3$ that belong to Ψ. Similarly, each of the other 3 branches in the trellis diagram also has 2 parallel transitions. The emergence of parallel transitions is characteristic for set partitioning and enables instantaneous decisions to be taken. For example, the extended paths with $a_k = -1$ and $a_k = +3$ that pass through s_k^ψ have metrics $\lambda_k^\psi + (r_k - x(\mathrm{s}_k^\psi, -1))^2$ and $\lambda_k^\psi + (r_k - x(\mathrm{s}_k^\psi, 3))^2$. Since

λ_k^ψ is common to both metrics it does not affect the selection between both paths. All that matters is $x(\mathbf{s}_k^\psi, 3)$ and $x(\mathbf{s}_k^\psi, -1)$, and in particular which of the two is closest to r_k. When the channel is linear this instantaneous decision can be taken without explicit path metric computations via a single slicing operation.

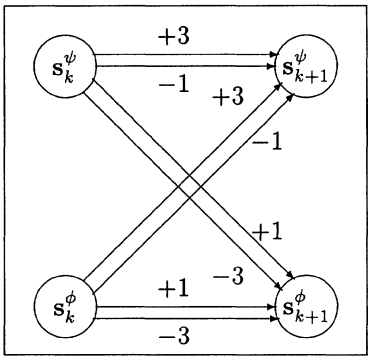


Figure 7.29: Segment of two-state trellis diagram for Viterbi detection of quaternary data by means of set partitioning. Noted along each transition is the corresponding value of a_k.

Exercise 7.4 Assume that the channel has impulse response (q_0, q_1). Express the required threshold level of the slicer in terms of a_{k-1}, q_0 and q_1.

Similar instantaneous decisions are taken for the remaining 3 branches. At this point we are left with a 2-state trellis segment without parallel transitions, for which detection proceeds in the regular fashion.

A comparison of Figs. 7.8 and 7.29 reveals that set partitioning has halved the number of states and transitions. Since only two states are left, difference metrics come into play naturally (see Section 7.6.1). Half of all selections is among parallel transitions and as such comparatively simple. For these reasons, set partitioning has more than halved VD complexity. Performance, on the other hand, has typically not degraded.

Example 7.15 Consider a linear system with symbol response $(1, \alpha)$. Within both subsets Φ and Ψ, data levels have a spacing $\Delta = 4$. Noise n_k must have an amplitude of at least $\frac{\Delta}{2} = 2$ to cause confusion between two symbols within the same subset. This corresponds to a Euclidian distance $d^2_{\min}(\text{intraset}) = 4$. By comparison, decisions between sets are not instantaneous and involve the full power of the VD. However, since the minimum spacing of data levels in different sets is only 2, the error sequences e_k that govern the performance of the VD may have amplitudes as small as 1, just as for binary transmission. The minimum interset Euclidian distance $d^2_{\min}(\text{interset}) = 1^2 + \alpha^2$ is generated by the single bit error $e_k = \delta_k$. Hence $d^2_{\min}(\text{intraset}) \geq d^2_{\min}(\text{interset})$ whenever $|\alpha| < \sqrt{3}$. This covers all minimum-phase channels (which are characterized by $|\alpha| < 1$). Conversely, a full-fledged VD remains needed when $|\alpha| > \sqrt{3}$.

Possible set partitionings for 4-level data are depicted in Fig. 7.30. Partitionings of this type are sometimes called Ungerboeck partitionings in recognition of their inventor

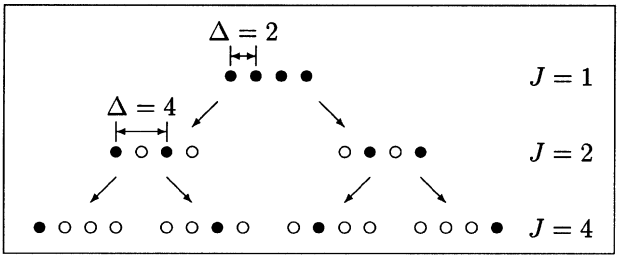


Figure 7.30: Ungerboeck partitionings for 4-level data.

[63]. The one-way partitioning of Fig. 7.30 ($J = 1$) yields a single set with 4 data symbols of spacing $\Delta = 2$. Decisions between these symbols are instantaneous, just as in the DFE. The four-way partitioning ($J = 4$) yields four sets with one data symbol each. This corresponds to the full-fledged VD. The two-way partitioning ($J = 2$) yields two sets, each with two data symbols of spacing $\Delta = 4$. This corresponds to the two-state VD with set partitioning that was described above.

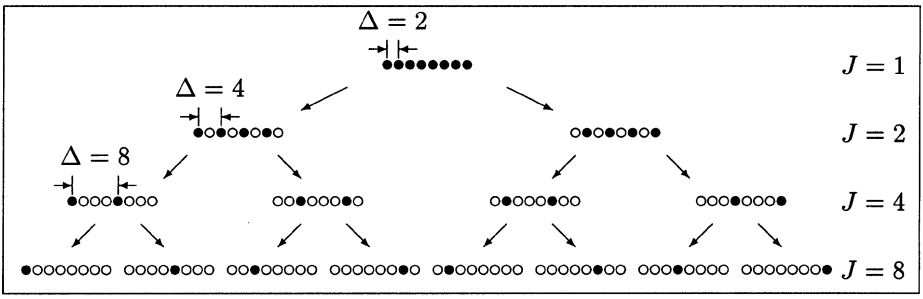


Figure 7.31: Ungerboeck partitionings for 8-level data.

As the number of data levels L goes up, the minimum intrasubset distance Δ can be kept at $\Delta = 4$ by accommodating more data levels within each subset. Conversely, Δ can be increased by using more subsets. This is illustrated in Fig. 7.31 for $L = 8$. For any value of L, a partitioning into J subsets ($J \in \{1, \ldots, L\}$) suffices to achieve a distance $\Delta = 2J$. The degenerate cases $J = 1$ and $J = L$ correspond to the DFE and full-fledged VD, respectively. Typically $\Delta = 4$ suffices for optimum performance when the channel has minimum phase [16]. Then only $J = 2$ states are needed, irrespective of L.

Combined set partitioning and sequence feedback Set partitioning may be viewed as an instantaneous form of sequence feedback and is therefore a natural complement to it [16]. Just as sequence feedback, it works best for system responses q_k with minimum

phase. For $M > 1$ and $L > 2$ both techniques may be combined to achieve substantial complexity reductions at little or no expense to performance.

Let us assume that the symbol a_{k-i} in the state vector $\mathbf{s}_k = [a_{k-M}, \ldots, a_{k-1}]$ is partitioned into $J^i \in \{1, \ldots, L\}$ subsets. Associated with a_{k-i} is an integer $x^i_{k-i} \in \{0, \ldots, J^i - 1\}$ that defines which subset a_{k-i} is in. Together the subsets x^i_{k-i} form the *subset state* $\mathbf{t}_k \triangleq [x^M_{k-M}, \ldots, x^1_{k-1}]$ of the sequence a_k at instant k. For Viterbi detection to be possible, it is necessary that the next subset state $\mathbf{t}_{k+1}$ is fully determined by the current state $\mathbf{t}_k$ and the newly appended subset x^0_k. This requires the set partitionings to be constrained such that

1. the numbers J^i are non-increasing (i.e. $J^1 \geq J^2 \geq \ldots \geq J^M$), and

2. partitioning i is a further partition of the partitionings $i + 1, \ldots, M$.

The trellis generated by the subset states is referred to as a *subset trellis*. For uncoded transmission it contains a total of $J^1 J^2 \ldots J^M$ states. The VD with local sequence feedback is a special case that emerges when $J^i = L$ for $i \in \{0, \ldots, M'\}$ and $J^i = 1$ for $i \in \{M' + 1, \ldots, M\}$. The subset trellis of Fig. 7.29 arises for $M = M' = 1$ and $J^1 = 2$.

7.9 RESTRICTED-DELAY DETECTION

We have argued in previous sections that small detection delays may suffice for close-to-optimum performance, particularly when data is coded. In this section we consider the issue of restricted-delay maximum-likelihood detection in more detail. We first consider the basic detection technique and compare it to the Viterbi detector. We then develop a dedicated detector structure that is most practical for small detection delays. After a performance analysis we finally discuss some sub-optimum restricted-delay schemes.

Fixed-delay tree search In the system of Fig. 7.32, the detector operates on a sequence $z_k = (a * q)_k + n_k$, where q_k is the system response and n_k is Gaussian white noise. The smallest and largest values of i for which $q_i \neq 0$ are referred to as the *pure delay* and *memory length* of the channel, and are denoted D_0 and M, respectively. For simplicity we neglect any pure delay, i.e. we take $\mathsf{D}_0 = 0$. The detector has a *detection delay* of D symbol intervals, i.e. at instant k it produces a decision $\hat{a}_{k-\mathsf{D}}$ with respect to $a_{k-\mathsf{D}}$, taking account of all detector inputs z up until z_k.

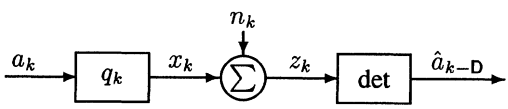


Figure 7.32: Discrete-time transmission system with restricted-delay detector.

Let us assume that transmission starts at instant $k = 0$, i.e. that $a_k = z_k = 0$ for $k < 0$. Then the first decision must be made at time $k = \mathsf{D}$ and pertains to a_0. The detector must base this decision on the $\mathsf{D}+1$ nontrivial input samples $z_0, \ldots, z_{\mathsf{D}}$. These samples depend in part on digits a_k that are already known ($a_k = 0$ for $k < 0$), and for the rest on $\mathsf{D}+1$ digits $a_0, \ldots, a_{\mathsf{D}}$ that are as yet unknown. Since the transmission code is known, we can compile a list that contains the relevant portion of all possible data sequences. To distinguish these *candidate data patterns* from each other and from the actual one we shall label them with a superscript n. To detect a_0, a path metric

$$\lambda_{\mathsf{D}}^n \triangleq \sum_{k=0}^{\mathsf{D}} (z_k - x_k^n)^2 \tag{7.24}$$

is computed for every data sequence a_k^n in the list. The smallest of all metrics demarcates the data sequence with greatest current likelihood. The zeroth digit of this sequence serves as the decision $\hat{a}_0$ with respect to a_0.

We are now ready to detect the next digit a_1. Here we are allowed to observe one additional input sample $z_{\mathsf{D}+1}$, and our list of candidate data patterns must be extended to cover $k = \mathsf{D}+1$. For uncoded binary data this would cause the list to double in size, and further doublings would occur at subsequent time instants. This explosive growth is clearly unpractical. It can be avoided by basing detection of a_1 on the assumption that $\hat{a}_0$ is correct. All candidate data patterns that are incompatible with this decision are dropped from the extended list. This shortens the list back to its original size. Furthermore, since all digits $\hat{a}_0^n$ are now identical, the same will be true for all noiseless channel outputs x_0^n and all branch metrics $(z_0 - x_0^n)^2$. It follows that we can disregard these metrics, so that detection of a_1 can be based on path metrics

$$\lambda_{\mathsf{D}+1}^n \triangleq \sum_{k=1}^{\mathsf{D}+1} (z_k - x_k^n)^2. \tag{7.25}$$

These metrics again depend on a total of $\mathsf{D} + 1$ branch metrics.

Detection of subsequent symbols proceeds analogously. At instant k, it is assumed that all past decisions $\hat{a}_0, \ldots, \hat{a}_{k-\mathsf{D}-1}$ are correct, and the candidate data patterns can differ only for the instants $k - \mathsf{D}, \ldots, k$. The complexity of the detection process is governed by the size of the list of candidate data patterns, which is constant in time but grows exponentially with D. For example, if data is uncoded and binary then at any instant k the list contains $2^{\mathsf{D}+1}$ entries. This clearly favours a small detection delay.

Example 7.16 Let data be uncoded and binary, and let $\mathsf{D} = 2 < M$. Then there are $2^{\mathsf{D}+1} = 8$ possible data patterns $a_{k-\mathsf{D}-M}^n, \ldots, a_k^n$ that are represented in Fig. 7.33 in the form of a binary tree. In essence, the tree can be viewed as a sliding portion of a trellis diagram. Values of candidate data symbols are noted alongside the nodes of the tree. Every i^{th} branch of the tree (i.e. every connection between nodes at times $i - 1$ and i) uniquely determines a noiseless channel output x_i and thereby a branch metric $\beta_i = (z_i - x_i)^2$. Each candidate data sequence corresponds to a single leaf of the tree (i.e. to a single final node) and vice versa. The path metric of a leaf is just the sum of all branch metrics along the branches to the leaf.

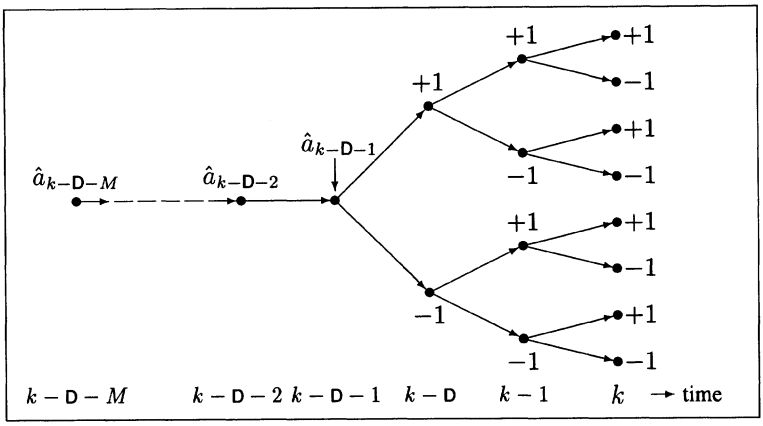


Figure 7.33: Data tree with depth $\mathsf{D} = 2$ for uncoded binary data.

The tree structure remains unchanged from one symbol interval to the next. As time progresses, the root node slides along a path that forms the sequence of restricted-delay decisions. Except for possible error propagation effects, these decisions are optimum for the detection delay D at hand. Just as in the VD, path metrics can be updated in a recursive fashion by adding the metrics of the most recently added branches.

Exercise 7.5 Assume that the system response is $1 - \mathcal{D}^2$. Note noiseless channel outputs x_k alongside the branches in the tree of Fig. 7.33.

In view of the sliding-tree representation of Fig. 7.33, the detection technique at hand is sometimes called *fixed-delay tree search* (FDTS) [44].

For $\mathsf{D} \geq M$, the restricted-delay detector is essentially equivalent to a normal VD with path registers of $\mathsf{D} - M$ stages, in which the final decision is taken from the path with smallest metric [15]. For $\mathsf{D} < M$, on the other hand, it is essentially equivalent to a VD with sequence feedback (see Section 7.7) in which the path registers are altogether omitted. For uncoded binary data this detector has 2^{D} states, and decisions $\hat{a}_{k-\mathsf{D}}$ with a delay of D symbol intervals are determined by the state that is most likely at instant k. These are applied to a shift register with $M - \mathsf{D}$ stages whose outputs $\hat{a}_{k-\mathsf{D}-1}, \ldots, \hat{a}_{k-M}$ are fed back for the branch metric computations. For $\mathsf{D} < M$, because of the feedback operation, the detection technique is sometimes succinctly called *fixed-delay tree search with decision feedback* (FDTS/DF) [44].

Finite-window ML bit detector The detection process described above can be carried out efficiently by means of a *finite-window ML detector* that is rooted in work of Gitlin and Ho [27]. In order to develop this detector we recall that detection of the first symbol

a_0 is based on the $\mathsf{D}+1$ input samples

$$\begin{array}{rcccccccc} z_0 = & & & & & a_0 q_0 & + & n_0, \\ z_1 = & & & a_0 q_1 & + & a_1 q_0 & + & n_1, \\ \vdots & & & \vdots & & \vdots & & \vdots \\ z_{\mathsf{D}} = & a_0 q_{\mathsf{D}} + \ldots + & & a_{\mathsf{D}-1} q_1 & + & a_{\mathsf{D}} q_0 & + & n_{\mathsf{D}}. \end{array} \tag{7.26}$$

Once a decision $\hat{a}_0$ is taken, it is desirable to undo any effect of a_0 on the detector inputs so as to facilitate upcoming decisions. To approximate this ideal the detector forms adjusted inputs $z_0' = z_0 - \hat{a}_0 q_0, z_1' = z_1 - \hat{a}_0 q_1, \ldots$. These will not depend on a_0 if $\hat{a}_0$ is correct, as we will assume. This means that we neglect the effect of *error propagation*. The adjusted symbol $z_0' = n_0$ consists only of noise and is irrelevant to the detection process provided that noise is white, as we will assume. For this reason, detection of the second symbol a_1 can be based on the $\mathsf{D}+1$ adjusted inputs

$$\begin{array}{rcccccccc} z_1' = & & & & & a_1 q_0 & + & n_1, \\ z_2' = & & & a_1 q_1 & + & a_2 q_0 & + & n_2, \\ \vdots & & & \vdots & & \vdots & & \vdots \\ z_{\mathsf{D}+1} = & a_1 q_{\mathsf{D}} + \ldots + & & a_{\mathsf{D}} q_1 & + & a_{\mathsf{D}+1} q_0 & + & n_{\mathsf{D}+1}. \end{array} \tag{7.27}$$

It is obvious that the same decision rule that was applied to produce $\hat{a}_0$ can be re-used to form $\hat{a}_1$. From this point on we can re-adjust the detector inputs to eliminate their dependence on both a_0 and a_1, and subsequent processing can proceed analogously. As a result, detection at instant k is based on a total of $\mathsf{D}+1$ adjusted inputs $s_k^0, \ldots, s_k^{\mathsf{D}}$. In the absence of past decision errors these are given by

$$\begin{array}{rcccccccc} s_k^0 = & & & & & a_{k-\mathsf{D}} q_0 & + & n_{k-\mathsf{D}}, \\ s_k^1 = & & & a_{k-\mathsf{D}} q_1 & + & a_{k-\mathsf{D}+1} q_0 & + & n_{k-\mathsf{D}+1}, \\ \vdots & & & \vdots & & \vdots & & \vdots \\ s_k^{\mathsf{D}} = & a_{k-\mathsf{D}} q_{\mathsf{D}} + \ldots + & & a_{k-1} q_1 & + & a_k q_0 & + & n_k. \end{array} \tag{7.28}$$

Let us consider s_k^{D} in more detail. Two cases are of interest:

$\mathsf{D} \geq M$: Here s_k^{D} depends linearly on *all* nonzero values of the system response, i.e. $s_k^{\mathsf{D}} = (a * q)_k + n_k = z_k$ (compare Fig. 7.32).

$\mathsf{D} < M$: Here s_k^{D} may be regarded as a version of z_k in which all trailing ISI with a span larger than D has been removed. To construct s_k^{D} from z_k, past decisions $\hat{a}_{k-\mathsf{D}-1}, \ldots, \hat{a}_{k-M}$ may be applied to a feedback filter with coefficients $q_{\mathsf{D}+1}, \ldots, q_M$, whose output $\sum_{j=\mathsf{D}+1}^{M} \hat{a}_{k-j} q_j$ is subtracted from z_k. One easily verifies that the resulting sequence s_k^{D} indeed satisfies (7.28) when the involved decisions are all correct.

The remaining decision variables $s_k^{\mathsf{D}-1}, \ldots, s_k^1, s_k^0$ may be constructed in a recursive fashion. Specifically, with (7.28) one establishes that s_k^{i-1} and s_{k-1}^i are related according to $s_k^{i-1} = s_{k-1}^i - \hat{a}_{k-\mathsf{D}-1} q_i$ for $k = 1, 2, \ldots$ and $i = 1, 2, \ldots, \mathsf{D}$. We arrive at the

detector structure of Fig. 7.34. The feedback filter (FBF) degenerates when $\mathsf{D} > M$. At every instant k, a $(\mathsf{D}+1)$-dimensional memoryless detector produces a decision $\hat{a}_{k-\mathsf{D}}$ based on observation of the vector $\underline{s}_k \triangleq [s_k^0, \ldots, s_k^{\mathsf{D}}]$ of decision variables. It remains to specify the decision rule. We consider two simple cases in more detail.

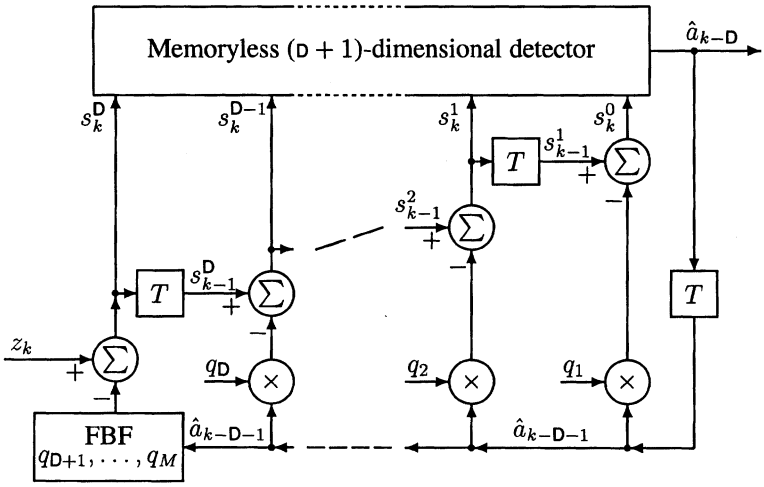


Figure 7.34: Finite-window ML bit detector.

Example 7.17 For $\mathsf{D} = 0$ there is only a single decision variable $s_k^0 = a_k + n_k$. To form this variable, the FBF in Fig. 7.34 has impulse response $q_1, \ldots, q_M$ and cancels all trailing ISI in z_k. A sign operation on s_k^0 suffices to obtain an instantaneous decision $\hat{a}_k$. We arrive at the familiar feedback detector of Fig. 7.35.

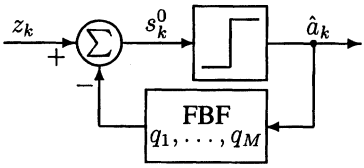


Figure 7.35: For $\mathsf{D} = 0$ the structure of Fig. 7.34 degenerates into the back-end of a decision feedback equalizer.

Fig. 7.35 adheres to 'standard' DFE usage in that the FBF incorporates a pure delay of T seconds. In Fig. 7.34, this delay occurs between the output of the detector and the FBF input.

Example 7.18 For $\mathsf{D} = 1$ there are two decision variables $s_k^0 = a_{k-1}q_0 + n_{k-1}$ and $s_k^1 = a_{k-1}q_1 + a_k q_0 + n_k$. Let us denote their data components as x_k^0 and x_k^1, i.e. $x_k^0 = a_{k-1}q_0$ and $x_k^1 = a_{k-1}q_1 + a_k q_0$. If data is binary then there are only four possible vectors $\underline{x}_k \triangleq [x_k^0, x_k^1]$. The actual detector input $\underline{s}_k$ is a noisy version of one of these nominal inputs. Fig. 7.36 illustrates the signal constellation that arises for the system response of Fig. 7.23, which has

$q_0 = 1$ and $q_1 = 0.5$. Displacements of (s_k^0, s_k^1) away from the nominal inputs are entirely due to the noise components (n_{k-1}, n_k). The detector determines the nominal input $\underline{x}_k$ that is closest to $\underline{s}_k$ in the Euclidian sense, and outputs the corresponding digit a_{k-1} as its decision. This digit has maximum likelihood because n_{k-1} and n_k are mutually uncorrelated and have equal power. The detection plane of Fig. 7.36 can be subdivided into *decision regions*. The threshold that separates both regions runs halfway between the closest dot and square. The decision $\hat{a}_{k-1} = -1$ is taken whenever (s_k^0, s_k^1) falls left of the threshold, while $\hat{a}_{k-1} = 1$ to the right of it. A decision error cannot occur unless the noise pair (n_{k-1}, n_k) has a combined power $n_{k-1}^2 + n_k^2$ of at least $5/4$. This corresponds to an improvement of around 1 dB over the DFE.

Problem 7.25 develops a circuit that implements the memoryless 2D-detector. A feedback detector that is essentially equivalent to the one at hand was described in [76] and is dubbed *extended DFE* (EDFE) .

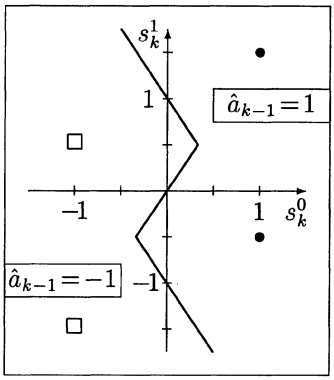


Figure 7.36: Possible detector inputs (s_k^0, s_k^1) may be represented in a two-dimensional graph. The noiseless inputs (x_k^0, x_k^1) with $a_{k-1} = 1$ and $a_{k-1} = -1$ are indicated with dots and squares, respectively. The zig-zag line demarcates the two-dimensional detection threshold.

Detection proceeds in a similar way when $\mathsf{D} > 1$. Conceptually, the detector keeps track of the $(\mathsf{D}+1)$-dimensional signal constellation, i.e. of all nominal input vectors $\underline{x}_k$. The detector finds the nominal input that is closest to the actual input $\underline{s}_k$ in the Euclidian sense, and outputs the corresponding digit $a_{k-\mathsf{D}}$ as its decision. The multidimensional decision threshold is fully determined by the signal constellation. Unfortunately, the 'shape' of this threshold tends to become complicated for large D, and complexity tends to grow accordingly. For this reason the detector is most suited for small values of D, say $\mathsf{D} = 1$ or $\mathsf{D} = 2$.

All of the above detectors can be viewed as high-SNR approximations to the delay-constrained optimum detector devised by Abend and Fritchmann [1]. Further results in this context may be found in [15]. Just as the DFE and VD, the detectors may be tailored to nonlinear channels via a look-up table that stores the input-output relation of the channel [43].

Performance of the optimum restricted-delay detector Let $\underline{a}^1$ and $\underline{a}^2$ be two admissible data sequences. Then the difference $\underline{e} \triangleq (\underline{a}^1 - \underline{a}^2)/2$ is said to be an *admissible*

error sequence. Denote by E the set of all admissible error sequences $\underline{e}$ with $e_0 \neq 0$ and $e_k = 0$ for $k \notin \{0, \ldots, \mathsf{D}\}$. Let noise n_k at the detector input be white and have variance σ^2. Then for any detection delay D, the *restricted-delay weight* $d^2(\underline{e}, \mathsf{D})$ of a given sequence $\underline{e} \in E$ is defined as

$$d^2(\underline{e}, \mathsf{D}) \triangleq \sum_{k=0}^{\mathsf{D}} (e * q)_k^2. \tag{7.29}$$

The smallest possible restricted-delay weight is denoted $d^2(\mathsf{D})$, i.e.

$$d^2(\mathsf{D}) = \min_{\underline{e} \in E} d^2(\underline{e}, \mathsf{D}). \tag{7.30}$$

In Appendix 7B it is shown for sufficiently high signal-to-noise ratios that the restricted-delay detector achieves a predetection signal-to-noise ratio $\rho_{\mathrm{RDD}} = d^2(\mathsf{D})/\sigma^2$, were σ^2 denotes the variance of the noise n_k.

We recall from Chapter 3 that the minimum distance $d_{\min}$, which governs the performance of the full-fledged VD, is independent of the phase characteristics of the channel. This is, in general, untrue for the restricted-delay distance.

Example 7.19 Let us reconsider the systems of Example 7.14 with uncoded binary data, memory length $M = 2$, and system responses $(q_0, q_1, q_2) = (4, 4, 1)$, $(2, 5, 2)$ and $(1, 4, 4)$ (see Fig. 7.27). As noted earlier, these systems have identical amplitude characteristics (see Fig. 7.28) and minimum, linear and maximum phase, respectively. For all channels, it is easy to verify that $d(\mathsf{D})$ is generated by the alternating error sequences $\underline{e} = \ldots + - + - + - \ldots$ whenever $d(\mathsf{D}) < d_{\min}$, i.e. for $\mathsf{D} \leq 10$, 13 and 17, respectively. Beyond these points $d(\mathsf{D}) = d_{\min}$, and both distances are generated by the double bit-error $e_k = \pm(\delta_k - \delta_{k-1})$. By comparison, the distance d^2_{MFB} that characterizes the matched-filter bound is generated by single bit-errors ($e_k = \pm\delta_k$) and amounts to $d^2_{\mathrm{MFB}} = q_0^2 + q_1^2 + q_2^2 = 33$. An overview of these distances is given in Table 7.5. Especially for small values of D, minimum phase is clearly preferable. This may be shown to hold very generally [8].

(q_0, q_1, q_2)	$d^2(0)$	$d^2(1)$	$d^2(2)$	$d^2(3)$	$d^2(4)$	$\ldots$	$d^2(\infty) = d^2_{\min}$	d^2_{MFB}
$(4, 4, 1)$	16	16	17	18	19	$\ldots$	26	33
$(2, 5, 2)$	4	13	14	15	16	$\ldots$	26	33
$(1, 4, 4)$	1	10	11	12	13	$\ldots$	26	33

Table 7.5: Restricted-delay distance profile of the channels of Fig. 7.27.

A comparison of the restricted-delay distances $d^2(\mathsf{D})$ with the distances $d^2_{M'}$ of Table 7.4 reveals that $d^2(\mathsf{D}) \leq d^2_{M'}$ whenever $\mathsf{D} = M'$. This verifies that preliminary decisions with a delay of $\mathsf{D} = M'$ symbol intervals in the VD with sequence feedback cannot be more reliable than the corresponding final decisions.

Clearly $d(\mathsf{D}) \leq d_{\min}$ for any channel and any $\mathsf{D} \geq 0$. Typically $d(\mathsf{D}) = d_{\min}$ as $\mathsf{D} \to \infty$. A possible exception to this rule is formed by channels whose transfer completely vanishes at one or more frequencies. Here so-called *quasi-catastrophic*

error propagation may occur, and in some cases becomes serious enough to govern restricted-delay distances. Then $d(\mathsf{D}) < d_{\min}$ even as $\mathsf{D} \to \infty$. Major cases in point are partial-response channels. Appendix 7B.1 provides a review of restricted-delay effects in first-order channels, including $1 \pm \mathcal{D}$ partial-response channels.

7.9.1 Suboptimum restricted-delay detectors

Apart from the optimum detector discussed above, various suboptimum restricted-delay schemes have been described in the literature. These schemes tend to be comparatively simple and some of them are quite effective.

In one category of schemes, exemplified by the so-called ISI canceler [26] and by the detectors of [69], [35], [42], [53] and [74], decisions are produced in two or more stages. A simple first stage (usually a DFE) produces *preliminary* decisions, which are used in a second stage to cancel pre-cursive ISI. That ISI typically arises because the prefilter of the second stage is more bandlimited than that of the first one so as to achieve a better predetection signal-to-noise ratio. Post-cursive ISI in the second stage is canceled via feedback of the decisions of that stage. Those decisions may serve as final decisions, as in the ISI canceler, or further stages may be added as in [53]. Early performance projections [26], [46] have suggested that the ISI canceler can achieve the matched-filter bound and this has triggered a flurry of interest. Later studies have revealed that error propagation from the first stage to subsequent ones tends to restrict performance [70], [4], [8]. As a result, the matched-filter bound is generally well beyond reach but modest improvements with respect to the DFE are sometimes feasible [70, 4].

A second category of schemes, exemplified by [27], [33], [13] and [9], is closely related to the VD with sequence feedback. The detector of [9], which is rooted in the one of [13], is particularly simple and is illustrated in Fig. 7.37. Designated dual DFE (DDFE), it uses two DFEs that run independently of each other and most of the time deliver the same decisions. Both DFEs are identical except that the threshold levels of their bit-detectors have antisymmetric offsets $\pm\alpha$ where $\alpha \simeq 0.2 - 0.3$. Most of the time, noise is small and decisions $\hat{a}_k^1$ and $\hat{a}_k^2$ of both DFEs are correct and identical. Occasionally, however, the decision variables $\tilde{a}_k^1$ and $\tilde{a}_k^2$ of the DFEs fall within the *erasure zone* $[-\alpha, \alpha]$. In this event the decisions are uncertain and are, moreover, different. Since both decisions are applied to a shift register with D stages, the detector can permit itself a total of D symbol intervals to figure out which of both DFE outputs to settle on. Throughout this *erasure period* detection thresholds are zeroed so that both DFEs will produce unbiased decisions. The DFE that delivers correct decisions is likely to produce small error samples $e_k = \tilde{a}_k - \hat{a}_k$ during the erasure period, while the error samples of the other DFE are likely to be relatively large because the erroneous decision causes imperfect cancellation of trailing ISI. A comparison of the energy of both errors across the erasure period is used to select between both DFEs. The switch in Fig. 7.37 is set accordingly, and both DFEs are realigned by transferring the register contents of the 'selected' DFE to the other one. Detection thresholds then re-assume their default values $\pm\alpha$, and erasure detection recommences.

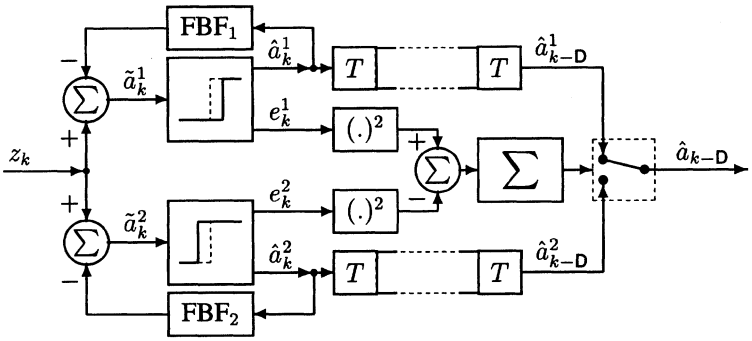


Figure 7.37: Schematic model of DDFE. Control logic is not shown.

For the scheme of Fig. 7.37 to work properly erasures must, upon average, be spaced sufficiently far apart so as not to 'bite' each other. This is true at SNRs of practical interest provided that noise at the detector input is approximately white, as it is for the DFE. The scheme prevents initial bit-errors from occurring, and thereby also prevents any error propagation that these initial errors would have induced. Bursts of errors tend to be shorter than for the DFE. Erasures occur much more frequently than bit errors and can be counted for an accurate prediction of the bit-error rate.

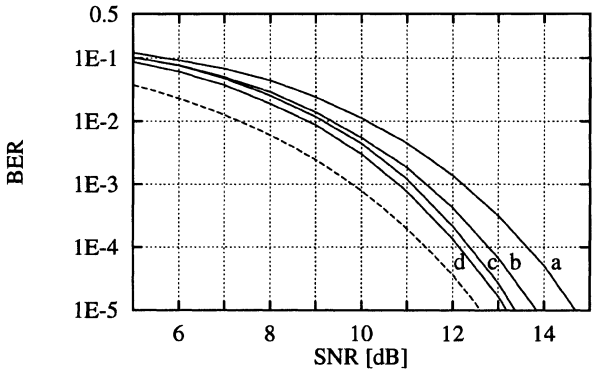


Figure 7.38: Bit-error-rate (BER) of various receivers versus signal-to-noise ratio (SNR) for a Lorentzian recording channel with white noise and normalized information density $t_{50}/T = 2$: a. DFE; b. optimum restricted-delay detector with $\mathsf{D} = 2$; c. DDFE ($\mathsf{D} = 4$, $\alpha = 0.25$); d. optimum restricted-delay detector with $\mathsf{D} = 4$. The matched-filter bound is also shown (dashed).

The DDFE typically bridges a sizable fraction of the gap between the DFE and the matched-filter bound. This is illustrated in Fig. 7.38, which pertains to uncoded storage on a Lorentzian recording channel with white noise and a normalized information density $t_{50}/T = 2$. The prefilter approximates a canonical whitened matched filter. Compared to the optimum restricted-delay detectors, which even for $\mathsf{D} = 2$ are much more complicated, the DDFE shows up remarkably well.

A final category of schemes that deserves mention is tailored to run-length-limited codes in conjunction with specific target responses, such as EPRIV or EEPRIV. Examples may be found in [54], [49], [50], [28], and [67]. These schemes are similar in flavour to the one of Section 7.7.2.

7.10 FURTHER READING

The Viterbi algorithm was first developed for decoding of convolutional codes [65]. In the early seventies its applicability to intersymbol interference channels was recognized independently by Kobayashi, Forney and Omura [34, 23, 48]. Tutorial expositions of the Viterbi algorithm may be found in the excellent papers of Forney [23], [24] and in the book of Viterbi and Omura [66]. Prefiltering was already suggested in [23] and first elaborated by Falconer and Magee [17]. Sequence feedback in the form discussed above has arrived only recently [14, 16, 5, 11],[2] but has some early precursors [51, 33]. Set partitioning originates in the classical paper of Ungerboeck on trellis-coded modulation [63]. Its use to simplify the Viterbi detector is due to Eyuboğlu and Qureshi [16]; a related approach is described in [71]. The restricted-delay feedback detector of Section 7.9 originates in [27]. The detectors treated here all operate on the output of a whitened matched filter (WMF). An alternative variant that operates directly on the sampled output of a matched filter, developed independently by Ungerboeck and MacKechnie [62], [38], has not been discussed. Other subjects that were underexposed include analysis and bounds of VD performance [22], [39], [41], [61], efficient VD implementation (digital: [29], [68], [55], [19], [52], [78], [56], [57], [21] and many more; analog: [59] and references therein), and the many heuristic simplifications of the VD that have pervaded the literature (see [2], [12], [13], [49], [53], [54] for a small sample). These simplified VDs are often tailored to specific codes and/or channels, and can be quite effective.

It is worth mentioning that the Viterbi algorithm is closely related to other fast algorithms for applications such as Fourier transformation and sorting [10]. A slightly modified form can be used to determine minimum Euclidian and Hamming distances. Other applications that were not touched upon here include the decoding of convolutional codes, demodulation of continuous-phase FSK signals, and text recognition [24].

APPENDIX 7A

Viterbi detection for charge-constrained codes

7A.1 VD performance

In Fig. 7A.1. a charge sequence $c_k \in \{C_l, \ldots, C_u\}$ is transformed into a binary data sequence $a_k = c_k - c_{k-1} \in \{-1, 1\}$ that is transmitted. The detector input z_k is a filtered

[2] See also [40] for an essentially equivalent variant.

and noisy version of a_k according to $z_k = (a*f)_k + n_k$, where f_k is the impulse response of the channel and n_k is white noise. The Viterbi detector (VD) produces decisions $\hat{c}_k$ with respect to c_k, from which decisions $\hat{a}_k$ with respect to a_k are derived by means of a $1 - \mathcal{D}$ operation.

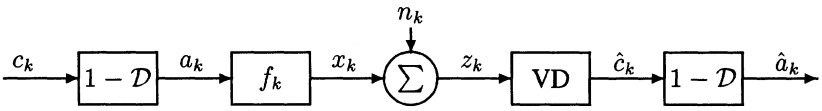


Figure 7A.1: Equivalent discrete-time model of transmission system with charge-constrained modulation code and Viterbi detector.

The detector must distinguish the actually transmitted charge sequence c_k from other allowable charge sequences. Equivalently, it must distinguish the transmitted data sequence a_k from other admissible data sequences, or the actual noiseless channel output x_k from other admissible outputs. Error performance of the detector is determined by the periods during which it fails to do this correctly. At the beginning of such an *error event*, the detected charge sequence $\hat{c}_k$ diverges from c_k, and at the end it remerges. The corresponding data sequence $\hat{a}_k$ will differ from a_k in at least two bit positions, and the error sequence $e_k \triangleq (a_k - \hat{a}_k)/2$ will be DC-free. A direct consequence is that single bit-errors cannot occur. Error performance of the VD is determined by the minimum distance $d_{\min}$ defined in Section 3.11. In VD terminology, this is basically the minimum Euclidian distance between all sequences x_k that are compatible with the trellis diagram underlying the VD. Exploitation of the DC-constraint restricts the class of admissible sequences x_k and thereby the class of admissible error sequences e_k. This often results in a performance improvement.

Example 7.20 Let us assume that the channel does not introduce any ISI, i.e. that r_k is just a noisy version of a_k ($f_k = \delta_k$), as in Example 7.5. We consider two cases:
bit-by-bit detection: Here decisions $\hat{a}_k$ are produced without regard to the DC constraint, and single bit-errors can occur. The corresponding error sequence has Euclidian weight $d^2_{\min} = \sum_k e_k^2 = 1$.
Viterbi detection: Exploitation of the DC-constraint ensures that error sequences will be DC-free. Of all sequences within this class, the double bit error has the smallest Euclidian weight $d^2_{\min} = 1^2 + 1^2 = 2$. This is a factor 2, i.e. 3 dB, better than for bit-by-bit detection. The predetection signal-to-noise ratio improves accordingly.

Example 7.21 Let the channel be duobinary $(1 - \mathcal{D})$, so that $r_k = a_k - a_{k-1} + n_k$. Three cases are of interest:
Bit-by-bit detection: Here instantaneous decisions $\hat{a}_k$ with respect to a_k are produced without any detection delay. This is possible, for example, by means of the feedback detector of Section 6.2. This detector seeks to eliminate all post-cursive ISI, and exploits only the energy of the first sample $f_0 = 1$ of f_k. Single bit-errors can occur. Correspondingly $d^2_{\min} = f_0^2 = 1$.
Partial-response Viterbi detection: Here the partial-response nature of the received sequence is exploited, but the DC-constraint is not. A two-state VD is needed (see Section 7.4). Single bit-errors can still occur, yet performance improves by 3 dB because the VD exploits all

received data energy ($d^2_{\min} = f_0^2 + f_1^2 = 2$).

Full-fledged Viterbi detection: Here the DC constraint is exploited as well. This is possible with a 6-state detector (see Section 7A.3). Valid error sequences are all DC-free. The double bit-error $e^1_k = \delta_k - \delta_{k-1}$ has filtered version $(e^1 * f)_k = \delta_k - 2\delta_{k-1} + \delta_{k-2}$ and Euclidian weight $1^2 + 2^2 + 1^2 = 6$. Its counterpart $e^2_k = \delta_k - \delta_{k-2}$ has filtered version $(e^2 * f)_k = \delta_k - \delta_{k-1} - \delta_{k-2} + \delta_{k-3}$ and weight $1^2 + 1^2 + 1^2 + 1^2 = 4 < 6$. No other valid sequence has smaller weight, i.e. $d^2_{\min} = 4$. This is 3 dB better than for partial-response Viterbi detection, and 6 dB beyond bit-by-bit detection.

Exercise 7.6 Assume that the channel introduces a single ISI term according to $f_k = \delta_k + \alpha\delta_{k-1}$ for some $\alpha \in [-1, 1]$.

1. For which values of α does the channel have a low-pass nature? For which values does it have a high-pass nature?
2. Determine $d^2_{\min}$ for a Viterbi detector that exploits all received data power but does not exploit the DC constraint.
3. Determine $d^2_{\min}$ for a full-fledged Viterbi detector. For which values of α does this detector improve upon the previous one by 3 dB? What type of channel does this correspond to? What is the smallest possible improvement, and when does it occur?

Two graphs that relate to this exercise may be found in Example 4.28 (Fig. 4.37).

This exercise shows that introduction of a DC constraint and exploitation of it in the detection process is particularly beneficial when the channel has a high-pass nature. Similarly, introduction and exploitation of constraints at other frequencies (such as the Nyquist frequency) may be advantageous when the channel passes these frequencies poorly. This insight is at the heart of an abundance of recent work on coding for partial-response channels (see, for example, [32] and references therein; see also Section 4.9.6).

7A.2 Two-state VD for full-response detection

In this section we retrace the steps of [72] to develop a simplified two-state VD for detection of the digits a_k of a charge-constrained code (see Example 7.5) in the presence of additive noise. We will see that the detector observes successive input samples and stores the value and position of the smallest such sample. The corresponding bit a_k is deemed 'ambiguous' and the accumulated charge for both possible values of the bit is considered. As the reception process progresses, one of these two values will exceed a charge constraint, thus resolving the ambiguity. At that point the receiver proceeds to select a new 'ambiguous' bit, and the process continues.

The trellis of Fig. 7.11 gives rise to branch metrics $\beta(c_k, c_{k+1}) = (z_k - a^i_k)^2$ where $a^i_k = c_{k+1} - c_k$ is the bit that is associated with the branch from c_k to c_{k+1}. Possible branch metrics are $(z_k - 1)^2$ and $(z_k + 1)^2$. The selection process depends on which of these metrics is largest. This means that we are free to subtract a common factor and to scale the result. If we take $(z_k - 1)^2$ to be the common factor and subsequently divide

by 4, then we arrive at the simplified branch metric

$$\beta'(c_k, c_{k+1}) = \begin{cases} z_k & \text{if } a_k = +1 \text{ and} \\ 0 & \text{if } a_k = -1. \end{cases} \tag{7A.1}$$

At any instant k, the two most likely states must be adjacent (see Example 7.5), i.e. they must have charges c_k and $c_k - 2$. It is only these two states that the VD is desired to keep track of. A direct consequence is that at most three state charges $c_k - 3$, $c_k - 1$ and $c_k + 1$ are accessible at instant $k+1$. One of these must be discarded while two can survive. Only a single path to each of both states can survive. Two types of transition are possible:

1. *Parallel transitions* (Fig. 7A.2.b, c): Here the surviving path metrics are both increased by the same amount (either z_k or 0), i.e. their difference Δ does not change;

2. *Merges* (Fig. 7A.2.a, d): Here the surviving paths both emanate from the same state (either c_k or $c_k - 2$), and the new path metrics differ as much as the corresponding branch metrics β'_k do, i.e. $\Delta_k = z_k - 0 = z_k$.

This shows that $\Delta_k = z_p$, where p denotes the previous instant at which a merge has occurred. This *difference metric* does not change for parallel transitions, but is reset whenever a merge occurs. At this moment the ambiguity with respect to a_p is resolved, and the current bit is next in line to be flagged ambiguous ($a_p := a_k$; $p := k$). Parallel transitions uniquely determine a current bit a_k and do not give rise to additional ambiguity.

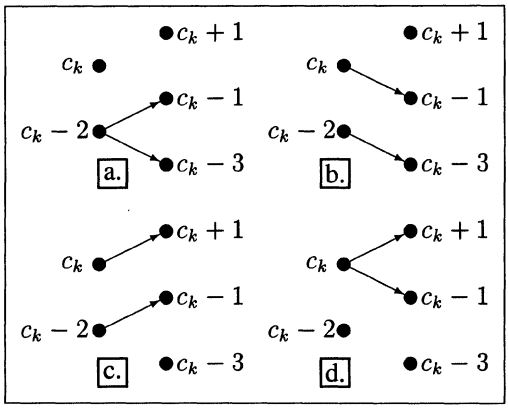


Figure 7A.2: Possible transitions in simplified two-state Viterbi detector for charge-constrained code.

Selection among the 4 possible situations of Fig. 7A.2 can be based on the difference metric z_p and the input sample z_k, without any need to keep track of the path metrics

$\lambda(c_k)$ and $\lambda(c_k - 2)$. This is similar to what happens in the Ferguson VD (see Section 7.6.2). By way of example we consider the merge of Fig. 7A.2.d. There are two situations in which this merge will occur:

1. The charge $c_k - 3$ is forbidden (i.e. $c_k - 1 = C_l(k)$, where $C_l(k) \in \{-2, -3\}$ denotes the 'bottom' of the trellis) and at the same time the path to $c_k - 1$ via c_k is more likely than that via $c_k - 2$, i.e.

$$\lambda(c_k) + \beta'(c_k, c_k - 1) < \lambda(c_k - 2) + \beta'(c_k - 2, c_k - 1) \Leftrightarrow z_p < z_k. \quad (7A.2)$$

2. The charge $c_k - 3$ is not forbidden, $c_k + 1$ is allowed (i.e. $c_k - 1 \neq C_u(k)$, where $C_u(k) \in \{2, 3\}$ denotes the current 'ceiling' of the charge trellis), and at the same time the two paths via c_k are more likely than both paths via c_{k-2}. The latter condition can be split into two sub-conditions that must be met simultaneously:

   (a) The path to $c_k - 1$ via c_k must be more likely than that via $c_k - 2$; this is condition (7A.2).

   (b) The path to $c_k + 1$ via c_k must be more likely than the path to $c_k - 3$ via $c_k - 2$, i.e.

$$\lambda(c_k)+\beta'(c_k, c_k+1) < \lambda(c_k-2)+\beta'(c_k-2, c_k-3) \Leftrightarrow z_p+z_k < 0. \quad (7A.3)$$

In summary, the merge of Fig. 7A.2.d occurs if and only if $[z_k > z_p \wedge c_k - 1 = C_l(k)] \vee [z_k > z_p \wedge z_k < -z_p \wedge c_k - 1 \neq C_u(k)]$. The involved comparisons are simple and concern only z_p, z_k and c_k. There is no need to keep track of path metrics λ. Similar conditions can be derived for the other three situations of Fig. 7A.2. The resulting detection rules are summarized in Table 7.1.

Case	Conditions	VD output	Updates c_{k+1}	z_p	p
a.	$[z_k < z_p \wedge c_k - 1 = C_l(k))\vee$ $[z_k < z_p \wedge z_k > -z_p \wedge c_k - 1 \neq C_u(k)]$	$a_p := -1$	$c_k - 1$	z_k	k
b.	$[z_k > z_p \wedge c_k - 1 = C_l(k))\vee$ $[z_k > z_p \wedge z_k > -z_p \wedge c_k - 1 \neq C_u(k)]$	$a_k := -1$	$c_k - 1$	z_p	p
c.	$[z_k < z_p \wedge c_k - 1 = C_l(k))\vee$ $[z_k < z_p \wedge z_k < -z_p \wedge c_k - 1 \neq C_u(k)]$	$a_k := 1$	$c_k + 1$	z_p	p
d.	$[z_k > z_p \wedge c_k - 1 = C_l(k))\vee$ $[z_k > z_p \wedge z_k < -z_p \wedge c_k - 1 \neq C_u(k)]$	$a_p := 1$	$c_k + 1$	z_k	k

Table 7.1: Detection rules of simplified two-state VD for charge-constrained binary data.

7A.3 Six-state VD for partial-response detection

We first consider the Bipolar $(1-\mathcal{D})$ response. Here the channel has memory length $M = 1$, and a noiseless channel output $x_k = a_k - a_{k-1}$ is determined by three successive charges c_{k-2}, c_{k-1}, c_k. This gives rise to states $\mathbf{s}_k = [c_{k-2}, c_{k-1}]$. For $C_l = C_u = 3$ (as in Miller-Squared) there are 6 states for both even and odd k. The corresponding trellis diagram is depicted in Fig. 7A.3. Upgoing and downgoing transitions all have $x = 0$. Horizontal transitions alternatingly have $x = \pm 2$.

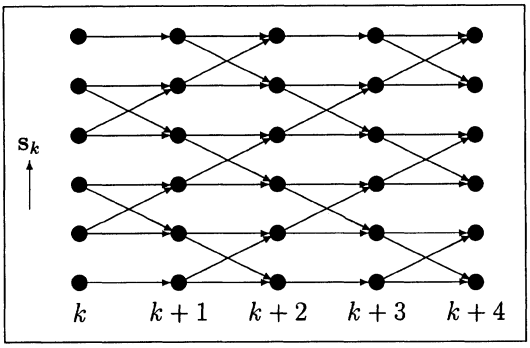


Figure 7A.3: Trellis diagram for detection of DC-constrained data ($C_l = C_u = 3$) and a channel with memory length $M = 1$.

Exercise 7.7 Note noiseless channel outputs x_k as well as the corresponding data symbols a_k alongside transitions in Fig. 7A.3.

The 6-state VD is fully determined by the trellis and is not described in further detail for the sake of brevity. Fig. 7A.4 compares its performance with that of a bit-by-bit feedback detector and a Ferguson detector. Just as for uncoded data, the Ferguson detector requires around 3 dB less SNR for a given bit-error rate than the feedback detector. The 6-state VD can provide a further advantage of around 2 dB for bit-error rates below 10^{-5}. This necessitates a detection delay of at least 100 symbol intervals.

For a Duobinary $(1 + \mathcal{D})$ channel, the modulation code should preferably be constrained at the Nyquist frequency because the channel does not pass this frequency at all. The development of a VD for this situation is completely analogous to the one that was just sketched. The Class IV $(1 - \mathcal{D}^2)$ channel rejects both DC and the Nyquist frequency. Here it is possible to interleave two independent DC-constrained data sequences to obtain a data sequence a_k that has no content at either frequency (see Section 4.9.4). De-interleaving in the receiver (as in Section 7.6) permits the even and odd bit streams to be detected independently. This is possible by means of two detectors that are identical to the above one but run at half the data rate.

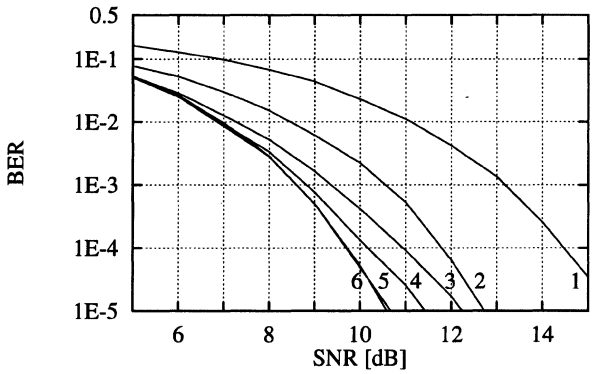


Figure 7A.4: Bit error rate (BER) versus signal-to-noise ratio (SNR) for various detectors operating on a $1 - \mathcal{D}$ channel with DC-constrained (M^2) data. 1: Bit-by-bit feedback detector; 2: Ferguson detector ($\mathsf{D} = 16$); 3-6: Full-fledged Viterbi detector (3: $\mathsf{D} = 8$; 4: $\mathsf{D} = 32$; 5: $\mathsf{D} = 64$; 6: $\mathsf{D} = 128$).

APPENDIX 7B

Effects of a restricted detection delay

This appendix, based largely on [8], is concerned with the performance of the maximum-likelihood restricted-delay detector of Section 7.9. We first recapitulate and extend some nomenclature. Let $\underline{a}^1$ and $\underline{a}^2$ be two admissible data sequences. Then the difference $\underline{e} \triangleq (\underline{a}^1 - \underline{a}^2)/2$ is called an *admissible error sequence*. Let E be the set of all admissible error sequences $\underline{e}$ with $e_0 \neq 0$ and $e_k = 0$ for $k \notin \{0, \ldots, \mathsf{D}\}$. For any detection delay D, the *restricted-delay weight* $d^2(\underline{e}, \mathsf{D})$ of a given sequence $\underline{e} \in E$ is defined as

$$d^2(\underline{e}, \mathsf{D}) \triangleq \sum_{k=0}^{\mathsf{D}} (e * q)_k^2. \tag{7B.1}$$

The smallest possible restricted-delay weight is denoted $d^2(\mathsf{D})$, i.e.

$$d^2(\mathsf{D}) = \min_{\underline{e} \in E} d^2(\underline{e}, \mathsf{D}). \tag{7B.2}$$

When data is binary, admissible error sequences $\underline{e}$ are all ternary. For this case let $\overline{\underline{e}}$ be the error sequence with smallest Hamming weight among all sequences with weight $d^2(\underline{e}, \mathsf{D}) = d^2(\mathsf{D})$, and denote this smallest weight by $W(\mathsf{D})$. (The Hamming weight of an error sequence $\underline{e}$ with symbols $e_k \in \{-1, 0, 1\}$ is defined as the number of positions k where e_k is nonzero.)

Theorem 7.1 *For the restricted-delay detector of Section 7.9, the probability $Pr[E, \mathsf{D}]$ of a first decision-error occurring at some arbitrary instant k_1 is bounded by*

$$K_0 Q(\frac{d(\mathsf{D})}{\sigma}) \leq Pr[E, \mathsf{D}] \leq K_1 Q(\frac{d(\mathsf{D})}{\sigma}) \tag{7B.3}$$

at sufficiently high signal-to-noise ratios, where $Q(x) \triangleq (2\pi)^{-1/2} \int_x^\infty e^{-y^2/2} dy$. The error coefficients K_0 and K_1 depend only on the data statistics and are typically of the same order of magnitude. In particular for uncoded binary data, $K_0 = 2^{1-W(\mathsf{D})}$ and $K_1 \leq 2^{2-W(\mathsf{D})}$.

Note 1: According to the Theorem, the detector achieves an effective predetection signal-to-noise ratio $\rho(\mathsf{D}) = d^2(\mathsf{D})/\sigma^2$ at sufficiently high signal-to-noise ratios. This should be compared with the ratios $\rho_{\text{MLSD}} = d^2_{\min}/\sigma^2$ and $\rho_{\text{MFB}} = \text{SNR}$ for the MLSD with unrestricted detection delay and the matched-filter bound, respectively. In Section 7B.1 we will compare these ratios for various first-order channels.

Note 2: Typically all error sequences $\underline{e}$ that generate $d(\mathsf{D})$ have the same Hamming weight $W(\mathsf{D})$. Then $K_0 = K_1 = 2^{1-W(\mathsf{D})}$.

Proof: In the absence of error propagation, detection of the k_1^{th} symbol is fully equivalent to detection of the first symbol a_0 at instant $k = \mathsf{D}$. Since noise is white, the maximum-likelihood rule for this decision is to compare z_k during the interval $0, \ldots, \mathsf{D}$ with the filtered versions $(a^i * q)_k$ of all admissible data sequences $\underline{a}^i$ (the superscript i is meant to distinguish these sequences from each other). The differences $z_k - (a^i * q)_k = ((a - a^i) * q)_k + n_k$ are all squared and integrated between $k = 0$ and $k = \mathsf{D}$. Out of the resulting path metrics

$$\lambda^i_{\mathsf{D}} \triangleq \sum_{k=0}^{\mathsf{D}} [((a - a^i) * q)_k + n_k]^2, \tag{7B.4}$$

the smallest one is selected, and the digit $\hat{a}_0$ of the corresponding data sequence $\underline{\hat{a}}$ is the maximum-likelihood estimate of a_0. The digits $\hat{a}_1, \ldots, \hat{a}_{\mathsf{D}}$ can be regarded as preliminary decisions that will be replaced one after another by definite decisions in subsequent detection steps. The decision $\hat{a}_0$ is erroneous whenever $\hat{a}_0 \neq a_0$. The error sequence $\underline{e}$ with symbols $e_k \triangleq (a_k - \hat{a}_k)/2$ is well-defined for all instants $k \leq \mathsf{D}$, is nonzero for $k = 0$, and is immaterial to the upcoming decision $\hat{a}_0$ for $k > \mathsf{D}$. Since we wish to bound the probability of a first decision-error occurring at $k = 0$, we take $\hat{a}_k$ to be correct for all $k < 0$, i.e. we restrict attention to error sequences $\underline{e}$ that vanish for $k < 0$. (It suffices for $\hat{a}_k$ to be correct for $k = -\max(M, \mathsf{D}), \ldots, -1$ to ensure that the detector is completely free of the influence of past errors at moment $k = 0$). Based on these considerations, we see that only sequences $\underline{e} \in E$ need to be considered.

Let us consider a given error sequence $\underline{e} \in E$. In order for the data sequence $\underline{\bar{a}} \triangleq \underline{a} - 2\underline{e}$ to be more likely than the actual data sequence $\underline{a}$, the following three conditions must be met simultaneously:

$\mathcal{C}_1$: at $k = 0$ the detector is free of the influence of past decision errors;

$\mathcal{C}_2$: $\underline{a}$ must be such that $\overline{\underline{a}} = \underline{a} - 2\underline{e}$ is admissible;

$\mathcal{C}_3$: a bad span of noise n_k during the interval $0, \ldots, \mathsf{D}$ causes the path metric $\overline{\lambda}_{\mathsf{D}}$ of $\overline{\underline{a}}$ to be smaller than the path metric λ_{D} of $\underline{a}$.

The probability of $\mathcal{C}_1$ not being true is of the order of the error proability, so that $\Pr(\mathcal{C}_1) \simeq 1$ under operating conditions of practical interest.

The probability of $\mathcal{C}_2$ depends only on the statistics of a_k. Since $e_k = 0$ for $k < 0$, $\overline{a}_k = a_k$ for $k < 0$, i.e. $\overline{a}_k$ is certainly admissible for $k < 0$. Thus $\mathcal{C}_2$ holds if $\overline{a}_k$ is admissible for all instants between 0 and D.

Example 7.22 Let us assume that data a_k is uncoded and can assume one of L different levels. Consider the ith sample of $\underline{e}$, and assume that $|e_i| = j$ for some nonnegative integer j. Then only $L - j$ values of a_i are permissible. Correspondingly

$$\Pr(\mathcal{C}_2) = \prod_{i=0}^{\mathsf{D}} \frac{L - |e_i|}{L}, \tag{7B.5}$$

because successive digits of $\underline{a}$ are independent and data levels are equiprobable. For uncoded binary data (i.e. $L = 2$) this simplifies to $\Pr(\mathcal{C}_2) = 2^{-W(\underline{e})}$ where $W(\underline{e})$ denotes the Hamming weight of $\underline{e}$, i.e. the number of positions k where e_k is nonzero.

Condition $\mathcal{C}_3$ will hold if

$$\overline{\lambda}_{\mathsf{D}} = \sum_{k=0}^{\mathsf{D}} [((a - \overline{a}) * q)_k + n_k]^2 < \lambda_{\mathsf{D}} = \sum_{k=0}^{\mathsf{D}} n_k^2, \tag{7B.6}$$

where we have used (7B.4). Upon on realizing that $a_k - \overline{a}_k = 2e_k$ we obtain after some elementary rearrangements the equivalent condition

$$\sum_{k=0}^{\mathsf{D}} (e * q)_k^2 + n_k (e * q)_k < 0. \tag{7B.7}$$

Due to its linear dependence on the white Gaussian noise n_k, the left hand side of (7B.7) is a Gaussian stochastic variable with mean

$$\mathcal{M} = \sum_{k=0}^{\mathsf{D}} (e * q)_k^2 = d^2(\underline{e}, \mathsf{D}) \tag{7B.8}$$

and variance (compare Problem 1.18)

$$\mathcal{V} = \sigma^2 \sum_{k=0}^{\mathsf{D}} (e * q)_k^2 = \sigma^2 d^2(\underline{e}, \mathsf{D}). \tag{7B.9}$$

It follows that

$$\Pr(\mathcal{C}_3) = Q(\frac{\mathcal{M}}{\sqrt{\mathcal{V}}}) = Q(\frac{d(\underline{e}, \mathsf{D})}{\sigma}). \tag{7B.10}$$

Since $\mathcal{C}_2$ and $\mathcal{C}_3$ are statistically independent, the probability $\Pr[\underline{e}, \mathsf{D}]$ that $\overline{\underline{a}} = \underline{a} - 2\underline{e}$ will be more likely at $k = \mathsf{D}$ than $\underline{a}$ is $\Pr[\underline{e}, \mathsf{D}] = \Pr(\mathcal{C}_2)\Pr(\mathcal{C}_3)$. From this point on it is a simple matter to derive tight upper and lower bounds to the probability $\Pr[E, \mathsf{D}]$ of a first decision error $\hat{a}_0$ occurring at instant $k = \mathsf{D}$:

1. A decision error *may* occur whenever one or more sequences $\underline{e} \in E$ exist for which $\overline{\lambda}_{\mathsf{D}} < \lambda_{\mathsf{D}}$. Even then it does not necessarily occur, however, because an admissible data sequence $\underline{a}' \neq \underline{a}$ with $a'_0 = a_0$ may exist with still smaller path metric at $k = \mathsf{D}$. From this and the fact that the probability of a union of events cannot exceed the sum of their individual probabilities, we conclude that

$$\Pr[E, \mathsf{D}] \leq \sum_{\underline{e} \in E} \Pr[\underline{e}, \mathsf{D}] = \sum_{\underline{e} \in E} \prod_{i=0}^{\mathsf{D}} \frac{L - |e_i|}{L} Q(\frac{d(\underline{e}, \mathsf{D})}{\sigma}). \tag{7B.11}$$

We now subdivide E into subsets E_δ with sequences $\underline{e}$ of equal weight $d(\underline{e}, \mathsf{D}) = \delta$, and denote the set of all possible values of $d(\underline{e})$ by Δ. Then

$$\Pr[E, \mathsf{D}] \leq \sum_{d(\underline{e}, \mathsf{D}) \in \Delta} Q(\frac{d(\underline{e}, \mathsf{D})}{\sigma}) \sum_{\underline{e} \in E_\delta} \prod_{i=0}^{\mathsf{D}} \frac{L - |e_i|}{L}. \tag{7B.12}$$

Since $Q(x)$ decreases exponentially in x, the right hand side of (7B.12) will be dominated at sufficiently high signal-to-noise ratios by the term involving the minimum value $d(\mathsf{D})$ of $d(\underline{e}, \mathsf{D})$, i.e.

$$\Pr[E] \leq K_1 Q(\frac{d(\mathsf{D})}{\sigma}), \tag{7B.13}$$

where

$$K_1 \triangleq \sum_{\underline{e} \in E_{d(\mathsf{D})}} \prod_{i=0}^{\mathsf{D}} \frac{L - |e_i|}{L} \tag{7B.14}$$

is a constant independent of σ, and $E_{d(\mathsf{D})}$ is the set of error events $\underline{e}$ of weight $d(\underline{e}, \mathsf{D}) = d(\mathsf{D})$.

Example 7.23 For uncoded binary data we have two sequences $\underline{e} \in E_{d(\mathsf{D})}$ with minimum Hamming weight $W(\mathsf{D})$, viz. one with $e_0 = 1$ and its sample-by-sample reversed version with $e_0 = -1$. Often $E_{d(\mathsf{D})}$ contains no other sequences $\underline{e}$. In that case $K_1 = 2.2^{-W(\mathsf{D})} = 2^{1-W(\mathsf{D})}$. The worst possible case is that $E_{d(\mathsf{D})}$ also contains pairs of sequences with Hamming weights $W(\mathsf{D}) + 1, W(\mathsf{D}) + 2, \ldots$. Then

$$K_1 = 2.2^{-W(\mathsf{D})} \sum_{k=0}^{\infty} 2^{-k} = 2^{2-W(\mathsf{D})}. \tag{7B.15}$$

In practice K_1 will be in between these two extremes.

2. Assume that a genie provides the receiver with only the sequences $\underline{a}$ and $\overline{\underline{a}} \triangleq \underline{a} - 2\underline{e}$ for some $\underline{e} \in E_{d(\mathsf{D})}$ to choose from. This obviously facilitates the task of the receiver and reduces its probability of making a decision error. When none of the sequences $\overline{\underline{a}}$ is admissible, the receiver will certainly select $\underline{a}$. Otherwise it will decide with probability $Q(d(\mathsf{D})/\sigma)$ in favour of any of the admissible sequences $\overline{\underline{a}}$. Consequently

$$\Pr[E] \geq K_0 Q(\frac{d(\mathsf{D})}{\sigma}), \tag{7B.16}$$

where $K_0 \leq 1$ is the probability that at least one of the sequences $\overline{\underline{a}}$ is admissible. When $E_{d(\mathsf{D})}$ contains the single bit-errors $e_k = \pm\delta_k$, obviously $K_0 = 1$.

Example 7.24 For uncoded binary data one easily verifies that $K_0 = 2^{-W(\mathsf{D})}$. For $L > 2$, typically $K_0 < K_1$, although the two are often close.

For certain channels with severe amplitude distortion, both K_0 and K_1 may become very small for large D (Section 7B.1 provides an example of this situation). Then some error sequences $\underline{e}' \notin E_{d(\mathsf{D})}$ may have a probability $\Pr[\underline{e}', \mathsf{D}] > K_1 Q(d(\mathsf{D})/\sigma)$ for most signal-to-noise ratios of practical interest. In this region $\Pr[\underline{e}', \mathsf{D}]$ rather than $K_0 Q(d(\mathsf{D})/\sigma)$ or $K_1 Q(d(\mathsf{D})/\sigma)$ should be taken as an approximation of $\Pr[E, \mathsf{D}]$. Here the theorem will apply only at very (and possibly unrealistically) high signal-to-noise ratios.

7B.1 Influence of D for a first-order channel

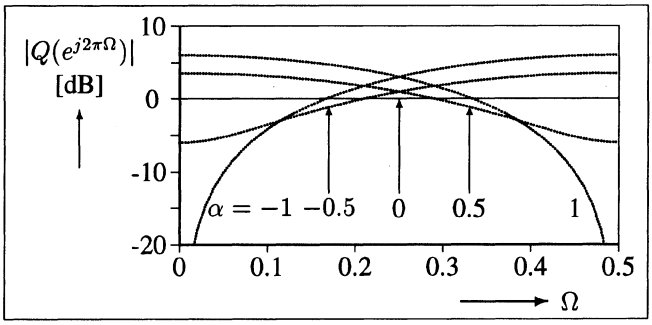


Figure 7B.1: Amplitude $|Q(e^{j2\pi\Omega})|$ versus normalized frequency Ω for various values of α.

By way of illustration we consider uncoded binary transmission across a system with $q_0 = 1$, $q_1 = \alpha$ for some $\alpha \in [-1, 1]$ and $q_k = 0$ otherwise. Depending on α, this first-order minimum-phase channel has a low-pass, all-pass or high-pass nature (Fig. 7B.1). For $\alpha = -1$ and $\alpha = 1$ we have the Bipolar and Duobinary partial-response channels, respectively. One easily verifies that $d^2_{\min} = 1 + \alpha^2$ and $d^2(0) = 1$ for all $\alpha \in [-1, 1]$. Both are generated by the single bit errors $e_k = \pm\delta_k$

with Hamming weight $W(0) = 1$. Thus $K_0 = K_1 = 0.5$. For $\mathsf{D} = 1$ the error sequences $\underline{e}$ with $e_k = \pm(\delta_k + \delta_{k-1})$, $\pm\delta_k$, and $\pm(\delta_k - \delta_{k-1})$ have restricted-delay weight $d^2(\underline{e}, \mathsf{D}) = 1+(\alpha+1)^2$, $1+\alpha^2$ and $1+(\alpha-1)^2$, and generate $d^2(1)$ for $\alpha \in [-1, -0.5]$, $[-0.5, 0.5]$ and $[0.5, 1]$, respectively. A similar behaviour occurs for $\mathsf{D} > 1$. Table 7.1 summarizes the generating events $\underline{e}(\mathsf{D})$, the corresponding weights $W(\mathsf{D})$, multipliers K_0 and K_1, and the restricted-delay distances $d^2(\mathsf{D})$.

α	$\underline{e}(\mathsf{D})$	$w(\mathsf{D})$	K_0, K_1	$d^2(\mathsf{D})$
$\in [-1, -\frac{\sqrt{\mathsf{D}}}{1+\sqrt{\mathsf{D}}}]$	$+ + + + + \ldots$ (D+1 terms)	$1+\mathsf{D}$	$2^{-\mathsf{D}}$	$1+\mathsf{D}(1+\alpha)^2$
$\in [-\frac{\sqrt{\mathsf{D}}}{1+\sqrt{\mathsf{D}}}, \frac{\sqrt{\mathsf{D}}}{1+\sqrt{\mathsf{D}}}]$	$+$ (single bit-error)	1	0.5	$1+\alpha^2$
$\in [\frac{\sqrt{\mathsf{D}}}{1+\sqrt{\mathsf{D}}}, 1]$	$+ - + - + \ldots$ (D+1 terms)	$1+\mathsf{D}$	$2^{-\mathsf{D}}$	$1+\mathsf{D}(1-\alpha)^2$

Table 7.1: Restricted-delay characteristics of $1 + \alpha\mathcal{D}$ channel for detection delays $\mathsf{D} \geq 0$.

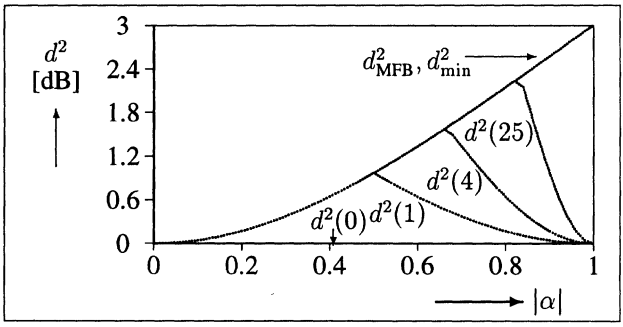


Figure 7B.2: Distances $d^2_{\mathrm{MFB}} = d^2_{\min}$ and $d^2(\mathsf{D})$ for selected values of D versus $|\alpha|$.

The dependence of $d^2_{\mathrm{MFB}} = d^2_{\min}$ and $d^2(\mathsf{D})$ on α is depicted in Fig. 7B.2. In order for $d^2(\mathsf{D})$ to coincide with $d^2_{\min}$, $|\alpha|$ should not exceed $\sqrt{\mathsf{D}}/(1+\sqrt{\mathsf{D}})$. Equivalently $\mathsf{D} \geq \lceil \alpha^2/(1-|\alpha|)^2 \rceil$, where $\lceil x \rceil$ denotes the smallest integer equal to or greater than x (Fig. 7B.3). Required delays $\overline{\mathsf{D}}$ increase progressively with α. For $|\alpha| \leq 0.5$, $Q(e^{j2\pi\Omega})$ has little amplitude distortion (see Fig. 7B.1), and $\overline{\mathsf{D}} = 1$. Conversely, a spectral zero appears as $|\alpha| \to 1$, and $\overline{\mathsf{D}} \to \infty$.

Of special interest is the partial-response case $|\alpha| = 1$. Here $W(\mathsf{D}) = 1 + \mathsf{D}$, $K_0 = K_1 = 2^{-\mathsf{D}}$ and $d^2(\mathsf{D}) = 1 < d^2_{\min} = 2$ for all D. Correspondingly $\Pr[E, \mathsf{D}] \simeq 2^{-\mathsf{D}}Q(1/\sigma)$ for sufficiently high signal-to-noise ratios SNR $\triangleq 2/\sigma$. Now $d^2(\mathsf{D})$ does not depend on D, while the multiplier $2^{-\mathsf{D}}$ becomes very small for large D. By contrast, the single bit error (which generates $d^2_{\min}$) has a probability $\Pr[\underline{e}, \mathsf{D}] = Q(2/\sigma)$ that does not depend on D and may exceed $2^{-\mathsf{D}}Q(1/\sigma)$ for SNRs of practical interest. For bit-error rates $\Pr[E]$ of practical interest the cross-over point may be determined with the help of the approximation [77, p. 83] $Q(x) \simeq (\sqrt{2\pi}x)^{-1}e^{-x^2/2}$. It is demarcated by $\mathsf{D} \simeq -0.5\log_2 \mathrm{P}[E]$. Hence modest delays $\mathsf{D} \simeq 7 - 15$ suffice for practical values

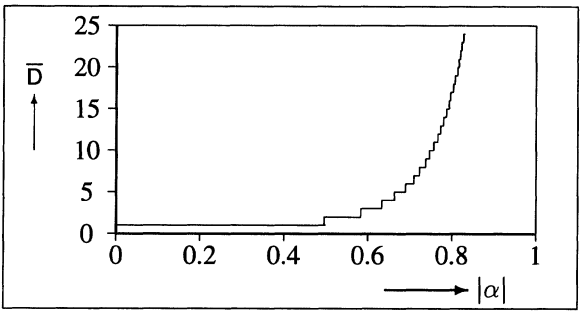


Figure 7B.3: Minimum detection delay $\overline{\mathsf{D}}$ needed for $d^2(\mathsf{D})$ to coincide with $d^2_{\min}$.

$\mathrm{P}[E] \simeq 10^{-5} - 10^{-10}$, while $\mathsf{D} \to \infty$ is needed as $\mathrm{P}[E] \downarrow 0$. Here $d^2(\mathsf{D}) = 1$ rather than $d^2_{\min} = 2$ is the effective distance.

As $\mathsf{D} \to \infty$, the error sequence $\underline{e}(\mathsf{D})$ that generates $d(\mathsf{D})$ corresponds to a state of error with *finite* weight $d^2(\mathsf{D}) = 1$ from which the system never recovers. This behaviour is dubbed *quasi-catastrophic error propagation* because it is similar to catastrophic error propagation with convolutional codes, but can be ignored as $\mathsf{D} \to \infty$ because $K_0 = K_1 \downarrow 0$. It arises whenever $Q(e^{j2\pi\Omega})$ has spectral zeros and an admissible semi-infinite error pattern $\underline{e}$ has all its energy concentrated at those zeros. Partial-response channels are the most relevant examples.

APPENDIX 7C

Performance of VD with prefilter

In this appendix we analyze the performance of a VD that is preceded by a prefilter. The filter serves to transform the symbol response of the channel into a prescribed target response to which the VD is tailored. As such the filter is essentially a partial-response equalizer as discussed in Chapter 6, and we will use the results of that chapter to assess the performance degradation that accrues if the VD is replaced by a bit-by-bit detector. Our VD analysis builds upon Appendix 3D, which was concerned with the effect of colored noise on the performance of the maximum-likelihood sequence detector. For simplicity we model the system in a discrete-time form. For notational convenience we omit the argument $e^{j2\pi\Omega}$ of discrete-time Fourier transforms throughout the appendix.

The received sequence $r_k = (a*\gamma)_k + n_k$ in Fig. 7C.1 represents the sampled output of a whitened matched filter (WMF). Here γ_k is the impulse of the system up until the WMF and n_k is additive white Gaussion noise of variance 1. As shown in Chapter 3, the amplitude characteristic $|\Gamma|$ is fully determined by the folded signal-to-noise ratio of the original continuous-time channel according to $|\Gamma|^2 = \mathrm{FSN}$. The prefilter that precedes the VD has impulse response w_k. It transforms γ_k into a target response $g_k = (\gamma * w)_k$

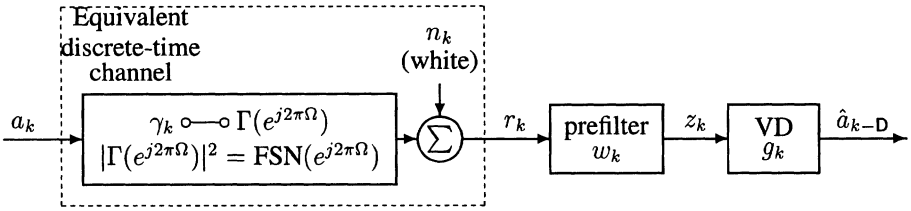


Figure 7C.1: Discrete-time cascade of channel, prefilter and VD. The prefilter serves to transform the symbol response γ_k of the channel into the target response g_k underlying the VD.

that is typically desired to have a small duration so as to simplify the VD. The VD input z_k may be written $z_k = (a * g)_k + u_k$, where $u_k \triangleq (n * w)_k$ is filtered noise. This noise is, in general, no longer white. For this reason the decisions produced by the VD are no longer optimum.

Noise at the output of the prefilter has PSD $\mathcal{U} = \mathcal{N}|W|^2$, where $\mathcal{N}$ is the PSD of n_k and W is the transfer function of the prefilter. Now n_k is white with unit variance, i.e. $\mathcal{N} = 1$ for all Ω. Furthermore, $W = G/\Gamma$, whence $|W|^2 = |G|^2/|\Gamma|^2 = |G|^2/\mathrm{FSN}$. It follows that $\mathcal{U} = |G|^2/\mathrm{FSN}$. This substitution specializes the results of Appendix 3D (and in particular Theorem 3.2) to the situation of Fig. 7C.1.

Example 7.25 In many instances, g_k is a partial response of the $1 \pm \mathcal{D}$ or $1 - \mathcal{D}^2$ variety. Here $\int_{-0.5}^{0.5} |G|^2 d\Omega = \sum_k g_k^2 = 2$. Provided that the $|G|^2$ resembles FSN, single bit-errors are likely to prevail. In this event the VD achieves a predetection signal-to-noise ratio

$$\rho_{\mathrm{PR\ VD}} = \frac{4}{\displaystyle\int_{-0.5}^{0.5} |G|^2 \mathcal{U} d\Omega} = \frac{4}{\displaystyle\int_{-0.5}^{0.5} \frac{|G|^4}{\mathrm{FSN}} d\Omega}. \tag{7C.1}$$

By comparison, we have seen in Chapter 6 that the zero-forcing partial-response equalizer with bit-by-bit detection achieves a predetection signal-to-noise ratio

$$\rho_{\mathrm{ZF\ PRE}} = \frac{1}{\displaystyle\int_{-0.5}^{0.5} \frac{|G|^2}{\mathrm{FSN}} d\Omega}. \tag{7C.2}$$

As argued in Chapter 6, $|G|^2$ should be selected to resemble FSN. Ideally $|G|^2 \simeq C\mathrm{FSN}$ for some frequency-independent constant C. Here we find that $\rho_{\mathrm{PR\ VD}}/\rho_{\mathrm{ZF\ PRE}} \simeq 2$, i.e. some 3 dB is gained, as conjectured in Section 7.6. In practice $|G|^2$ may not resemble FSN very well for any of the responses $1 \pm \mathcal{D}$ and $1 - \mathcal{D}^2$, and gains are bound to be more modest.

Exercise 7.8 Assume that the target system response is Class IV (i.e. $|G|^2 = 4\sin^2 2\pi\Omega$) and that single bit-errors prevail, so that (7C.1) applies. Argue that the VD is hardly sensitive to disturbances in the vicinity of both DC and the Nyquist frequency. What is the implication for allowable prefilter tolerances near these two frequencies? In which frequency region are the accuracy requirements on the prefilter most severe?

APPENDIX 7D

Performance of VD with sequence feedback

Denote by $\hat{\mathbf{s}}_k$ the VD state with greatest likelihood (i.e. smallest path metric λ_k) at instant k. For the full-fledged VD, an error event is said to occur between the instants k_1 and k_2 if $\hat{\mathbf{s}}_k$ coincides with the actual state $\mathbf{s}_k$ of the channel at $k = k_1$ and $k = k_2$ but $\hat{\mathbf{s}}_k \neq \mathbf{s}_k$ for all $k \in \{k_1 + 1, \ldots, k_2 - 1\}$ [23]. Unlike the full-fledged VD, the branch metrics β_k for the VD with sequence feedback depend not only on the current state $\mathbf{s}_k^i$ from which the branch emanates and the newly appended digit a_k, but also on the $M - M'$ most recent digits $a_{k-M}^i, \ldots, a_{k-M'-1}^i$ of the surviving path for state $\mathbf{s}_k^i$. Therefore, to ensure that no error propagation was in progress at the instant k_1, it is necessary to assume that $\hat{\mathbf{s}}_k$ was correct not only at $k = k_1$, but also during the $M - M'$ instants immediately preceding k_1. Also, the condition $\hat{\mathbf{s}}_{k_2} = \mathbf{s}_{k_2}$ does not necessarily demarcate the end of a cluster of errors now, because incorrect preliminary decisions $\hat{a}_{k_2-M}, \ldots, \hat{a}_{k_2-M'-1}$ will cause erroneous contributions in the metric calculations during the instants $k_2, \ldots, k_2+M-M'-1$. In this appendix we neglect this error propagation problem. We do this by splitting a cluster of errors up into an error event that is defined to occur between the instants k_1 and k_2, and a subsequent propagation-induced part that is not accounted for in the analysis.

Let $\underline{\hat{a}}$ be the data sequence corresponding to the detected state sequence $\underline{\hat{\mathbf{s}}}$. Then $\hat{a}_k$ coincides with the actual data symbols a_k for $k \in \{k_2 - M', \ldots, k_2 - 1\}$ because $\hat{\mathbf{s}}_k = [\hat{a}_{k-M'}, \ldots, \hat{a}_{k-1}]$ is correct at $k = k_2$. However, $\hat{a}_{k_1} \neq a_{k_1}$ and $\hat{a}_{k_2-M'-1} \neq a_{k_2-M'-1}$ because $\hat{\mathbf{s}}_{k_1+1} \neq \mathbf{s}_{k_1+1}$ and $\hat{\mathbf{s}}_{k_2-1} \neq \mathbf{s}_{k_2-1}$, respectively. Thus the error sequence $\underline{e}_k$ with symbols $e_k \triangleq (a_{k+k_1} - \hat{a}_{k+k_1})/2$ starts at instant $k = 0$ and has length $N \triangleq k_2 - k_1 - M'$. The corresponding error event starts at instant $k = k_1 + 1$ and has length $k_2 - k_1 - 1 = N + M' - 1$. Following Forney [23] we observe that the following conditions must be met simultaneously for this event to occur:

$\mathcal{C}_1$: Between $k_1 - M'$ and k_1 we must have $\hat{\mathbf{s}}_k = \mathbf{s}_k$.

$\mathcal{C}_2$: Between k_1 and k_1+N-1 the actual data sequence $\underline{a}$ must be such that the symbols $a_k - 2e_{k+k_1}$ form part of an admissible data sequence $\underline{\hat{a}}$.

$\mathcal{C}_3$: During this interval a bad span of noise causes $\underline{\hat{a}}$ to have greater likelihood than $\underline{a}_k$. Equivalently, $\underline{\hat{\mathbf{s}}}$ should be more likely than the actual state sequence $\underline{\mathbf{s}}$ for $k \in \{k_1 + 1, \ldots, k_2 - 1\}$.

The probability of $\mathcal{C}_1$ not being true is of the order of the error probability, so that $\Pr(\mathcal{C}_1) \simeq 1$ under normal operating conditions. Conditions $\mathcal{C}_2$ and $\mathcal{C}_3$ are analogous to $\mathcal{C}_2$ and $\mathcal{C}_3$ in Appendix 7B, and their probabilities may be assessed in a similar fashion. In particular, $\mathcal{C}_3$ will hold if $\hat{a}_k$ survives at instant $k = k_2$, i.e. if $\hat{\lambda}_{k_2} < \lambda_{k_2}$, where $\hat{\lambda}_k$ and λ_k denote the path metrics of $\hat{a}_k$ and a_k, respectively. Now $\hat{\lambda}_k = \sum_{j=0}^{k-1} \hat{\beta}_j$ and $\lambda_k = \sum_{j=0}^{k-1} \beta_j$, where $\hat{\beta}_k$ and β_k denote the branch metrics of $\hat{\mathbf{s}}_k$ and $\mathbf{s}_k$, respectively.

Hence

$$\hat{\lambda}_{k_2} - \lambda_{k_2} = \sum_{k=k_1}^{k_2-1} (\hat{\beta}_k - \beta_k), \tag{7D.1}$$

where we have used the fact that $\hat{s}_{k_1} = s_{k_1}$, i.e. that $\hat{\lambda}_{k_1} = \lambda_{k_1}$. Upon recalling that $\hat{\beta}_k = (z_k - (\hat{a} * q)_k)^2$, $\beta_k = (z_k - (a * q)_k)^2$, and $z_k = (a * q)_k + n_k$, we find after some elementary manipulations that

$$\hat{\lambda}_{k_2} - \lambda_{k_2} = 4 \sum_{k=0}^{N+M'-1} [(e * q)_k^2 + n_{k+k_1}(e * q)_k]. \tag{7D.2}$$

Due to its linear dependence on the white Gaussian noise n_k, the right hand side of (7D.2) is a Gaussian stochastic variable with mean

$$\mathcal{M} = 4 \sum_{k=0}^{N+M'-1} (e * q)_k^2 = 4d^2(\underline{e}) \tag{7D.3}$$

and variance (compare Problem 1.18)

$$\mathcal{V} = 16\sigma^2 \sum_{k=0}^{N+M'-1} (e * q)_k^2 = 16\sigma^2 d^2(\underline{e}). \tag{7D.4}$$

Here σ^2 denotes the variance of n_k. Thus

$$\Pr(\mathcal{C}_3) = Q(\frac{\mathcal{M}}{\sqrt{\mathcal{V}}}) = Q(\frac{d(\underline{e})}{\sigma}). \tag{7D.5}$$

From this point on, a development akin to that of Appendix 7B reveals that for sufficiently high signal-to-noise ratios, the probability $\Pr[E]$ of a first error occurring at some arbitrary time instant k_1 may be bounded by

$$K_0 Q(\frac{d_{M'}}{\sigma}) \leq \Pr[E] \leq K_1 Q(\frac{d_{M'}}{\sigma}). \tag{7D.6}$$

Here $K_0 \leq 1$ is the probability that the actual data sequence $\underline{a}$ is such that the symbols $\hat{a}_k = a_k + 2e_{k+k_1}$ form part of an admissible data sequence for at least one $\underline{e} \in E_{d_{M'}}$, where $E_{d_{M'}}$ is the set of error events $\underline{e}$ of weight $d(\underline{e}) = d_{M'}$. Furthermore, for uncoded L-level data,

$$K_1 \triangleq \sum_{\underline{e} \in E_{d_{M'}}} \prod_{i=0}^{N-1} \frac{L - |e_i|}{L}. \tag{7D.7}$$

In many instances K_0 and K_1 are of the same order of magnitude, and the above bounds will be close even for modest signal-to-noise ratios. Then $\rho_{M'} \triangleq d_{M'}^2/\sigma^2$ may be viewed as the effective predetection signal-to-noise ratio achieved by the VD with sequence feedback.

Exercise 7.9 Show that $d_{M'}$ is independent of the phase characteristics of q_k if and only if $M' = M$, i.e. for the full-fledged VD.

APPENDIX 7E

Exploitation of run-length constraints

In this appendix we show how run-length constraints can be exploited for a $1-D$ channel to arrive at a simple Viterbi detector with small detection delay. By way of illustration we consider MFM. For convenience the run-length trellis diagram of MFM is reproduced in Fig. 7E.1.

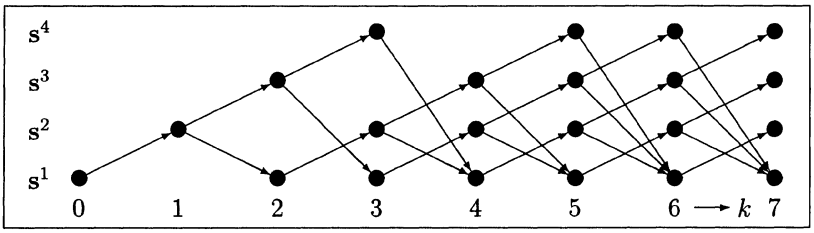


Figure 7E.1: Run-length trellis diagram of MFM.

As explained in Section 7.7.2, the diagram has four states $\mathbf{s}^1, \ldots, \mathbf{s}^4$ whose labels indicate the length of an ongoing run. Upgoing branches correspond to extension of an ongoing run by one bit, while downgoing branches correspond to a transition. For a $1-\mathcal{D}$ channel, noiseless channel outputs $x_k = a_k - a_{k-1}$ are zero for upgoing branches and $2a_k \in \{-2, 2\}$ for downgoing ones. Corresponding branch metrics are $\beta(\mathbf{s}_k, \mathbf{s}_{k+1}) = (z_k - x_k)^2$, where z_k is the detector input. Clearly $\beta(\mathbf{s}_k, \mathbf{s}_{k+1}) = z_k^2$ and $(z_k - 2a_k)^2$ for upgoing and downgoing branches, respectively. The selection process depends only on which of these metrics is largest. This means that we are free to subtract a common factor and to scale the result. If we take z_k^2 to be the common factor and subsequently divide by 4, then we arrive at the simplified branch metric

$$\beta'(\mathbf{s}_k, \mathbf{s}_{k+1}) \triangleq \begin{cases} 0 & \text{for upgoing branches, and} \\ 1 + a_{k-1} z_k & \text{for downgoing branches.} \end{cases} \tag{7E.1}$$

Associated to every state $\mathbf{s}_k^i$ is a path metric $\lambda(\mathbf{s}_k^i)$ and a path $\underline{p}(\mathbf{s}_k^i) = [a_{k-\mathsf{D}}^i, \ldots, a_{k-1}^i]$, where D is the detection delay. At the beginning of transmission we start in state $\mathbf{s}^1$. Paths are extended along the trellis, and the only add-compare-select operation that takes place at each stage pertains to the surviving path that leads to state $\mathbf{s}^1$. The straightforward implementation of this detection process requires 4 cross-coupled shift registers of length D, 4 path metric registers, and metric renormalization hardware that serves to prevent path metric overflow. We proceed to show that the complexity of this process can be significantly reduced by exploiting the special structure of the trellis.

Difference metrics The detection process may be cast in terms of difference metrics

$$\Delta_k^i \triangleq \lambda(\mathbf{s}_k^i) - \lambda(\mathbf{s}_k^1) \; (k = 1, 2, 3, 4) \tag{7E.2}$$

that vary in a restricted range around zero and hence do not grow without bound. To see how this can be done, we first note that states s^2, s^3 and s^4 are only accessible from their immediate predecessor via branches that carry no weight ($\beta' = 0$). This means that $\lambda(s_k^i) = \lambda(s_{k-1}^{i-1})$ for $i = 2, 3, 4$. Correspondingly

$$\Delta_k^2 = \lambda(s_k^2) - \lambda(s_k^1) = \lambda(s_{k-1}^1) - \lambda(s_k^1), \tag{7E.3}$$

$$\Delta_k^3 = \lambda(s_k^3) - \lambda(s_k^1) = \lambda(s_{k-2}^1) - \lambda(s_k^1) = \Delta_{k-1}^2 + \Delta_k^2 \tag{7E.4}$$

and

$$\Delta_k^4 = \lambda(s_k^4) - \lambda(s_k^1) = \lambda(s_{k-3}^1) - \lambda(s_k^1) = \Delta_{k-2}^2 + \Delta_{k-1}^2 + \Delta_k^2. \tag{7E.5}$$

The difference path metric for state s^1 is always zero and needs not be computed or stored. The nontrivial difference metrics are uniquely determined by the current and two most recent versions of $\Delta_k^2 = \lambda(s_k^2) - \lambda(s_k^1)$; all that is needed is a two-stage shift register and two adders (Fig. 7E.2).

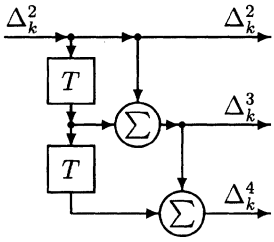


Figure 7E.2: All nontrivial difference metrics may be formed from Δ_k^2 by means of a simple delay-and-add network.

In terms of these difference metrics, detection proceeds as follows:

Survivor selection: Selection among the extended branches to s_{k+1}^1 via s_k^2, s_k^3 and s_k^4 is determined by the minimum of the extended path metrics $\lambda_k^2 + \beta'(s_k^2, s_{k+1}^1)$, $\lambda(s_k^3) + \beta'(s_k^3, s_{k+1}^1)$ and $\lambda(s_k^4) + \beta'(s_k^4, s_{k+1}^1)$. Subtraction of a common factor $\lambda(s_k^1)$ of all path metrics does not affect the selection process. With (7E.2) and (7E.1) this means that we must minimize

$$\mu_k^i \triangleq \Delta_k^i + 1 + a_{k-1}^i z_k \tag{7E.6}$$

over all $i \in \{2, 3, 4\}$. The minimizing index j and value μ_k^j must be memorized.

Update of difference metrics: With the aid of (7E.3) we see that

$$\Delta_{k+1}^2 = \lambda_k^1 - \lambda_{k+1}^1 = \lambda_k^1 - (\lambda_k^j + 1 + a_{k-1}^j z_k) = -\Delta_k^j - 1 - a_{k-1}^j z_k = -\mu_k^j. \tag{7E.7}$$

This shows that the new difference metric Δ_{k+1}^2 is just the inverse of μ_k^j. The remaining metrics Δ^3 and Δ^4 are updated via the circuit of Fig. 7E.2.

Determination of the most likely state: This state is determined by the minimum of $\lambda(\mathrm{s}^1_{k+1}), \ldots, \lambda(\mathrm{s}^4_{k+1})$, i.e. by the minimum of $0, \Delta^2_{k+1}, \Delta^3_{k+1}, \Delta^4_{k+1}$. This minimum is fully determined by the three outputs of the circuit of Fig. 7E.2.

Path memory State s^4 corresponds to the event that a transition has occurred four symbol intervals ago. If the foremost digit a^4_{k-1} of the surviving path $\underline{p}(\mathrm{s}^4_k)$ is known, then the 4 preceding digits are also known ($a^4_{k-2} = a^4_{k-3} = a^4_{k-4} = a^4_{k-1}$, $a^4_{k-5} = -a^4_{k-1}$). In fact, since any valid path through the trellis must satisfy the minimum run-length constraint, a^4_{k-6} is also known ($a^4_{k-6} = a^4_{k-5} = -a^4_{k-1}$). Similarly, the digits $a^3_{k-5}, \ldots, a^3_{k-2}$ and $a^2_{k-4}, \ldots, a^4_{k-2}$ are fully determined by a^3_{k-1} and a^2_{k-1}, respectively.

Let $j \in \{2, 3, 4\}$ denote the index of the surviving branch to state s^1_{k+1}. If $j = 2$, then the surviving path $\underline{p}(\mathrm{s}^1_{k+1})$ is a copy of $\underline{p}(\mathrm{s}^2_k)$ with a transition appended, i.e.

$$\underline{p}(\mathrm{s}^1_{k+1}) = [a^2_{k-\mathsf{D}+1}, \ldots, a^2_{k-2}, a^2_{k-1}, -a^2_{k-1}]. \tag{7E.8}$$

In this case, knowledge of the foremost digit $a^1_k = -a^2_{k-1}$ suffices to know $a^1_{k-4}, \ldots, a^1_{k-1}$. Similarly, $\underline{p}(\mathrm{s}^1_k)$ is known even further into the past if $j = 3$ or $j = 4$.

Because of the run-length constraints, a detection delay $\mathsf{D} = 3$ will suffice to approach the limiting performance for $\mathsf{D} \to \infty$ closely (we will soon quantify the degradation). In all of the above cases, a^i_{k-3} is fully determined by the index j of the surviving branch to state s^1_{k+1} and the most recent digits a^i_k of the suriving paths to states $\mathrm{s}^1_{k+1}, \ldots, \mathrm{s}^4_{k+1}$ (Table 7.1).

State	j	a^i_{k-3}
s^4_{k+1}		a^4_k
s^3_{k+1}		$-a^3_k$
s^2_{k+1}		$-a^2_k$
s^1_{k+1}	2	a^1_k
	3, 4	$-a^1_k$

Table 7.1: Possible VD outputs a^i_{k-3} for a detection delay $\mathsf{D} = 3$.

This shows that we only need to keep track of the foremost digits a^i_k ($k = 1, 2, 3, 4$) along with the index j in order to realize a detection delay $\mathsf{D} = 3$; a full-fledged path memory is not needed. The VD output digit $\hat{a}_{k-3}$ is determined via Table 7.1 according to which of the four states is most likely.

At this point we recall that states s^4, s^3 and s^2 are accessible only via their immedate predecessor. A direct consequence is that the associated surviving paths are delayed versions of each other, and in particular that $a^4_k = a^3_{k-1}$, $a^3_k = a^2_{k-1}$ and $a^2_k = a^1_{k-1}$. Hence only a^1_k needs to be determined via an add-compare-select operation; a^2_k, a^3_k and a^4_k can then be derived from a^1_k with a 3-stage shift register.

Performance The above simplifications together lead to a (d, k) detector that is roughly as simple as Ferguson's detector (see Section 7.6.2). Performance, however,

is better because the latter detector is meant for uncoded data and hence does not exploit run-length constraints. Fig. 7E.3 provides a comparison.

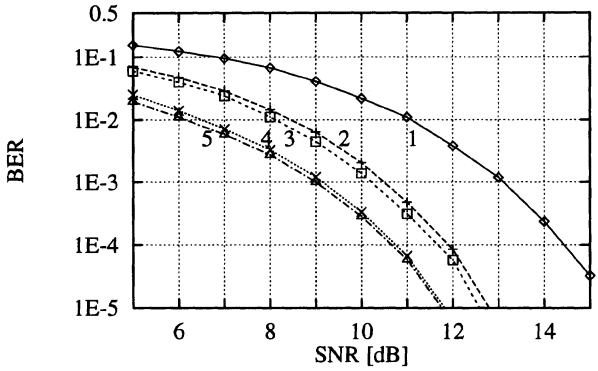


Figure 7E.3: Bit error rate (BER) versus signal-to-noise ratio (SNR) for various detectors operating on a $1 - \mathcal{D}$ channel with MFM data. 1. Bit-by-bit feedback detector; 2. Ferguson detector (D = 16); 3. Schneider detector (D = 3); 4. (d, k) detector with simplified path register (D = 3); 5. Full-fledged (d, k) detector (D = 16).

Exploitation of the run-length constraints leads to an improvement of around 1 and 4 dB over Ferugson and bit-by-bit feedback detection, respectively. For completeness fig. 7E.3 includes the detector due to Schneider [54], which exploits only the maximum run-length constraint. This detector is even simpler than the simplified (d, k) detector, yet lags around 1 dB behind it.

Exercise 7.10 Can the above approach be extended to $1 + \mathcal{D}$ detection? Would this be meaningful? (Hint: consider Example 7.7).

PROBLEMS

7.1 Consider the $1 - \mathcal{D}$ system of Section 7.3. Assume that the sequence $\underline{z} = (-0.2, 0.5, -1.5, -0.2)$ is received. Determine all path metrics in the trellis of Fig. 7.6, and detect the corresponding data sequence by means of Viterbi detection. Is this sequence reasonable?

7.2 Identify the digits a_{-1} and a_5 for the trellis diagram of Fig. 7.8. How would the diagram change if other *a priori* choices for these two digits were made?

7.3 Redraw the trellis diagram of Fig. 7.6 in terms of states.

7.4 Show that the rightmost two branches in Fig. 7.9 have equal branch metrics. What is the implication? Identify the digits a_{-2}, a_{-1}, a_5 and a_6. How would the trellis diagram change if other *a priori* choices for these 4 digits were made?

7.5 Construct a trellis diagram for uncoded transmission of $N = 5$ bits across a channel with memory length $M = 3$.

7.6 Consider transmission of a message of N symbols $a_0, \ldots, a_{N-1}$ across a channel with memory length M. Assume that digits a_k with $k < 0$ and $k \geq N$ are known *a priori*. Which of the channel outputs z_k depend in part on the initial state $\mathbf{s}_0 = [a_{-M}, \ldots, a_{-1}]$? How long does it take before the trellis diagram reaches its full breadth $|S|$? What is the index of the final state? Which of the channel outputs z_k depend in part or fully on this state? How long is the trailing portion of the trellis diagram?

7.7 Repeat Example 7.4 for a channel memory length $M = 4$. How large are $|S|$ and B? Compare these figures with those for uncoded binary transmission. Generalize these results to arbitrary M.

7.8 Redo Example 7.4 and Problem 7.7 for a run-length-limited code with $d = 2$.

7.9 The $(2, 7)$ modulation code has $d = 2$ and rate $R = 1/2$ (see Section 4.8). Use the results of the Problem 7.8 to compare the numerical effort of a Viterbi detector for $(2, 7)$ with one for uncoded transmission. Compare on the basis of equal information throughput, i.e. assume that $(2, 7)$ produces two encoded symbols every symbol interval T as opposed to one symbol in the uncoded case, so that a channel memory length M for uncoded storage gives a memory length $M' \simeq 2M$ for $(2, 7)$. How do the figures compare as $M \to \infty$?

7.10 In this problem we are concerned with the system of Fig. 7.13 for the situation that data a_k is binary and run-length-limited with minimum run-length $d = 1$ as in Example 7.4. We take noise n_k to be white with variance N_0.

1. Determine the channel signal-to-noise ratio SNR.
2. The detector input y_k is of the form $(a * q)_k + u_k$, where q_k is the overall system response and u_k is noise. Determine q_k and the variance σ_u^2 of u_k.
3. The receiver of Fig. 7.13 is just a (causal) matched filter followed by a bit-by-bit detector. In Chapter 3 we have seen that this structure is optimum whenever only a single bit a_0 is transmitted. This bit is detected at $k = 1$ because of the delay introduced by the matched filter. Determine the amplitude A of the data component of y_1 and the predetection signal-to-noise ratio $\rho_{\text{MFB}} = A^2/\sigma_u^2$. How do ρ_{MFB} and SNR relate?
4. We next consider an arbitrary data sequence a_k. On which digits a_k does y_1 depend? Which data amplitudes A are now possible? Does the $d = 1$ constraint preclude certain amplitudes? What is the smallest possible amplitude $A_{\min}$? Determine the predetection signal-to-noise ratio $\rho_{d=1} = A_{\min}^2/\sigma_u^2$. Show that $\rho_{d=1} = \rho_{\text{MFB}}$ whenever $\alpha \geq 0$. What does this mean for the bit error rate of the receiver of Fig. 7.13?

7.11 In Section 7.6.2 we have seen that only three types of merges $(\mathbf{m}^-, \mathbf{m}^0, \mathbf{m}^+)$ are possible in Viterbi detection of binary data across a $1 \pm \mathcal{D}$ channel; crossover transitions ($\mathbf{s}_k^+ \to \mathbf{s}_{k+1}^-$, $\mathbf{s}_k^- \to \mathbf{s}_{k+1}^+$) cannot occur. Does this remain true for all $1 + \alpha\mathcal{D}$ channels? If not, then for which values of α can crossover transitions occur? What are the implications for detector implementation?

7.12 Construct a picture akin to Fig. 7.20 that depicts the action of a bit-by-bit detector. When does this detector make erroneous decisions? Verify that x_k is outside the band $[z_k - 1, z_k + 1]$ in this case.

7.13 Derive Ferguson's detector for a $1 + \mathcal{D}$ channel.

7.14 Redo Problem 7.1 on the basis of Ferguson's algorithm.

7.15 In this problem we are concerned with threshold adaptation for the Ferguson detector (see Section 7.6.2) when operating on a $\alpha(1 - \mathcal{D})$ channel with *a priori* unknown gain α. The detector of Fig. 7.19 becomes suited for this purpose if the constant value 1 at the input of both summators is replaced by an estimate $\tilde{\alpha}$ of α. In the alternative detector of Fig. 7.21, the constant values -2 and $+2$ at the inputs of the leftmost switch must be replaced by $-2\tilde{\alpha}$ and $2\tilde{\alpha}$, respectively. For simplicity we restrict attention to the detector of Fig. 7.19, and we take noise to be absent. The effects of noise on adaptation are treated in Chapter 8.

1. Assume that a merge has occurred at instant $k - 1$. Express the difference metric Δ' in terms of α, $\tilde{\alpha}$, a_{k-2} and a_{k-1}.
2. The comparators in Fig. 7.19 determine the sign of sequences $d_k^+ \triangleq \Delta'_k - z_k - \tilde{\alpha}$ and $d_k^- \triangleq \Delta'_k - z_k - \tilde{\alpha}$. Assume that a merge has occurred at instant $k - 1$. Express d_k^+ and d_k^- in terms of α, $\tilde{\alpha}$, a_{k-2}, a_{k-1} and a_k. For what bit-combinations are d_k^+ and d_k^- a measure of the threshold error $\alpha - \tilde{\alpha}$?
3. Use the foregoing observations to describe a simple iterative technique for increasing or decreasing $\tilde{\alpha}$ based on a logic combination of binary sequences, including preliminary decisions $\hat{a}_{k-2}, \hat{a}_{k-1}$ and $\hat{a}_k$ with respect to a_{k-2}, a_{k-1} and a_k.

A more systematic approach towards threshold adaptation is discussed in Section 8.8.1.

7.16 In this problem we generalize the Ferguson detector along the lines of [7] to an $\alpha - \beta\mathcal{D}$ channel. For simplicity we assume that $\beta > 0$; the case $\beta < 0$ is entirely similar.

1. Redo the derivation of Section 7.6.2 for this case. Verify that the resulting algorithm is as in Table 7.1, with $\Delta' \triangleq \Delta/(4\alpha)$ and $\gamma \triangleq (\alpha - \beta)/\alpha$.

$\Delta'_k + \gamma z_k$	Δ'_{k+1}	Survivor update
$> z_k + \beta$	$z_k + \beta$	$\mathbf{m}^+$
$\in [z_k - \beta, z_k + \beta]$	$\Delta'_k + \gamma z_k$	$\mathbf{m}^0$
$< z_k - \beta$	$z_k - \beta$	$\mathbf{m}^-$

Table 7.1: Generalization of Ferguson's algorithm to $\alpha - \beta\mathcal{D}$ channel; $\gamma \triangleq (\alpha - \beta)/\alpha$.

2. Recast Table 7.1 in terms of an alternative difference metric $\overline{\Delta} \triangleq \alpha\Delta'/\beta$ and parameter $\rho \triangleq (\alpha - \beta)/\beta$. Sketch block diagrams of the corresponding detector and the one according to Table 7.1. How do their complexities compare? Could it be allowable to quantize γ and/or ρ strongly, say to a negative power of 2? What would this mean for a digital implementation?

7.17 Consider the $(1, 3)$-detector of Section 7.7.2. Verify that any possible data sequence that satisfies the $(1, 3)$ run-length constraints corresponds to a single path through the trellis of Fig. 7.26 and vice versa. Generalize the trellis to arbitrary (d, k) codes.

7.18 Verify that the detector of Section 7.7.2 indeed uses local feedback.

7.19 This problem is rooted in [6]. Consider the adaptive branch metric computation unit of Example 7.12 (Fig. 7.25). Assume that the system has a sufficiently small memory length M to be able to do without sequence feedback. How does the unit come to look in this case? Make a drawing of a convenient digital implementation with an up/down counter that is increased or decreased depending on the sign of e. Would it be a good idea to replace the squaring operation by a modulus operation?

7.20 This problem is rooted in [6]. Consider the adaptive branch metric computation unit of Example 7.12 (Fig. 7.25). Assume that the channel output r_k depends mainly on past bits $a_{k-1}, \ldots, a_{k-M}$ and hardly on the present bit a_k. Argue that the selection sequence s_k^{li} of Fig. 7.25 becomes unreliable in this case. What is the consequence for RAM adaptation? List at least two solutions for this problem.

7.21 Let q_k be the first-order exponential function

$$q_k = \begin{cases} 0, & k < 0, \text{ and} \\ q_0^{-k}, & k \geq 0 \end{cases} \tag{7P.1}$$

for some $q_0 > 1$. Show that single bit-errors prevail for $q_0 \geq 2$. Redo Example 7.13 for $q_0 = 3/2$. Which error sequences prevail in this case?

7.22 This problem is concerned with the performance of the VD with sequence feedback of Section 7.7. Let the system response q_k have memory length $M = 2$ (i.e. $q_k = 0$ for $k \notin \{0, 1, 2\}$).

1. Which responses in this category have minimum phase?
2. Find $d_{M'}^2$ (see Section 7.7.3 for a definition) and the generating error sequence e_k for $M' = 0, 1$ and 2. For which impulse responses do single bit errors prevail? Do these responses have minimum phase? Sketch their amplitude-frequency characteristics.
3. What are the worst minimum-phase channels in the sense that d_0^2/d_2^2 and d_1^2/d_2^2 are as small as possible? How large are the corresponding losses d_0^2/d_2^2 and d_1^2/d_2^2?

7.23 Consider uncoded transmission with $L \geq 2$ data levels across a $(1+\mathcal{D})^2$ partial-response channel.

1. How many states does a full-fledged VD for this system have?
2. Consider a VD with set partitioning such that $J^1 = J^2 = 2$. How many states does this VD have? Extend the reasoning of Section 7.8 to show that this VD achieves the same performance as the full-fledged VD, irrespective of L.
3. Consider a VD with set partitioning and sequence feedback such that $J^1 = 2$ and $J^2 = 1$. How many states does this VD have? How far does it lag behind its full-fledged counterpart?

7.24 Consider uncoded transmission with an L-level data alphabet. Express the constant K_1 of (7B.15) in terms of the *Lee weight* $W_L(\mathsf{D}) = \sum_{k=0}^{D} |e_k|$ of sequences $e_k \in E_{d(\mathsf{D})}$.

7.25 In this problem we develop a circuit that implements the detection process of Example 7.18. For the sake of generality we consider a system impulse response with $q_0 = 1$ and arbitrary q_1.

1. Define transformed coordinates $u = s_k^1 - q_1 s_k^0, v = -q_1 s_k^1 - s_k^0$. What is the decision threshold in the (u, v) plane?
2. Prove that the decision $\hat{a}_{k-1}$ can be taken according to

$$\hat{a}_{k-1} = \text{sgn}[-v - \frac{1}{1 + q_1 + q_1^2}\text{SAT}(u)] \tag{7P.2}$$

where SAT(u) denotes the unit saturation function according to

$$\text{SAT}(u) = \begin{cases} 1, & u > 1, \\ x, & |u| \leq 1, \\ -1, & u < -1. \end{cases} \tag{7P.3}$$

3. Validate the circuit of Fig. 7P.1 for $q_1 = 0.5$. What coefficients are needed for arbitrary q_1?

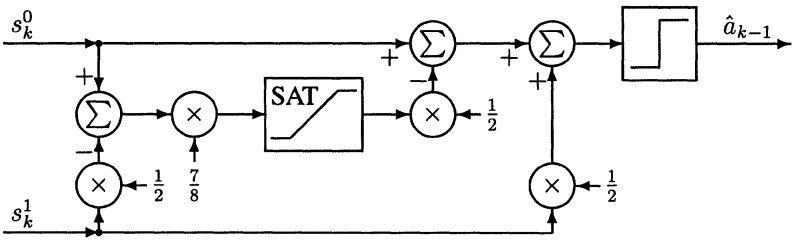


Figure 7P.1: Two-dimensional detector that takes decisions $\hat{a}_{k-1}$ in accordance with the detection threshold of Fig. 7.36.

7.26 Consider the finite-window ML detector of Fig. 7.34 for uncoded binary data and a detection delay $\mathsf{D} = 2$. Determine the number of squaring operations, additions and comparisons that are needed in a straightforward implementation of the memoryless $(\mathsf{D} + 1)$-dimensional detector. Are any simplifications possible? How do the numbers grow beyond $\mathsf{D} = 2$?

7.27 Redo Problem 7.26 for binary run-length-limited data with minimum-run-length parameter $d = 1$, starting with $\mathsf{D} = 1$. Compare the results with those of Example 7.18 and Problem 7.26.

7.28 Outline how the detection process of Section 7A.3 may be recast in terms of difference metrics. Hint: Note that the trellis consists of independent 2×2 'butterflies' in the form of a $1 - \mathcal{D}$ trellis segment to which Ferguson's algorithm is applicable.

7.29 Consider the condition $\mathcal{C}_2$ for the optimum restricted-delay detector as defined in Section 7B. Assume that data is coded. Will $\Pr(\mathcal{C}_2)$ increase or decrease relative to (7B.5)?

7.30 Let data be binary and coded according to the Bi-phase encoding rule ($a_{k-1} = -a_k$ for all even k). Redo Example 7.18 for this situation. How many points does the signal constellation have? Find a simple circuit that implements the detection rule for an arbitrary system response with $q_0 = 1$. Is performance any better than in Example 7.18 when $q_1 = 0.5$? For what values of q_1 does an improvement occur? What type of channel does this correspond to?

7.31 Extend the development of Appendix 7E to arbitrary minimum and maximum run-lengths. Pay attention in particular to data without minimum run-length constraint ($d = 0$). How does the complexity of the detector depend on the maximum run-length?

7.32 Consider Class IV ($1 - \mathcal{D}^2$) detection of $(1, 7)$. Subdivide the received data stream into even and odd bits. Determine the minimum and maximum run-length of both interleaved data streams. Which partial response is associated to each interleaved data stream? What kind of detector would be appropriate?

REFERENCES

[1] K. Abend and B.D. Fritchmann, 'Statistical Detection for Communication Channels with Intersymbol Interference', *Proc. IEEE*, Vol. 58, pp. 779-785, May 1970.

[2] J.B. Anderson and S. Mohan, 'Sequential Decoding Algorithms: A Survey and Cost Analysis', *IEEE Trans. Commun.*, Vol. COM-32, pp. 169-176, Feb. 1984.

[3] C.T. Beare, 'The Choice of the Desired Impulse Response in Combined Linear-Viterbi Algorithm Equalizers', *IEEE Trans. Commun.*, Vol. COM-26, pp. 1301-1307, Aug. 1978.

[4] J.W.M. Bergmans and Y.C. Wong, 'A Simulation Study of Intersymbol Interference Cancellation', *AEÜ (Electronics and Communication)*, Vol. 41, No. 1, pp. 33-37, 1987.

[5] J.W.M. Bergmans, S.A. Rajput and F.A.M. van de Laar, 'On the Use of Decision Feedback for Simplifying the Viterbi Detector', *Philips J. Res.*, Vol. 42, No. 4, pp. 399-428, 1987.

[6] J.W.M. Bergmans, S. Mita and M. Izumita, 'Receiver for Data Transmission System with Nonlinearities', U.S. patent no. 5,131,011, Jul. 14, 1992.

[7] J.W.M. Bergmans, K.D. Fisher and H.W. Wong-Lam, 'Variations on the Ferguson Viterbi Detector', *Philips J. Res.*, Vol. 47, No. 6, pp. 361-386, 1993.

[8] J.W.M. Bergmans, F.M.J. Willems and G.S.M. Kerpen, 'On the Performance of Data Receivers with a Restricted Detection Delay', *IEEE Trans. Commun.*, Vol. 42, No.6, pp. 2315-2324, June 1994.

[9] J.W.M. Bergmans and J.O. Voorman, 'Dual Decision Feedback Equalizer', submitted to *IEEE Trans. Commun.*, 1996.

[10] H. Burkhardt and L.C. Barbosa, 'Contributions to the Application of the Viterbi Algorithm', *IEEE Trans. Inform. Theory*, Vol. IT-31, No. 5, pp. 626-634, Sept. 1985.

[11] P.R. Chevillat and E. Eleftheriou, 'Decoding of Trellis-Encoded Signals in the Presence of Intersymbol Interference and Noise', *IEEE Trans. Commun.*, Vol. COM-37, No. 7, pp. 669-676, Jul. 1989.

[12] A.P. Clark and S.N. Abdullah, 'Near-Maximum-Likelihood Detectors for Voiceband Channels', *IEE Proc.*, Pt. F, Vol. 132, No. 6, pp. 485-490, Oct. 1985.

[13] E. Dahlman and B. Gudmundson, 'Performance Improvement in Decision Feedback Equalisers by Using 'Soft Decision", *Electronics Lett.*, Vol. 24, No. 17, pp. 1084-1085, Aug. 1988.

[14] A. Duel-Hallen and C. Heegard, 'Delayed Decision-Feedback Sequence Estimation', *IEEE Trans. Commun.*, Vol. COM-37, No. 5, pp. 48-436, May 1989.

[15] J. Erfanian, S. Pasupathy and G. Gulak, 'Reduced Complexity Symbol Detectors with Parallel Structures for ISI Channels', *IEEE Trans. Commun.*, Vol. COM-42, No. 2/3/4, pp. 1661-1671, Feb./March/Apr. 1994.

[16] V.M. Eyuboğlu and S.U.H. Qureshi, 'Reduced-State Sequence Estimation with Set Partitioning and Decision Feedback', *IEEE Trans. Commun.*, Vol. COM-36, No. 1, pp. 13-20, Jan. 1988.

[17] D.D. Falconer and F.R. Magee, Jr., 'Adaptive Channel Memory Truncation for Maximum Likelihood Sequence Estimation', *Bell Syst. Tech. J.*, Vol. 52, No. 9, pp. 1541-1562, Nov. 1973.

[18] M.J. Ferguson, 'Optimal Reception for Binary Partial Response Channels', *Bell Syst. Tech. J.*, Vol. 51, No. 2, pp. 493-505, Feb. 1972.

[19] G. Fettweis and H. Meyr, 'Parallel Viterbi algorithm Implementation: Breaking the ACS Bottleneck', *IEEE Trans. Commun.*, Vol. COM-37, No. 8, pp. 785-790, Aug. 1989.

[20] G. Fettweis, 'Algebraic Survivor Memory Management Design for Viterbi Decoders', *IEEE Trans. Commun.*, Vol. 43, No. 9, pp. 2458-2463, Sept. 1995.

[21] G. Feygin and P.G. Gulak, 'Architectural Tradeoffs for Survivor Sequence Memory Management in Viterbi Decoders', *IEEE Trans. Commun.*, Vol. COM-41, No. 3, pp. 425-429, March 1993.

[22] G.D. Forney, Jr., 'Lower Bounds on Error Probability in the Presence of Large Intersymbol Interference', *IEEE Trans. Commun.*, Vol. COM-20, No. 2, pp. 76-78, Feb. 1972.

[23] G.D. Forney, Jr., 'Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference', *IEEE Trans. Inform. Theory*, Vol. IT-18, No. 3, pp. 363-378, May 1972.

[24] G.D. Forney, Jr., 'The Viterbi Algorithm', *Proc. IEEE*, Vol. 61, No. 3, pp. 268-278, March 1973.

[25] S.A. Fredricsson, 'Joint Optimization of Transmitter and Receiver Filters in Digital PAM Systems with a Viterbi Detector', *IEEE Trans. Inform. Theory*, Vol. IT-22, No. 2, pp. 200-210, Mar. 1976.

[26] A. Gersho and T.L. Lim, 'Adaptive Cancellation of Intersymbol Interference for Data Transmission', *Bell Syst. Tech. J.*, Vol. 60, No. 11, pp. 1997-2021, Nov. 1981.

[27] R.D. Gitlin and E.Y. Ho, 'A Null-Zone Decision Feedback Equalizer Incorporating Maximum-Likelihood Bit Detection', *IEEE Trans. Commun.*, Vol. COM-23, No. 11, pp. 1243-1250, Nov. 1975.

[28] N.H. Gottfried, 'Low Complexity Viterbi Detector for Magnetic Disc Drives', *IEE Proc.-E*, Vol. 140, No. 1, pp. 78-80, Jan. 1993.

[29] P.G. Gulak and T. Kailath, 'Locally Connected VLSI Architectures for the Viterbi Algorithm', *IEEE J. Selected Areas Commun.*, Vol. SAC-6, No. 3, pp. 527-537, Apr. 1988.

[30] A.P. Hekstra, 'An Alternative to Metric Rescaling in Viterbi Decoders', *IEEE Trans. Commun.*, Vol. COM-37, No. 11, pp. 1220-1222, Nov. 1989.

[31] A.P. Hekstra, 'On the Numerical Range of the Path Metrics in a Binary Viterbi Decoder', to appear in *IEEE Trans. Inform. Theory.*

[32] R. Karabed and P. Siegel, 'Matched Spectral-Null Codes for Partial-Response Channels', *IEEE Trans. Inform. Theory*, Vol. IT-37, No. 3, pt. II, pp. 818-855, May 1991.

[33] G. Kawas-Kaleh, 'Double Decision Feedback Equalizer', *Frequenz*, Vol. 33, No. 5, pp. 146-149, 1979.

[34] H. Kobayashi, 'Application of Probabilistic Decoding to Digital Magnetic Recording Systems', *IBM J. Res. Develop.*, pp. 64-74, Jan. 1971.

[35] R. Kohno, H. Imai, and M. Hatori, 'Automatic Equalizer Including a Decoder of Error-Correcting Code and Its Development', *Electron. and Commun. in Japan*, Pt. I, Vol. 68, No. 11, pp. 66-77, 1985.

[36] H. Kubo, K. Murakami and T. Fujino, 'An Adaptive Maximum-Likelihood Sequence Estimator for Fast Time-Varying Intersymbol Interference Channels', *IEEE Trans. Commun.*, Vol. COM-42, No. 2/3/4, pp.. 1872-1880, Feb./March/Apr. 1994.

[37] H.-D Lin and D.G. Messerschmitt, 'Parallel Viterbi Decoding Methods for Uncontrollable and Controllable Sources', *IEEE Trans. Commun.*, Vol. COM-41, No. 1, pp. 62-69, Jan. 1993.

[38] L.R. MacKechnie, 'Maximum-Likelihood Receivers for Channels Having Memory', Ph.D. Dissertation, Dep. Elec. Eng., Univ. of Notre Dame, Notre Dame, Jan. 1973.

[39] F.R. Magee, Jr. and J.G. Proakis, 'An Estimate of the Upper Bound on Error Probability for Maximum-Likelihood Sequence Estimation on Channels Having a Finite-Duration Pulse Response', *IEEE Trans. Inform. Theory*, Vol. IT-19, pp. 699-702, Sept. 1973.

[40] K. Matsushita, A. Iketani, A. Ide and C. Yamamitsu, 'Path Feedback Viterbi Detection Without Open Eye Patterns', *IEEE Transl. J. on Magn. in Japan*, Vol. 3, No. 8, pp. 642-652, Aug. 1988.

[41] D.G. Messerschmitt, 'A Geometric Theory of Intersymbol Interference - Part II: Performance of the Maximum-Likelihood Detector', *Bell Syst. Tech. J.*, Vol. 52, No. 9, pp. 1521-1539, Nov. 1973.

[42] S. Mita, M. Izumita, N. Doi and Y. Eto, 'Automatic Equalizer for Digital Magnetic Recording Systems', *IEEE Trans. Magn.*, Vol. MAG-23, No. 5, pp. 3672-3674, Sept. 1987.

[43] C.S. Modlin and J.M. Cioffi, 'Reduced-State Nonlinear Equalization for Magnetic Storage', *Proc. GLOBECOM'94*, pp. 1134-1138, San Fransisco, Nov. 28 - Dec. 2, 1994.

[44] J. Moon and L.R. Carley, 'Performance Comparison of Detection Methods in Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-26, No. 6, pp. 3155-3172, Nov. 1990.

[45] R.E. Morley, Jr. and D.L Snyder, 'Maximum Likelihood Sequence Estimation for Randomly Dispersive Channels', *IEEE Trans. Commun.*, Vol. COM-27, No. 6, pp. 833-839, June 1979.

[46] S. Mueller and J. Salz, 'A Unified Theory of Data-Aided Equalization', *Bell Syst. Tech. J.*, Vol. 60, No. 9, pp. 2023-2038, Nov. 1981.

[47] S. Ölçer, 'Reduced-State Sequence Detection of Multilevel Partial-Response Signals', *IEEE Trans. Commun.*, Vol. COM-40, No. 1, pp. 3-6, Jan. 1992.

[48] J.K. Omura, 'On Optimum Receivers for Channels with Intersymbol Interference', (Abstract), presented at the IEEE Int. Symp. Information Theory, Noordwijk, Holland, June 1970.

[49] A.M. Patel, 'A New Digital Signal Processing Channel for Data Storage Products', *IEEE Trans. Magn.*, Vol. MAG-27, No. 6, pp. 4579-4584, Nov. 1991.

[50] A.M. Patel, R.A. Rutledge, and B.S. So, 'Performance Data for a Six-Sample Look-Ahead (1,7)ML Detection Channel', *IEEE Trans. Magn.*, Vol. MAG-29, pp. 4012-4014, Nov. 1993.

[51] S.U. Qureshi and E.E. Newhall, 'An Adaptive Receiver for Data Transmission over Time-Dispersive Channels', *IEEE Trans. Inform. Theory*, Vol. IT-19, pp. 448-457, July 1973.

[52] C.M. Rader, 'Memory Management in a Viterbi Decoder', *IEEE Trans. Commun.*, Vol. COM-29, No. 9, pp. 1399-1401, 1981.

[53] A. Sano and J.-I. Yamazaki, 'Generalized Decision Feedback Equalizer for Approximate Maximum-Likelihood Bit Detection of Digital Signals', *Int. J. Electronics*, Vol. 52, No. 2, pp. 167-176, 1982.

[54] R.C. Schneider, 'Sequence (Viterbi-Equivalent) Decoding', *IEEE Trans. Magn.*, Vol. MAG-26, No. 6, pp. 2539-2541, Nov. 1988.

[55] C. B. Shung, P.H. Siegel, G. Ungerboeck and H.K. Thapar, 'VLSI Architectures for Metric Normalization in the Viterbi Algorithm', *Proc. ICC'90*, Apr. 16-19, 1990, pp. 1723-1728.

[56] C.B. Shung, H.-D. Lin, R. Cypher, P.H. Siegel, and H.K. Thapar, 'Area-Efficient Architectures for the Viterbi Algorithm - Part I: Theory', *IEEE Trans. Commun.*, Vol. COM-41, No. 4, pp. 636-644, Apr. 1993.

[57] C.B. Shung, H.-D. Lin, R. Cypher, P.H. Siegel, and H.K. Thapar, 'Area-Efficient Architectures for the Viterbi Algorithm - Part II: Applications', *IEEE Trans. Commun.*, Vol. COM-41, No. 5, pp. 802-807, May 1993.

[58] P.H. Siegel, C.B. Shung, T.D. Howell, and H.K. Thapar, 'Exact Bounds for Viterbi Detector Path Metric Differences', *Proc. ICASSP'91*, pp. 1093-1096, May 13-16, 1991.

[59] R.R. Spencer and P.J. Hurst, 'Analog Implementations of Sampling Detectors', *IEEE Trans. Magn.*, Vol. MAG-27, No. 6, pp. 4516-4521, Nov. 1991.

[60] H.K. Thapar and J.M. Cioffi, 'A Block Processing Method for Designing High-Speed Viterbi Detectors', *Proc. ICC'89*, Boston, USA, June 11-14, 1989, pp. 1096-1100.

[61] W. Turin, 'Union Bounds on Viterbi Algorithm Performance', *AT&T Techn. J.*, Vol. 64, No. 10, pp. 2375-2385, Dec. 1985.

[62] G. Ungerboeck, 'Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data-Transmission Systems', *IEEE Trans. Commun.*, Vol. COM-22, No. 5, pp. 624-636, May 1974.

[63] G. Ungerboeck, 'Channel Coding with Multilevel/Phase signals', *IEEE Trans. Inform. Theory*, Vol. IT-28, No. 1, pp. 55-67, Jan. 1982.

[64] F.L. Vermeulen and M.E. Hellman, 'Reduced State Viterbi Decoders for Channels with Intersymbol Interference', Proc. ICC'74, Minneapolis, MN, pp.37B-1 to 37B-9, June 1974.

[65] A.J. Viterbi, 'Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm', *IEEE Trans. Inform. Theory*, Vol. IT-13, pp. 260-269, Apr. 1967.

[66] A.J. Viterbi and J.K. Omura, *Principles of Digital Communication and Coding*. Tokyo: McGraw-Hill Kogakusha, Ltd., 1979.

[67] D. Welland et al., 'Implementation of a Digital Read/Write Channel with EEPR4 Detection', *IEEE Trans. Magn.*, Vol. MAG-31, No. 2, pp. 1180-1185, March 1995.

[68] K.A. Wen and J.Y. Lee, 'Parallel Processing for Viterbi Algorithm', *Electronics Lett.*, Vol. 24, No. 17, pp. 1098-1099, Aug. 1988.

[69] K. Wesołowski and J.G. Proakis, 'A Simplified Two-Stage Equalizer with a Reduced Number of Multiplications for Data Transmission over Voiceband Telephone Links', *IEEE J. Selected Areas Commun.*, Vol. SAC-2, No. 5, pp. 731-742, Sept. 1984.

[70] K. Wesołowski, 'On the Performance and Convergence of the Adaptive Canceller of Intersymbol Interference in Data Transmission', *IEEE Trans. Commun.*, Vol. COM-33, No. 5, pp. 425-432, May 1985.

[71] K. Wesołowski, 'An Efficient DFE&ML Suboptimum Receiver for Data Transmission over Dispersive Channels Using Two-Dimensional Signal Constellations', *IEEE Trans. Commun.*, Vol. COM-35, No. 3, pp. 336-339, Mar. 1987.

[72] R.W. Wood, 'Viterbi Detection of Miller-Squared Code on a Tape Channel', Int. Conf. on Video, Audio and Data Recording, IERE Conf. Proc. 54, Southampton, England, Apr. 20-23, 1982, pp. 333-344.

[73] R.W. Wood and D.A. Petersen, 'Viterbi Detection of Class IV Partial Response on a Magnetic Recording Channel', *IEEE Trans. Commun.*, Vol. COM-34, no. 5, pp. 454-461, May 1986.

[74] R.W. Wood, 'New Detector for 1,k Codes Equalized to Class II Partial Response', *IEEE Trans. Magn.*, Vol. MAG-25, pp. 4075-4077, Sept. 1989.

[75] R.W. Wood, 'Magnetic Megabits', *IEEE Spectrum*, pp. 32-38, May 1990.

[76] R.W. Wood, 'Enhanced Decision Feedback Equalization', *IEEE Trans. Magn.*, Vol. MAG-26, No. 5, pp. 2178-2180, Sept. 1990.

[77] J.M. Wozencraft and I.M. Jacobs, *Principles of Communication Engineering*. New York: Wiley, 1965.

[78] A. Yamashita, T. Nakamura, Y. Tozawa, T. Katoh and M. Moriwake, 'A New Path Memory for Viterbi Decoders', Proc. GLOBECOM'87, pp. 2076-2079.

8

ADAPTIVE RECEPTION

8.1 INTRODUCTION

The use of adaptation techniques enables a data receiver to deal with variations of the channel parameters. These come in two types:

1. *Piece-wise variations*: here an ensemble of fixed channels is available and it is *a priori* unknown to which one the receiver will be connected. Examples include narrowband ISDN, where subscribers are located at different distances from the local telephone exchange, and digital recording channels with piece-wise differences as a result of mechanical tolerances, different bias currents or use of different brands of recording media. Variations of this type are all quasi-static, and in principle adaptation needs to occur only once.

2. *Temporal variations*: These occur when a single channel varies in time. The rate of variation can be slow, such as for thermal changes of an ISDN connection. Conversely, very rapid variations arise, for example, in high-density digital magnetic recording due to fluctuations of head-to-media spacing.

Receivers that are aimed at dealing with slow or piece-wise variations are sometimes called *automatic*. Adaptation to piece-wise variations is sometimes called *tuning*. For adaptation techniques to be effective, the rate of adaptation must be high enough to enable the bulk of all variations to be tracked. Unfortunately, fast adaptation is inevitably costly in terms of both complexity and steady-state performance, as we will see. For this reason one often settles for the lowest acceptable rate of adaptation.

Basic to the design of an adaptive receiver is *prior knowledge* about the system parameters. If it is known, for example, that only the gain of the channel can vary, then a circuit for automatic gain control (AGC) can constitute the only adaptive part of the receiver.[1] It is good engineering practice to fit the adaptive part of the receiver as a

[1]The timing recovery recovery subsystem is, strictly speaking, also adaptive, even though it is rarely if ever treated under this heading. This is perhaps due to its special nature, and also to its crucial importance: a fixed data receiver with timing recovery may work at least part of the time, but an adaptive one without it is permanently useless.

straitjacket around the uncertain part of the system characteristics. This leaves a minimum number of parameters to be adapted across the smallest possible range. Any additional adaptation flexibility may well introduce more problems than it can solve (see Section 8.2.4).

Once the adaptive part of the receiver is defined, an adaptation technique must be devised which extracts misadjustment information from signals internal to the receiver, and which transforms this information into control signals. A careful choice is needed to ensure adequate performance while minimizing complexity. Key issues in this tradeoff are the quality criterion that the technique should optimize, the signals on which it operates, the manner in which they are combined, and the numerical adaptation parameters.

Adaptation techniques have been and still are a subject of active research, as witnessed by a variety of books [2, 4, 19, 38, 41, 89]. In most of this work, the classical least mean-square (LMS) algorithm is taken as a starting point, and ample space is reserved for more complicated algorithms that are aimed at improving upon LMS in certain respects. This bias towards 'advanced' algorithms hardly reflects common engineering practice in data receivers, where LMS-like approaches are often preferred because of their simplicity and robustness. For this reason we do not move beyond these approaches in this chapter. Instead, we venture into directions that are less usual for textbooks yet relevant to data receivers. These include automatic gain control, continuous-time and non-data-aided forms of adaptation, and the use of nonlinearities to improve loop properties and lower strains on circuitry.

Throughout the chapter, adaptation schemes are primarily regarded as control loops that seek to force a measure of the parameter misadjustment towards zero. This non-algorithmic viewpoint remains valid in the presence of nonlinearities, and is equally applicable to continuous-time and discrete-time loops. To stress the latter fact, subscripts or time arguments are used only when it is necessary to make the domain of a signal explicit. In the second half of the chapter we cannot avoid the use of vectors and matrices, and we assume an elementary familiarity of the reader with matrix theory (see, for example, [76] for an introduction to this field).

The chapter is organized as follows. We start with an overview of structures of adaptive receivers and of the adaptive filters that they can contain. We then discuss automatic gain control. This function is significant enough to merit a separate treatment, and yields the simplest possible example of an adaptive filter. Many of the concepts and properties that are central to adaptive filtering are easily developed for automatic gain control, and carry over directly to more general adaptation problems. These are discussed in sections on adaptive equalization and adaptive detection, which cover non-data-aided slope adaptation as well more usual data-aided adaptation schemes of the zero-forcing (ZF) and minimum mean-square error (MMSE) type. The treatment of ZF adaptation is quite extensive and covers the classical partial zero-forcing (PZF) strategy as well as the more powerful minimum mean-square ISI (MMSI) approach. A final section presents a worked example of an automatic equalizer for high-bit-rate digital video recording. In this preview we should also refer ahead to Chapter 10, where we will see

that timing recovery may be based on close relatives of the ZF and MMSE adaptation loops. Many of the insights gained there are also applicable to adaptation.

8.2 STRUCTURE OF ADAPTIVE RECEIVERS

Previous chapters have portrayed a variety of baseband data receivers. A common trait of these receivers is that they all contain some form of equalizer or prefilter, along with a detector of some type. Both parts can be adaptive. We first consider the equalizer.

8.2.1 Adaptive equalization

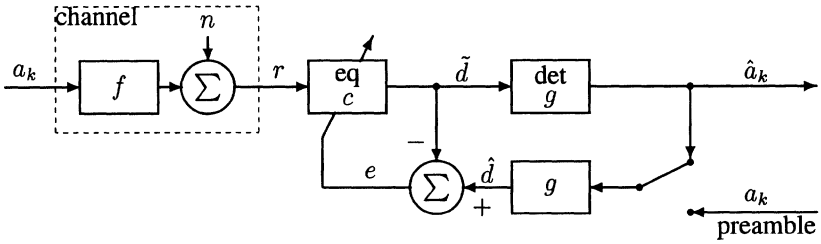


Figure 8.1: Data transmission system with adaptive equalizer.

The channel in Fig. 8.1 has a symbol response f that is *a priori* unknown. The detector is meant for a given system response g. The equalizer serves, roughly speaking, to transform f into g. To this end, its impulse response c is adapted under the control of an error signal e. This signal is a measure of the difference between the actual equalizer output $\tilde{d}$ and a desired output $d \triangleq (a * g)$ that would arise if noise were absent and the equalizer were perfectly adjusted. A potential problem with this approach is that d is, in general, not known to the receiver. However, an estimate $\hat{d}$ of d may be produced by applying bit decisions $\hat{a}_k$ to a filter with impulse reponse g. This will work well whenever bit errors are rare. Unfortunately, decisions tend to be incorrect at transmission startup, especially when the initial equalizer settings are far off. In that case adaptation based on $\hat{d}_k$ may still work, be it more slowly. A safer approach is to precede the actual data by a known preamble a_k. This enables the receiver to generate $d = (a * g)$ without error, thus permitting fast initial convergence. Chapter 9 uses this same approach as a means to speed up clock acquisition. As a rule, clock acquisition is best done with spectrally concentrated data a_k. By contrast, a wide spectrum is typically desirable for adaptation so as to define the equalizer transfer function at all relevant frequencies (see Example 8.21). To serve these distinct needs, preambles in practical transmission systems often contain separate patterns for clock run-in and equalizer adaptation.

For channels whose impulse response f is highly unpredictable, the equalizer in Fig. 8.1 will need many adjustable parameters. Conversely, only one or two may do when f is coarsely known *a priori*.

Example 8.1 Wideband transmission systems tend to suffer mainly from gain variations. These may be compensated by a circuit for automatic gain control (AGC). Digital magnetic recording channels are often mainly hampered by gain fluctuations and an unpredictable loss at high frequencies (see Chapter 2). These variations can be handled jointly by an AGC in conjunction with a *slope equalizer*. Such an equalizer has a variable degree of spectral roll-on or roll-off. It is also possible to combine both tasks into one equalizer structure (see Section 8.2.5).

Adaptation in Fig. 8.1 involves the data a_k and is therefore *data-aided*. Practical equalizer adaptation schemes are mostly of this type. A notable exception concerns equalizer parameters that are related to signal amplitude, such as gain and slope. Control information for such parameters can also be derived in a *non-data-aided* manner (see Sections 8.3 and 8.4).

8.2.2 Adaptive detection

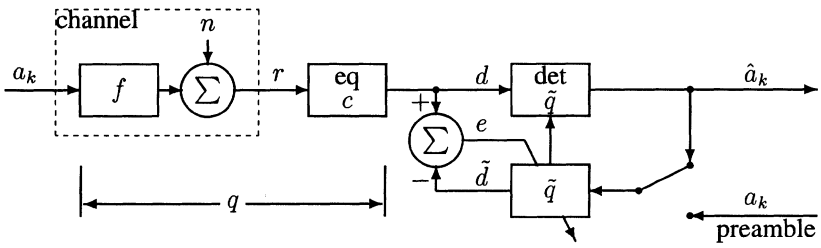


Figure 8.2: Data transmission system with adaptive detector.

In Fig. 8.2, a fixed equalizer or prefilter has impulse response c and transforms the symbol response f of the channel into an overall system response $q \triangleq (f * c)$. Since f is *a priori* unknown, so is q. The detector is meant for a system impulse response $\tilde{q}$ that can be adjusted under the control of an adaptation algorithm. Ideally $\tilde{q} = q$. To approach this ideal, the regenerated data $\hat{a}_k$ is applied to a filter with impulse response $\tilde{q}$, whose output $\tilde{d}$ is subtracted from the detector input d. In the absence of decision errors, the resulting error e is a measure of the resemblance of $\tilde{q}$ and q, and will only contain filtered channel noise when $\tilde{q} = q$. Adaptation of $\tilde{q}$ is based on e. Just as in Fig. 8.1, a preamble a_k can be used to overcome startup problems.

Compared to Fig. 8.1, the desired signal d and its approximation $\tilde{d}$ have changed roles, and the error has changed polarity. This reflects the different nature of the adaptation problem: in Fig. 8.1 the objective is to *undo* channel imperfections, while the aim in Fig. 8.2 is to *mimic* them. In the adaptation literature, the corresponding problem categories are called *signal correction* and *signal estimation*, respectively [16]. While e in Fig. 8.2 forms a valid basis for detector adaptation, it is not necessarily a good indicator of receiver performance.

Example 8.2 In digital magnetic recording, dropouts cause the data component of the replay signal r to fade. During dropouts, the noise component of r also tends to become smaller, but

usually not enough for signal-to-noise ratio to remain unaffected. If the adaptation scheme functions properly then e will consist mainly of noise and hence decrease during the dropout. Even so, detector performance will degrade. Thus e is not a valid indicator of receiver performance and cannot be a good basis for *joint* adaptation of equalizer and detector. Normalization of e may help (Problems 8.1, 8.2 and 8.3; see also Section 8.2.3).

Exercise 8.1 Argue that e_k is a good performance indicator in the system of Fig. 8.1.

In practice, the error e may already be available within the detector, or may be derived in a simpler fashion than in Fig. 8.2. We give three examples.

Example 8.3 In certain Viterbi detectors, an error sequence e_k^{ij} akin to the sequence e_k of Fig. 8.2 is formed for every possible extension j of every survivor i (see Chapter 7). A global error sequence e_k for adaptation may be formed by selecting at any instant k the error sequence e_k^{ij} that corresponds to the most likely extended survivor. This approach is simple but has certain disadvantages that are, however, easily resolved (see Problem 8.4).

Example 8.4 In the partial-response system of Fig. 8.3, a data sequence $a_k \in \{-1, 1\}$ is precoded and applied to a Class IV channel with *a priori* unknown gain q_0. The output $\tilde{b}_k$ of the channel is a noisy and scaled version of the partial-response sequence $b_k \in \{-2, 0, 2\}$. A ternary bit-by-bit detector produces decisions $\hat{b}_k \in \{-2, 0, 2\}$ with respect to b_k, which are converted into decisions $\hat{a}_k$ about a_k by means of a memoryless inverse mapping (MIM). For an explanation of precoder and MIM we refer back to Chapter 6.

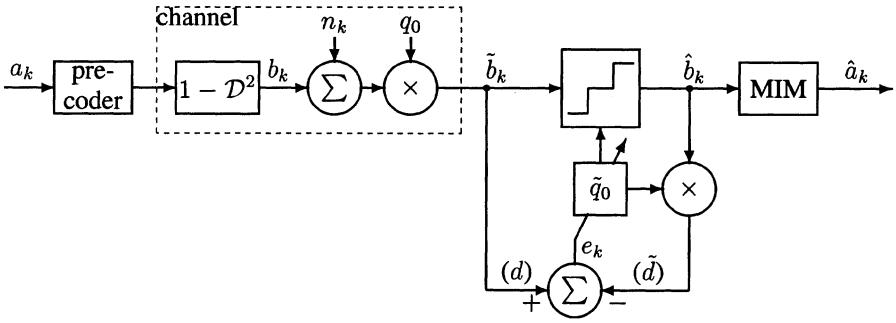


Figure 8.3: Partial-response transmission system with adaptive bit-by-bit detector.

Possible detector input levels in the absence of noise are 0 and $\pm 2q_0$. Ideally, detection thresholds are halfway between these levels, i.e. at $\pm q_0$. The actual levels are at $\pm\tilde{q}_0$; $\tilde{q}_0$ is updated under the control of an error sequence e_k. Just as in Fig. 8.2, this sequence is the difference of the actual detector input (d) and a synthetic one $(\tilde{d})$ that would emerge if noise were absent and the channel would have gain $\tilde{q}_0$. The sequence $\tilde{d}$ is conveniently produced by scaling the decisions $\hat{b}_k \in \{-2, 0, 2\}$ with the threshold amplitude $\tilde{q}_0$. This is easier than in Fig. 8.2, where binary decisions would have to be filtered by a scaled Class IV response. For a description of how $\tilde{q}_0$ can be adapted we refer ahead to Section 8.8.1.

This example shows that it is not always necessary to use an AGC in order to overcome gain variations. For the sake of completeness we mention that the Ferguson detector admits a very similar threshold adaptation technique (see Section 7.6.4).

Exercise 8.2 Mention advantages and disadvantages of threshold adaptation relative to automatic gain control. Consider in particular implementation issues such as dynamic ranges of signals, and behavior in the presence of deep fades or dropouts.

Example 8.5 In the feedback detector of Fig. 8.4, a feedback filter (FBF) with impulse response p_k serves to cancel trailing ISI. Ideally, p_k is a replica of the tail of the system response q_k up until the detector, i.e. $p_k = q_k$ for $k \geq 1$. We recall from Chapter 6 that application of this detector only makes sense if pre-cursive ISI is negligible (i.e. if $q_k \simeq 0$ for $k < 0$), and if q_k is not too heavy-tailed.

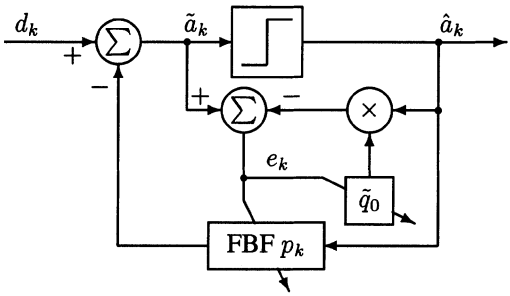


Figure 8.4: Adaptive feedback detector.

The FBF impulse response p_k is the tail of the assumed system response $\tilde{q}_k$, i.e. $\tilde{q}_k = p_k$ for $k \geq 1$. The FBF output is the assumed trailing ISI, which is subtracted from the actual ISI, as present in the detector input d. In order to arrive at an error sequence e_k equivalent to that in Fig. 8.2, it remains to subtract the assumed influence of the current symbol a_k. This involves a scaling of the decision $\hat{a}_k$ by an estimate $\tilde{q}_0$ of q_0. This estimate is also updated under the control of e_k. Just as in Fig. 8.3, $\tilde{q}_0$ can serve as a reference for nonzero detection thresholds. These are only needed when data a_k is nonbinary. It is not difficult to verify that the error e_k in Fig. 8.4 is equivalent to that of Fig. 8.2.

8.2.3 Joint equalizer and detector adaptation

Joint adaptation of equalizer and detector necessitates an error e that is a reasonable indicator of system performance. One option is to normalize the system gain to unity. By way of illustration we consider the decision feedback equalizer (DFE), which is just the cascade of a prefilter and the feedback detector of Fig. 8.4. The error e_k in Fig. 8.4 becomes a good indicator of DFE performance if the data amplitude at the input of the slicer is fixed, say at $q_0 = 1$. In that case the predetection signal-to-noise ratio will always become larger if e_k gets smaller and vice versa. This suggests that we fix $\tilde{q}_0$ in Fig. 8.4 at $\tilde{q}_0 = 1$, and adapt only the prefilter and feedback filter so as to minimize the power of e_k. The prefilter will provide gain control such that q_0 settles approximately at 1, and the filters will jointly converge towards the MMSE solution that we have determined in Chapter 6. A direct consequence is that the prefilter will approach a canonical whitened matched filter (WMF), which is also the optimum prefilter for the full-fledged VD (see Chapter 3). This suggests a simple scheme for joint adaptation of a preequal-

izer and VD [56]: the VD is connected in parallel to the feedback detector, whose FBF coefficients along with the fixed value $q_0 = 1$ determine the system response assumed by the VD. Disadvantage of this scheme is that the prefilter may not approach a WMF unless the feedback filter is long. While this renders the full-fledged VD unduly complicated, the VD with sequence feedback of Section 7.7 remains perfectly applicable. For that VD, the parallel feedback detector can, in fact, be omitted by making use of signals that are available within the VD [5]. VD complexity can also be restricted by constraining the duration of the system response (see Section 7.6). A technique for joint adaptation of preequalizer and desired system response (DSR) is described in [22]; alternative criteria for selecting the DSR may be found, for example, in [3, 24].

The small list of techniques for joint adaptation prefilter and detector further includes the one of [80]. Since that technique pertains to a full-fledged VD, complexity tends to be high.

8.2.4 How much adaptivity is needed?

At a first glance it may seem wise to maximize the amount of receiver adaptivity. This is, however, generally untrue.

Example 8.6 If the equalizer contains a transversal filter with more taps than strictly needed, then the position of the 'central' tap becomes ill-defined for decision-directed adaptation loops of the MMSE type (see Section 8.6). As a result, local optima emerge at which the loop might get stuck. Performance losses increase as the central tap shifts further away from its ideal position. If a long equalizer is unavoidable (as a consequence, for example, of the fact that the channel is highly unpredictable), then measures are needed to define the central tap position. One technique involves a gradient-directed search [68]. A simpler approach is to restrict the non-central taps to have a magnitude smaller than the central one [85]. This technique, like other ones, has its limitations.

Example 8.7 For decision-directed adaptation loops of the zero-forcing type (see Section 8.5), the steady-state coefficient settings are, in general, unambiguous if decision errors are rare. In particular for transversal filters, the central tap position is well-defined. Even so, the equalizer may get stuck at a totally wrong solution -causing mostly wrong decisions- if initial coefficient settings are so far off that many bit errors occur. To avoid this problem, a data-aided startup procedure must precede the decision-directed mode of adaptation whenever the channel is insufficiently predictable.

Example 8.8 For a transversal equalizer, MMSE tap settings tend to become ill-defined and convergence properties tend to degrade when the equalizer has more taps than strictly needed, especially when the equalizer input is not spectrally rich (see Example 8.21).

Example 8.9 Fractionally-spaced equalizers have a fundamental interval that extends beyond the Nyquist frequency. If little or no signal is present in parts of this interval, then the optimum equalizer transfer function becomes ill-defined there. As a result, very large transfer magnitudes may emerge for adaptation loops of the MMSE type, and equalizer coefficients will overflow or saturate. This problem is discussed in more detail in Section 8.6.5.

Example 8.10 Some adaptive equalizer and detector structures are flexible enough to be able to compensate for small time delays. Examples include fractionally-spaced equalizers, the analog equalizer of Fig. 8.7 if the first derivative is used, and the adaptive Viterbi detector. If this flexibility is present then adaptation may enter the realm of timing recovery, and severe interaction may occur. If the flexibility cannot be missed (it might be essential, say, to deal with phase distortion), then alternative measures are needed. These are typically *ad hoc* and specific for the equalizer structure and timing-recovery technique at hand (see [69], [81] and [33] for examples related to fractionally-spaced equalization).

Example 8.11 Functionalities of equalizer and detector may exhibit overlap. For example, an AGC in the equalizer is equivalent to an adaptive detector threshold as in Examples 8.4 and 8.5. If both are present, then the AGC gain could settle at an arbitrary value because the threshold tracks arbitrary system gains. As a result, detector input signals might either overflow/saturate or sink away in quantization/circuit noise. This is clearly unacceptable. The solution, in this case, is to omit either the AGC or the threshold adaptation. Other functional overlaps may be harder to eliminate.

These examples underscore the importance of a *minimum* amount of adaptivity. A further argument is that any adaptive parameter introduces *gradient noise* that lowers predetection signal-to-noise ratios to a certain extent (see Section 8.5.3). Thus, adaptivity is best viewed as a necessary evil that should be applied with care and restraint.

8.2.5 Adaptive filter structures

Transversal filters These filters take center stage in the adaptation literature. They contain a delay line with N stages whose outputs are tapped and summed via a linear combiner (Fig. 8.5). The tap values $c_0, \ldots, c_{N-1}$ define the impulse response and transfer function of the filter.

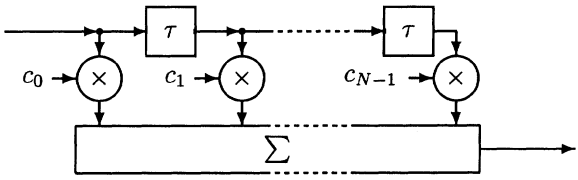


Figure 8.5: Transversal equalizer with N taps and tap spacing τ.

A T-spaced equalizer (for which the tap spacing τ amounts to one symbol interval T) cannot suppress out-of-band noise. A prefilter is needed to suppress this noise. Fractionally-spaced equalizers (for which $\tau < T$) can suppress out-of-band noise to a certain degree, and less or no prefiltering is needed. In digital transversal equalizers, τ must be an integer fraction or rational submultiple of T, such as $T/2$ or $3T/4$.

Filters that form part of adaptive detectors are also often of the transversal type; the standard tap spacing is T seconds. The filter input is typically a binary signal (compare Figs. 8.2 and 8.4) so that digital multiplications degenerate into simple sign reversals.

Table look-up filters [40] These filters are tailored to binary input sequences (Fig. 8.6). The binary data stream is clocked into a shift register. At any symbol interval T, the N most recent bits together address a table, whose contents specify the filter output. Advantages are twofold:

1. Filter outputs are looked up rather than computed. This facilitates high-speed operation.
2. Nonlinear input-output relations can be handled. This allows the detector to account for channel nonlinearities. These can be significant, for example, in disk drives with thin-film metallic media.

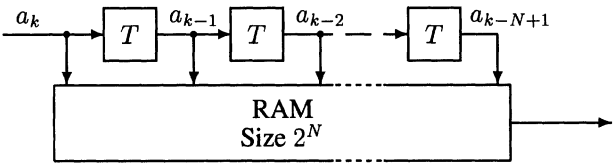


Figure 8.6: Table look-up filter.

Main disadvantage is that the required amount of storage grows exponentially with N. This restricts application to memory lengths N of up to 5 or 10. Even then, the number of parameters that must be adapted (2^N) exceeds that for a transversal filter (N). The result is increased gradient noise for a given tracking speed, or a lower speed for the same gradient noise (see section 8.8.3).

Problems 8.44 and 8.45 consider mixtures of transversal and table look-up filters that combine the advantages of both structures while deemphasizing the disadvantages.

Other digital structures These include various IIR filter types [42], Quadratic Residue (QR) arrays [54], lattice filters [25], frequency-domain adaptive filters [17], and a memory-accumulator architecture [18, 11]. These structures have yet to find widespread application.

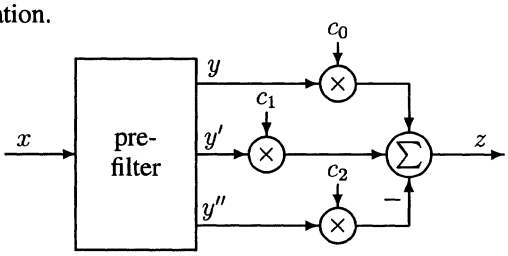


Figure 8.7: Analog derivative equalizer.

Analog structures Analog filters often have only a few adjustable parameters (among the exceptions are analog transversal and Laguerre filters [85]). Examples include slope equalizers for cables with only one or two adjustable parameters [77], a 'cosine' filter for magnetic recording [44], and a derivative equalizer whose origins can be traced back

to work of Schneider [74]. Non-adaptive variants of that equalizer are widely used in digital magnetic recording. A generic form is shown in Fig. 8.7.

The main output y of a low-pass prefilter is combined with its first and second derivatives y' and y''. The weighing factors c_0, c_1 and c_2 permit compensation of gain-, phase- and slope-errors. Except for an inversion the y'' output is in phase with the main output y. The inversion is undone by a second one in the summing block. Nominally $c_1 = 0$ and $c_0 = c_2$. For these coefficient settings, the overall equalizer is designed to produce a prescribed output (e.g. Class IV) under nominal channel conditions. The twofold differentiation in the y'' branch boosts high frequencies. The high-frequency portion of z is, therefore, mainly induced via the y'' branch while the low-frequency portion stems mainly from the y branch. Density or bandwidth variations of the channel cause a spectral unbalance between the y and y'' outputs, which can be compensated for by changing the ratio c_0/c_2. This is a form of slope equalization. An advantage of this technique is that it does not affect the phase characteristics of the equalizer. Gain variations can be compensated for by changing c_0 and c_2 equally. The coefficient c_1 does affect the phase characteristics of the equalizer and this permits a first-order compensation of variations of the phase characteristics of the channel relative to the nominal ones. The three 'knobs' c_0, c_1 and c_2 together permit compensation of a substantial portion of the dynamic parameter variations of practical recording channels. Control of these knobs is straightforward (see Example 8.20) and without many of the complications that more flexible filters would cause (cf. Section 8.2.4).

8.3 AUTOMATIC GAIN CONTROL

In digital transmission and recording systems, received signal levels may vary widely.

Example 8.12 In ISDN loops, signal levels easily range across 30 dB for cable lengths up to 5 km and a typical attenuation of 6 dB/km. In digital magnetic recording systems, replay levels may vary across more than 20 dB due to combined spread of media, heads and preamplifier ICs.

To handle variations of this type, the receiver front-end often contains a circuit for automatic gain control. It is common practice to designate by AGC the circuit that realizes automatic gain control. The AGC is typically an analog circuit. To avoid overload and to restrict power dissipation along with circuit noise, it may consist of several stages, especially when the input signal has a large dynamic range. Signal levels behind the AGC are more or less constant. This minimizes dynamic ranges in the remainder of the receiver (or, equivalently, word lengths for a digital implementation). Possible AGC topologies are manifold. We first consider non-data-aided topologies.

8.3.1 Non-data-aided automatic gain control

In Fig. 8.8, the received signal r of power P_r is scaled by a factor $c = \sqrt{P_d/\tilde{P}_r}$, where P_d represents the desired power of x and $\tilde{P}_r$ is an estimate of the power of r.

The scaled signal x has power $P_x = c^2 P_r = P_d[P_r/\tilde{P}_r]$. Hence P_x will approach P_d as long as the power of r is estimated sufficiently accurately. To compensate for the delay in the control path (which is essentially due to the LPF), a properly chosen delay τ may be added in the signal path.

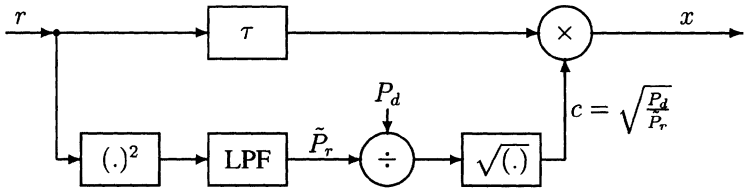


Figure 8.8: Non-data-aided open-loop AGC.

The AGC of Fig. 8.8 is of the open-loop type. It is non-data-aided (NDA) because it uses no knowledge of the transmitted data symbols a_k. The involved operations (power estimation, division, formation of square root, multiplication) are cumbersome when done digitally but are quite manageable in the analog domain. In spite of its intuitive appeal, the circuit of Fig. 8.8 has several disadvantages.

1. For optimum transmission quality, it is desired to normalize not r but only its data component so as to arrive at a well-defined system gain. This is intrinsically impossible with the AGC of Fig. 8.8 because it is *a priori* unknown how much noise r will contain. Normalization of the power of r inevitably induces a gain bias that gets larger as noise increases (Section 8.3.6, Problem 8.5).

2. Accurate estimation of the power of r tends to take much time (see Problem 8.6). This renders the AGC either slow or inaccurate beyond just a bias (Problem 8.7). This may not be a problem when signal levels vary only slowly or when the AGC is only meant to normalize signal levels coarsely.

3. Dynamic ranges in the control part of the circuit tend to be large. For example, if r varies across 30 dB, then $\tilde{P}_r$ varies across an unwieldy 60 dB. (Some relief is possible. For example, the range can be reduced to that of r by basing control on the envelope (modulus) rather than power of r.)

4. Any inaccuracy in the computation of c directly enters x. High sensitivity to parameter variations in the control path is a principal reason why open-loop AGCs are not often employed.

The *closed-loop* AGC of Fig. 8.9 alleviates the latter problem. This loop is still non-data-aided.

The scaled signal x has power P_x. The square x^2 is compared with the desired power P_d. The error η is weighed by a factor $\mu \geq 0$ and integrated. The integrator output c serves as the AGC gain. Provided that the feedback loop is stable, the integrator output will remain finite, i.e. its input $\mu\eta$ will be driven toward zero. This means that c will

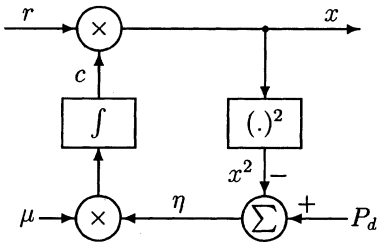


Figure 8.9: Non-data-aided closed-loop AGC.

settle at a value which ensures that $x^2 = P_d$ upon average, i.e. that $P_x = P_d$. Low-pass filtering is performed by the loop; square roots as in Fig. 8.8 need not be taken. Implementation inaccuracies play a much smaller role than in Fig. 8.8 because they are largely compensated for by the feedback loop. An exception is formed by DC offsets, which are misinterpreted by the loop as a nonzero error η. The result will be a steady-state gain bias (see Section 8.3.7 and Problem 8.8).

8.3.2 Exponential gain control

Properties of the AGC of Fig. 8.9 are analyzed in Appendix 8A and Problem 8.9. The analysis is compounded by the squaring operation in the loop. However, a mere glance at Fig. 8.9 reveals that loop gain must be proportional to r. This means that the loop will be slow at low input signal levels and fast or even unstable at high levels. It is clearly undesirable that loop properties depend directly on a signal level that can vary widely. The classical solution [84, 63] involves a replacement of the linear voltage-controlled amplifier (VCA) in 8.9 by an exponential one. To validate this approach we first note that the steady-state gain c in Fig. 8.9 is inversely proportional to the signal level of r. Multiplication of the loop gain by c will, therefore, compensate for the loop-gain increment that would otherwise occur. This observation suggests the control structure of Fig. 8.10.a. The integrator input z is the derivative of the output c so that the structure obeys the differential equation $dc/dt = x.c$.

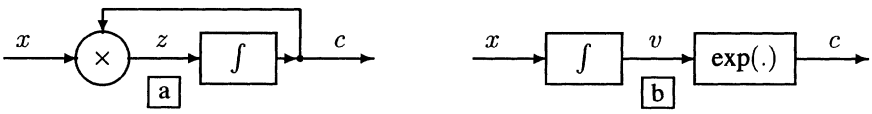


Figure 8.10: Two equivalent control structures for stabilization of the loop gain of the closed-loop non-data-aided AGC in the face of amplitude variations of the input signal.

In the alternative structure of Fig. 8.10.b, the integrator is followed by an exponential amplifier. One readily verifies that this structure also obeys the differential equation $dc/dt = x.c$, i.e. both structures are equivalent. The loop gain in Fig. 8.9, therefore, no longer depends on the input signal level if we insert an exponential amplifier between

the integrator and the linear VCA. The combination of amplifier and VCA is just an exponential VCA with *gain control characteristic* $c(v) = \exp\{v\}$ (Fig. 8.11).

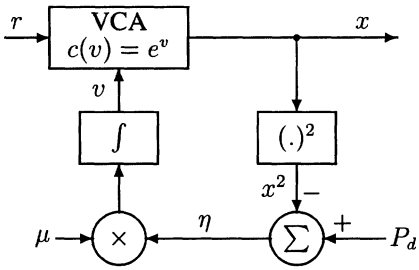


Figure 8.11: Non-data-aided closed-loop AGC with exponential voltage-controlled amplifier (VCA).

The gain in dB is a now linear function of the control voltage. The loop still acts to drive $\eta = x^2 - P_d$, upon average, toward zero. In steady state, x^2/P_d will fluctuate about 1. In that case we may apply the approximation $\ln[x^2/P_d] \simeq [x^2/P_d] - 1$. Correspondingly $\eta = P_d([x^2/P_d] - 1) \simeq P_d \ln[x^2/P_d]$. Upon realizing that $x^2 = r^2 \exp\{2v\}$ it follows that $\eta \simeq P_d[2v + \ln r^2 - \ln P_d]$. This leads to the model of Fig. 8.12.

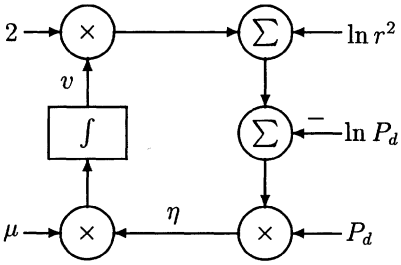


Figure 8.12: Linearized loop model corresponding to the AGC of Fig. 8.11.

The loop of Fig. 8.12 is fully linear and is of the first-order type. Loop properties are determined by the open-loop gain $K_t = 2P_d\mu$, and are indeed no longer affected by the amplitude of r. The control voltage v (and thereby the VCA gain in dB) converges exponentially with a time constant $1/K_t$. As a result, the VCA gain varies in a doubly exponential fashion.

The above derivation assumes that $x^2 \simeq P_d$. This is, in general, only true in steady state. It was shown by Ohlson [63] that the results of the analysis remain valid under a much wider range of conditions. We refer to [57, ch.7] for a more detailed discussion, including an assessment of the effect of noise. This effect can also be analyzed in a convenient manner by extending the analysis of Appendix 8B to the AGC at hand.

The exponential VCA is the only one that causes AGC properties to be independent of the level of the input signal. In practice the exponential characteristic may be approximated by a function of the form $c(v) = e^{-\alpha v} + \beta$ or $c(v) = (2 + \alpha v)/(2 - \alpha v)$. Both

are easily mechanized with analog circuitry. The squarer and subtractor in Figs. 8.9 and 8.11 together constitute a *square-law detector* of gain errors. Of comparable interest is the *envelope detector*, in which a gain-error estimate η is formed by subtracting the envelope $|x|$ from a desired value $|x|_d$.

8.3.3 Data-aided Automatic Gain Control

The first two disadvantages on Page 383 are inevitable when control is based on a 'noncoherent' signal feature such as its power or envelope. Relief is possible by invoking 'coherent' information, such as decisions about to the transmitted data (Fig. 8.13).

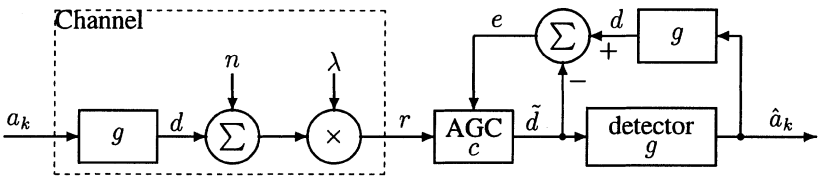


Figure 8.13: Transmission system with data-aided AGC.

A detector produces decisions $\hat{a}_k$ with respect to a_k that are assumed to be correct. The detector is designed for a target response g. The actual channel has symbol response λg and additive noise λn, where λ is the *a priori* unknown gain of the channel that the AGC is desired to compensate for.[2] The detector knows both $\hat{a}_k$ and g so that a desired signal $d \triangleq (a * g)$ can be formed that would have been received if noise and gain variations were absent. The AGC output $\tilde{d} = cr$ is desired to approach d, i.e. the error $e \triangleq d - \tilde{d}$ should be small. We will discuss two approaches that accomplish this.

8.3.4 Zero-forcing gain control

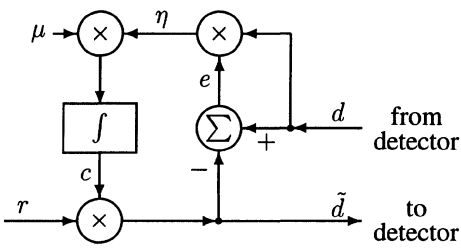


Figure 8.14: Zero-forcing data-aided AGC.

[2]In practice, noise in the received signal is often only partly dependent on λ, and addition of n after multiplication by λ may be a more natural model. Consequences of this change to the results of this section are straightforward and are left for the reader to elaborate.

Here the crosscorrelation of e and d serves as a basis for adaptation of c (Fig. 8.14). For simpicity the VCA is as yet linear. When the loop is stable, the integrator output will be finite, i.e. its input cannot contain any DC. This means that the loop will act to drive $\eta = e.d$, upon average, toward zero. For mathematical convenience we restrict attention to ensemble averages here. We will analyze under what condition $E[\eta]$ vanishes. To this end, we note that $e = d - \tilde{d} = d - rc = d - (d+n)\lambda c = d(1-\lambda c) - n\lambda c$. Correspondingly

$$\eta = ed = d^2(1 - \lambda c) - \nu, \tag{8.1}$$

where $\nu \triangleq dn\lambda c$ is the noise component of η. This noise has zero mean. Hence $E[\eta] = P_d(1 - \lambda c)$ where $P_d \triangleq E[d^2]$ represents the power of d. This expected value vanishes when c assumes it ideal value $c^* \triangleq 1/\lambda$. As desired, additive noise does not come into play. This overcomes the first disadvantage mentioned on page 383. The nomenclature of this control technique derives from the fact that it forces any gain mismatch to zero. This is similar to zero-forcing equalization, where all intersymbol interference within a certain span is eliminated.

Loop behavior Application of (8.1) leads to the simplified model of Fig. 8.15. The AGC gain c is compared with its ideal value $c^* = 1/\lambda$. The error η is a noisy and randomly amplified version of the gain mismatch $v \triangleq c^* - c$. The total open-loop gain is $K_t \triangleq \lambda d^2 \mu$. Loop properties can be controlled with the adaptation parameter μ. The proportionality of K_t to λ causes the loop to be slow at low signal levels and fast or even unstable at high levels. To overcome this problem we may use an exponential VCA as in Fig. 8.11. The development is entirely similar and is the subject of Problem 8.12. Another option is to insert a quadratic nonlinearity in the control loop (Problem 8.11). This lowers dependence of K_t on λ but does not eliminate it altogether.

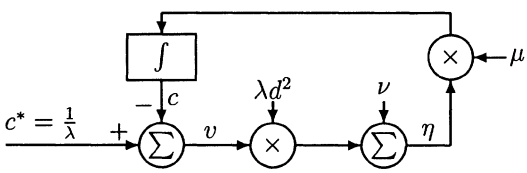


Figure 8.15: Equivalent model of the loop of Fig. 8.14.

Binary desired signal Since d is random, K_t is also random. The only exception occurs when d is binary and the data alphabet is centered about the origin ($d = 0$). In that case $d^2 = P_d$ at any given instant, and the loop gain $K_t = \lambda P_d \mu$ becomes deterministic. This case is sufficiently instructive to be considered first. Here we arrive at the loop model of Fig. 8.16 upon referring noise ν back to the reference terminal.

The signal s_k at the reference terminal contains a component $c^* = 1/\lambda$ that the loop is desired to track, and noise $u \triangleq \nu/(\lambda P_d) = dnc/P_d$ that it should reject. In steady state we have $c \simeq c^* = 1/\lambda$, i.e. u_k has variance $\sigma_u^2 \simeq \sigma_n^2/(\lambda^2 P_d)$. The lowest-frequency components of u can intrinsically not be distinguished from those of c^* and

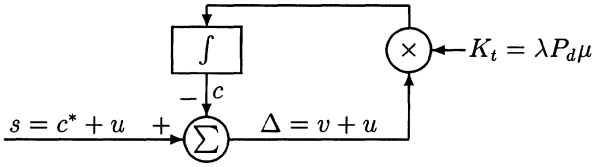


Figure 8.16: Simplified loop model for discrete-time data-aided AGC with a binary desired signal d.

will also be tracked. Tracking and noise rejection properties are governed by the closed-loop transfer functions $G_c \triangleq C/S$ and $G_\Delta \triangleq \Delta/S$. These functions are given by

$$G_c(z) = \frac{K_t}{K_t + z - 1} \text{ and } G_\Delta(z) = \frac{z-1}{K_t + z - 1} \tag{8.2}$$

when time is discrete, as we will assume for now. Essentially the same loop model and transfer functions are encountered for the first-order discrete-time PLL (Section 11.3). Major properties are:

Stability: The loop is stable for $0 \leq K_t \leq 2$.

Rate of convergence: In the practical event that $K_t \ll 1$, the time constant of adaptation τ amounts approximately to $1/K_t$ symbol intervals. Here and in what follows, we use a dimensionless time constant τ that is normalized in symbol intervals T. The mean-square error converges according to $E[e_k^2] \propto e^{-2k/\tau}$. Hence e_k^2 decreases, upon average, by some

$$-10\log_{10} e^{-2/\tau} \simeq 8.6859/\tau = 8.6859 K_t \text{ dB/iteration.} \tag{8.3}$$

Settling times are in the order of a few time constants. They are sometimes expressed in terms of a parameter ν_{20} that characterizes the number of iterations needed to lower an initial MSE by 20 dB [82]. One readily ascertains that $\nu_{20} \simeq 2\tau$.

Gradient noise: Noise n causes c to fluctuate about its steady-state value $c^* = 1/\lambda$. The gain mismatch $v = c^* - c$ has variance $\sigma_v^2 = \sigma_u^2 B_l$ where

$$B_l \triangleq \int_{-0.5}^{0.5} |G_c(e^{j2\pi\Omega})|^2 d\Omega = \frac{K_t}{2 - K_t} \tag{8.4}$$

is the noise bandwidth of the loop (cf. Fig. 8.16). The steady-state AGC output is

$$\tilde{d}_k = r_k c_k = (d_k + n_k)\lambda(c^* + v_k) \simeq d_k + n_k + \lambda d_k v_k, \tag{8.5}$$

where we have assumed that the noise×noise term $v_k n_k$ can be neglected. The last term in (8.5) represents so-called *gradient noise* due to coefficient fluctuations. This noise is proportional to n_k but is typically not correlated with it. It has variance

$$\sigma_{\text{gr}}^2 = \lambda^2 P_d \sigma_v^2 = \lambda^2 P_d B_l \sigma_u^2 = B_l \sigma_n^2, \tag{8.6}$$

and comes on top of the scaled channel noise $\lambda c^* n_k = n_k$. As a result, the signal-to-noise ratio of the detector input $\tilde{d}_k$ drops by a factor $1 + M$, where

$$M \triangleq \frac{\sigma_{\text{gr}}^2}{\sigma_n^2} \tag{8.7}$$

is the *misadjustment due to gradient noise* [88]. The factor $1+M$ is said to be the *excess mean-square error* (EMSE) due to gradient noise. For the AGC at hand we have

$$M = B_l = \frac{K_t}{2 - K_t} \simeq \frac{K_t}{2} \tag{8.8}$$

where the approximation applies in the usual case that $K_t \ll 2$.

For a negligible impact of gradient noise on transmission quality, EMSE should be kept to a fraction of a dB, and M should be restricted accordingly. For small EMSE we may write

$$\text{EMSE [dB]} = 10 \log_{10}(1 + M) \simeq \frac{10}{\ln 10} M \simeq 4M. \tag{8.9}$$

This linear relationship between EMSE in dB and M makes it easy to relate the two quantities. For example, for EMSE to be at most 0.5 dB, M should not exceed 0.125. Accordingly, the adaptation time constant $\tau = 1/K_t$ cannot be smaller than about $1/(2M) \simeq 4$ symbol intervals.

Efficiency: The tradeoff between EMSE and rate of adaptation (τ) is reflected in the *efficiency factor*

$$\gamma \triangleq \frac{1}{M\tau} = \frac{K_t}{M}, \tag{8.10}$$

which amounts to $2 - K_t$ for the loop at hand. A higher efficiency means faster adaptation for a given misadjustment and vice versa. In practice loop gains will be well within the stability range. Then efficiency is fixed ($\gamma \simeq 2$), and faster adaptation inevitably causes a commensurate increment of M. In [88, Appendix A] it is shown that this efficiency can basically not be improved upon by any other adaptation technique. With (8.9) we see that

$$\text{EMSE [dB]} \simeq \frac{4}{\gamma\tau} = \frac{4K_t}{\gamma}. \tag{8.11}$$

This expression clearly displays the tradeoff between EMSE and adaptation speed.

Exercise 8.3 Assume that the gain λ of the channel can vary across 20 dB. We insist that EMSE be at most 0.1 dB for all permissible gains. What value of μ is needed? Across what range do time constants vary?

In Appendix 8B we show that replacement of the linear VCA in Fig. 8.14 by any reasonable nonlinear one does not change γ. The basic argument is that the nonlinearity only affects the loop gain, of which γ is independent. Hence a given adaptation time constant τ always leads to the same EMSE, irrespective of the VCA type. To determine

EMSE it suffices to find τ. A direct (and typically more complicated) noise analysis is not needed.

Nonbinary desired signal Implications of d being random rather than deterministic are minimal. The loop gain $\mu\lambda d^2$ is now random with mean $E[\mu\lambda d^2] = \mu\lambda P_d = K_t$. The low-pass nature of the loop causes fluctuations about this mean to be suppressed. Hence K_t remains the effective loop gain. Adaptation time constants and EMSE levels do not change. However, short-term averages of d^2 can exceed the long-term average $P_d = E[d^2]$. This causes the loop gain to increase temporarily, possibly to the verge of instability. As a result, the error power e^2 may diverge before the instability limit derived above is reached.

8.3.5 Minimum mean-square error gain control

Here the objective is the same as in minimum mean-square error equalization, viz. to minimize the mean-square value (power) $\varepsilon \triangleq E[e^2]$ of the error signal e in Fig. 8.13. This is a quadratic function of c, with a unique minimum $\hat{c}$. To find $\hat{c}$, we differentiate ε with respect to c and require the derivative to vanish. Clearly $\partial\varepsilon/\partial c = 2E[e\partial e/\partial c] = -2E[er]$. The last step is due to the fact that $e = d - \tilde{d} = d - rc$; since d and r do not depend on c, $\partial e/\partial c = -r$. We see that ε is minimized when e and r have no crosscorrelation. This is exactly what the AGC of Fig. 8.17 strives for.

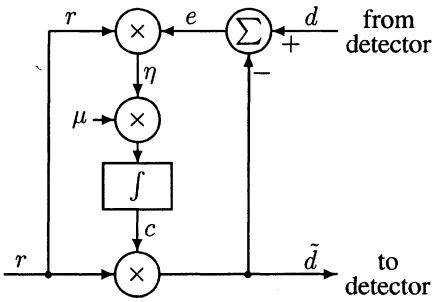


Figure 8.17: Minimum mean-square error (MMSE) data-aided AGC.

Steady-state solution, minimum mean-square error By analogy to the ZF DA AGC, the loop acts to suppress all DC at the input of the integrator, thereby eliminating any cross-correlation between r and e. In that case

$$E[er] = E[(d - rc)r] = E[dr] - cE[r^2] = 0, \tag{8.12}$$

i.e. $\hat{c} = E[dr]/P_r$ where $P_r = E[r^2]$ denotes the power of r. For the channel of Fig. 8.13 one readily ascertains that $E[dr] = \lambda P_d$ and $P_r = \lambda^2(P_d + P_n)$. Corres-

pondingly

$$\hat{c} = \frac{1}{\lambda}\frac{P_d}{P_d + P_n} = \lambda\frac{P_d}{P_r}. \tag{8.13}$$

This shows that $\hat{c}$ is a factor $\beta \triangleq P_d/(Pd + P_n)$ smaller than the ideal value $c^* = 1/\lambda$. For practical signal-to-noise ratios this *signal suppression factor* is close to 1, and the *bias* $1-\beta$ is small. The impact of bias on performance depends heavily on the type of data receiver and on the number of amplitude data levels. For example, a full-response receiver is completely insensitive to bias when data is binary, but becomes increasingly sensitive as the number of data levels increases (Problem 8.17). As a rule, sensitivity also increases as the receiver becomes more powerful (cf. Problem 8.18).

Exercise 8.4 What signal-to-noise ratio P_d/P_n is required for β to be smaller than 1 dB? What bias does this correspond to?

To determine the minimum mean-square error $\varepsilon_{\min}$, we note that $\varepsilon = E[(d-rc)^2] = P_d - 2cE[dr] + c^2P_r$. In the last step it is assumed that c is statistically independent of r. For $c = \hat{c}$ we get

$$\varepsilon_{\min} = P_d - \hat{c}E[dr] = P_d(1 - \lambda\hat{c}) = \frac{\lambda^2 P_d}{P_r}P_n. \tag{8.14}$$

It can be observed that $\varepsilon_{\min} \leq P_n$, with equality holding in the limit as $P_n/P_r \to 0$. By comparison, the ZF DA AGC invariably yields a larger MSE ($E[e_k^2] = P_n$) but no gain bias. Both AGCs behave identically as $P_r \gg P_n$. In practice c will fluctuate about $\hat{c}$. Let $v \triangleq c - \hat{c}$ denote the deviation. Then

$$\varepsilon = \varepsilon_{\min} + P_r v^2. \tag{8.15}$$

This confirms that deviations v will cause ε to grow beyond $\varepsilon_{\min}$. It should be noted that v cannot be interpreted as a gain mismatch now, because $\hat{c}$ exhibits a bias with respect to the ideal gain $c^* = 1/g$.

Adaptation properties The simplified loop model of Fig. 8.18.a emerges upon noting that $\eta = re = r(d - rc) = rd - cr^2$. The loop has gain μr^2. The effective loop gain is $K_t = E[\mu r^2] = \mu P_r = \mu\lambda^2(P_d + P_n)$. Loop properties depend quadratically on λ, as opposed to the linear dependence for the ZF DA AGC. This is clearly undesirable in view of the fact that signal levels at the AGC input may vary widely. Unlike the NDA and ZF DA AGCs, an exponential VCA now provides only partial relief (Problem 8.14). A somewhat less practical VCA characteristic that does cause tracking properties to become independent of λ is studied in Problem 8.15.

In steady state, η will fluctuate randomly about its mean $E[\eta] = 0$. The fluctuations are caused by the randomness of r^2 and rd, and are made explicit in Fig. 8.18.b. The random signals r^2 and rd are replaced by their deterministic averages P_r and $E[rd] = \hat{c}P_r$, respectively. The fluctuations of η are now captured in the form of noise z that is added into the loop. For practical loop bandwidths B_l, only a small low-frequency

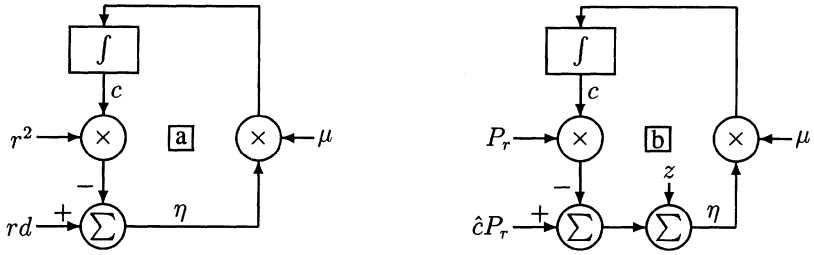


Figure 8.18: Simplified model of MMSE DA AGC (a) and approximation with loop gain and reference signal that are deterministic (b).

portion of z is suppressed by the loop. The rest feeds through into η, i.e.

$$\sigma_z^2 \simeq E[\eta^2] = E[(re)^2] = E[r^2]E[e^2] \simeq P_r \varepsilon_{\min}. \tag{8.16}$$

The second equality is due to the fact that the loop suppresses any correlation between r and e. The second approximation applies because $\varepsilon \simeq \varepsilon_{\min}$ whenever K_t (or, equivalently, B_l) is not too large.

Fig. 8.18.b can be simplified by shifting a common factor P_r across the leftmost summator, and by subsequently referring noise z back to the reference terminal. This leads to Fig. 8.19.

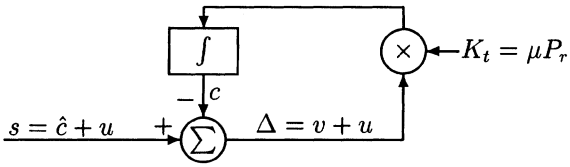


Figure 8.19: Simplified version of Fig. 8.18.b.

The signal s at the reference terminal contains a component $\hat{c}$ that the loop is desired to track, and noise $u \triangleq z_k/P_r$ of variance $\sigma_u^2 = \varepsilon_{\min}/P_r$ that it should reject. Tracking and noise-rejection properties are governed by the same transfer functions as for the ZF DA AGC. In steady state, the error $v = \hat{c} - c$ has variance

$$\sigma_v^2 = \sigma_u^2 B_l = \frac{\varepsilon_{\min}}{P_r} B_l. \tag{8.17}$$

Just as for the ZF DA AGC, fluctuations v of c will induce gradient noise in $\tilde{d}$. By analogy to (8.5) and (8.6), this noise has variance

$$\sigma_{\text{gr}}^2 = P_r \sigma_v^2 = \varepsilon_{\min} B_l. \tag{8.18}$$

As a result, the MSE floor $\varepsilon_{\min}$ is raised by a factor EMSE$= 1+M$, where $M = B_l$ is the misadjustment due to gradient noise. This coincides with the ZF DA AGC (cf. (8.8)). A similar remark applies to the efficiency factor $\gamma = K_t/M \simeq 2$ of the algorithm. Again, this factor is not affected by a nonlinear VCA (Appendix 8B).

Exercise 8.5 Repeat exercise 8.3 for the MMSE DA AGC. Compare the results with those for the ZF DA AGC.

8.3.6 AGC comparison

	AGC type		
	NDA	DA	
	OL, CL	ZF	MMSE
β	$\sqrt{\dfrac{P_d}{P_d+P_n}}$	1	$\dfrac{P_d}{P_d+P_n}$
γ	$\dfrac{8}{\kappa-1}\dfrac{P_n}{P_d}$	2	2

Table 8.1: Summary of AGC properties for the channel of Fig. 8.13. ZF=Zero-Forcing; MMSE = Minimum Mean-Square Error; κ = kurtosis of received sequence r_k; P_d = power of desired sequence d_k; P_n = power of noise n_k.

Key loop properties of the foregoing AGCs are summarized in Table 8.1 for the channel of Fig. 8.13. The open-loop (OL) and closed-loop (CL) NDA entries are extracted from Appendix 8A and Problems 8.5-8.9. ZF and MMSE entries are taken from Sections 8.3.4 and 8.3.5, respectively. The kurtosis $\kappa \triangleq E[r_k^4]/E[r_k^2]^2$ of r_k is 1 when r_k is binary, 2 when r_k is a ternary partial-response sequence, and 3 when r_k is Gaussian (see Problem 8.6). Bias $1-\beta$ and efficiency γ of both NDA AGCs are identical subject to the rather crude assumptions that underly their analysis; in practice a certain difference may show up. Signal suppression factors β are compared in Fig. 8.20 as a function of the signal-to-noise ratio P_d/P_n. The MMSE DA AGC has the largest bias; its ZF counterpart is bias-free. Even for the MMSE DA AGC, bias remains smaller than 3% when $P_d/P_n > 15$ dB. The effect of bias on performance for a full-response system is considered in Problem 8.17.

According to Table 8.1, the DA AGCs are both more efficient than their NDA counterparts for SNRs of practical interest (i.e. for $P_d \gg P_n$). The reason is that they base control in part on decisions that are ideally noise-free, whereas NDA AGCs use only noisy information. At poor SNRs decision errors become likely, and DA AGC performance will no longer live up to the projections of the table. The combined effect of bias and excess mean-square error on detector performance is quantified in Problem 8.18 for a partial-response system with bit-by-bit detection.

Example 8.13 Let d_k be a bipolar $(1-\mathcal{D})$ partial-response sequence in the alphabet $\{-2,0,2\}$ with $P_d = 2$, and let $P_d/P_n = 20$ dB. Then biases $1-\beta$ for the above AGCs are 0.5%, 0 and 1%, respectively. Corresponding steady-state mean-square errors $\varepsilon = (1-\beta)^2 P_d + \beta^2 P_n$ are virtually identical (-17.02, -16.99 and -17.03 dB). Since noise is small, the kurtosis κ of the AGC input r_k will be mainly determined by that of d_k, i.e. $\kappa \simeq 2$. Hence $\gamma \simeq 2/25$ for the NDA AGCs. This is more than an order of magnitude worse than for the DA AGCs. As

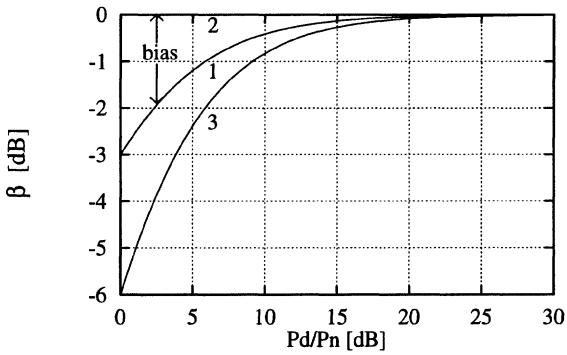


Figure 8.20: Signal suppression factor β versus signal-to-noise ratio P_d/P_n for various AGCs. 1. NDA; 2. ZF; 3. MMSE.

a result, convergence slows down substantially for a given excess mean-square error. This is visible in the *learning curves* of Fig. 8.21. These curves show how the squared error e_k^2 evolves as a function of time. The initial AGC gain is 25% too large, corresponding to an initial MSE of around -8 dB. To suppress the effect of the randomness of data and noise, the curves in Fig. 8.21 were averaged across 1024 experiments. Decisions for the DA AGCs were all correct; a first-order recursive low-pass filter was used for the open-loop AGC.

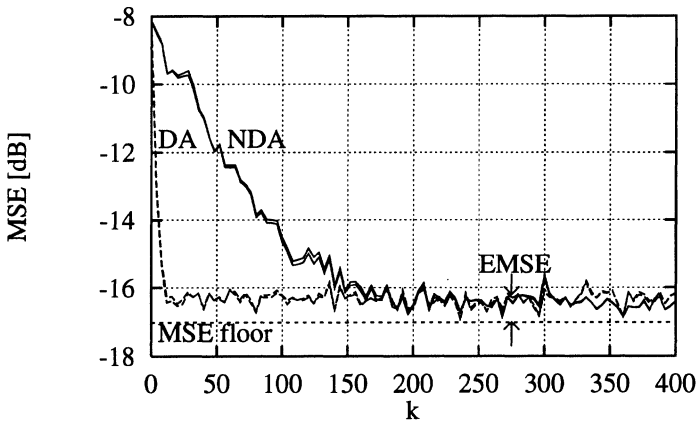


Figure 8.21: AGC learning curves.

All AGCs were dimensioned for an EMSE of 0.5 dB; the actually observed EMSE is slightly larger. Differences between ZF and MMSE curves are negligible; a similar remark applies to both NDA curves. AGC gain c converges initially with a time constant $\tau = 1/K_t \simeq 4/\{\gamma.(\text{EMSE}[dB])\}$ (compare (8.11)). According to (8.3), the squared error e_k^2 decreases initially by around $8.9K_t$ dB/iteration, i.e. 0.12 and 2 dB/iteration for the NDA and DA AGCs, respectively. The actual rates of decrement are slightly smaller.

8.3.7 Effect of nonidealities

For simplicity we have so far neglected various nonidealities. We now make up for this omission.

DC offsets Analog adaptation loops are inevitably plagued by offsets due to circuit imperfections. Offsets may also arise in a digital loop. For example, the integrator input $\mu e_k r_k$ in Fig. 8.17 will be represented with a small number of bits in a cost-conscious digital implementation. To achieve this, the results of the two multiplications that are needed to compute $\mu e_k r_k$ will have to be quantized. This introduces quantization noise that may not be DC-free, depending on the applied arithmetic and quantization technique. The net effect is an offset ξ that we model in Fig. 8.22 at the input of the integrator. (This is, in a sense, a pessimistic model in that it causes the effect of offsets to depend rather heavily on the loop gain K_t. In practice offsets may emerge earlier on in the loop, and the impact of K_t will be less drastic.) Noise is not of interest here and is therefore not modeled.

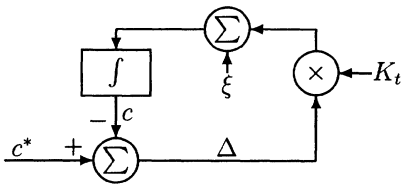


Figure 8.22: First-order control loop with offset ξ.

When referred back to the input terminal, the offset increases by a factor $1/K_t$. The result is that c will settle at $c^* + \xi/K_t$ instead of c^*. A large loop gain K_t is desirable to minimize the bias $\xi/(c^* K_t)$. Small loop gains may not be admissible in spite of the fact that they are desirable from the standpoint of gradient noise. Replacement of the linear VCA by an exponential one does not change these conclusions (see Problem 8.22).

It should be noted that both c^* and K_t depend on the gain λ of the channel. We have seen that $c^* = 1/\lambda$ and $K_t = \lambda\mu P_d$ for the ZF AGC of Fig. 8.14. The bias $\xi/(c^* K_t)$ grows with ξ but is independent of λ. For the MMSE DA AGC, $c^* \simeq 1/\lambda$ and $K_t \simeq \lambda^2 \mu P_d$ at high signal-to-noise ratios. The bias is proportional to ξ/λ, and can become very large for small channel gains. A similar -but aggravated- problem occurs in the adaptive MMSE equalizer (see Section 8.6.5). We will see that integrator leakage can provide partial relief.

Integrator leakage Continuous-time integrators are usually implemented by means of capacitors. These tend to be nonideal, especially when integrated on a chip. The main imperfection is that capacitor charge leaks away at a certain rate. The integrator then behaves as a low-pass filter with transfer function $W(\Omega) = 1/[2\pi(j\Omega + \Omega_c)]$. Here Ω is a normalized measure of frequency with $\Omega = 1$ corresponding to the baud rate $1/T$, and Ω_c is the cutoff frequency normalized to the baud rate. To see how this filter affects

loop properties, we replace the ideal integrator in figs. 8.16 and 8.19 by the low-pass filter. This leads to Fig. 8.23.

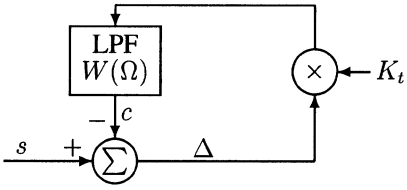


Figure 8.23: First-order control loop with leaky integrator.

The loop transfer function $G_c(\Omega) \triangleq C(\Omega)/S(\Omega)$ is

$$G_c(\Omega) = K_t W(\Omega)/[1 + K_t W(\Omega)] = K_t/[K_t + 2\pi\Omega_c + j2\pi\Omega]. \tag{8.19}$$

The effective loop gain is defined by the value of G_c at DC ($\Omega = 0$) and has increased from K_t to $K_t + 2\pi\Omega_c$. For $\Omega = 0$ (i.e. for a static input $s = S(0)$) we get a steady-state AGC gain

$$c = \frac{K_t}{K_t + 2\pi\Omega_c} s. \tag{8.20}$$

The signal suppression factor $\beta = K_t/(K_t + 2\pi\Omega_c)$ approaches 1 whenever $2\pi\Omega_c \ll K_t$. This condition also ensures that loop gain will not increase noticeably. We may, therefore, allow charge to leak away with a time constant $\tau_{leak} = 1/(2\pi\Omega_c)$ that is much larger than the loop time constant $\tau = 1/K_t$ (both τ_{leak} and τ are normalized in symbol intervals T and therefore dimensionless.) Life gets easier as adaptation gets faster. Since loop gains are proportional to channel gain λ for both ZF and MMSE AGC, the largest permissible cutoff frequency will depend on the smallest possible value of λ.

Exercise 8.6 Assume that λ can vary between 0.1 and 1, and that the loop is designed to have a time constant $\tau = 100$ for $\lambda = 1$. Let the data rate $1/T$ be 10 MHz. What integrator cutoff frequency and time constant τ_{leak} are needed for c to approach s within 10%? Answer this question for both ZF and MMSE AGC. Are the answers different? Which AGC is to be preferred?

As noted earlier, DC offsets effectively cause the reference input s to be biased. The bias is independent of λ for the ZF AGC, but causes s to blow up in proportion to $1/\lambda$ for the MMSE AGC when λ vanishes. The latter effect is clearly undesirable, but is undone by leakage because c/s vanishes quadratically with λ for $K_t < 2\pi\Omega_c$ (see (8.20), and recall that $K_t \propto \lambda^2$ for the MMSE DA AGC). As a result, c vanishes. This is preferable to the unbounded growth that would otherwise occur.

We will find in Section 8.6.5 that MMSE-adaptive equalizers may be 'tamed' in essentially the same manner. The counterpart of λ in equalization is the gain of the channel at a certain frequency (or in a certain frequency band). Tap leakage causes the equalizer to suppress frequencies with negligible content rather than to enhance them up to the point of coefficient overflow or worse. The combination of MMSE adaptation and integrator leakage is sometimes referred to as *Leaky LMS* [32] and is usually described

in a discrete-time form (See Problem 8.23). This is somewhat ironical, given the fact that leakage comes for free in analog loops and requires extra hardware in digital ones.

Loop delay Data-aided AGCs react to signals that are internal to the receiver and that have typically experienced a delay of several symbol intervals since passing the AGC. This delay affects stability and convergence properties.

All loops that were analyzed above are of the first-order type. Thus loop delays have the same effect as for the first-order PLL in Section 11.3.2 (see also [43]). For a negligible degradation of adaptation properties, the loop delay must be much smaller than the adaptation time constant $\tau = 1/K_t$.

AGC overload The preceding linear analysis does not apply when the AGC overloads as a result, for example, of a sudden signal jump. Consequences of overload are ill-predictable and often nasty. For this reason overload is best avoided. This is possible, for example, with the aid of a limiter that precedes the AGC; limit level may be coupled to the instantaneous AGC gain in such a manner as to ensure that signal levels throughout the AGC circuit remain within the range that can be handled.

Decision errors Decision errors do not hamper the non-data-aided AGC but may affect the data-aided AGC adversely. In general, AGC gain will move in the right direction in spite of decision errors, be it more slowly and randomly than otherwise. An important exception to this rule arises for partial-response systems when the initial AGC gain is more than 6 dB too small.

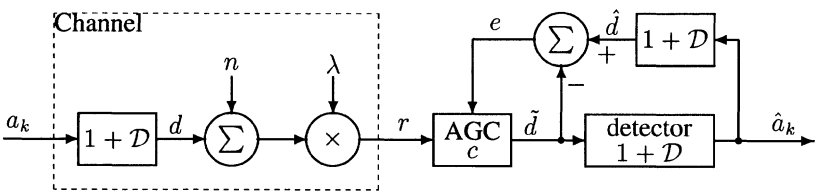


Figure 8.24: Duobinary $(1+\mathcal{D})$ transmission system with data-aided AGC.

Example 8.14 In Fig. 8.24, the desired sequence d_k is of the duobinary $(1+\mathcal{D})$ type, i.e. $d_k = a_k + a_{k-1} \in \{-2, 0, 2\}$, where $a_k \in \{-1, 1\}$ is a random binary data sequence that the detector attempts to reconstruct. When c has the ideal value $1/\lambda$ and noise is not too large, decisions $\hat{a}_k$ will be reliable. However, when c is more than 6 dB too small, $\tilde{d}_k$ will continuously be within the range $[-1, 1]$ that the detector associates with $d_k = 0$. Since Duobinary has a zero at the Nyquist frequency, the all-zero sequence d_k emerges for a data sequence $\ldots - + - + - + \ldots$. This limit cycle $\hat{a}_k$ will be detected irrespective of the actual data a_k. The AGC will strive to adjust c in order for $\tilde{d}_k$ to best fit the sequence $d_k = 0$ that corresponds with the detected data. To this end, c is decreased, i.e. AGC gain moves in the wrong direction.

Similar limit cycles exist for other partial responses. To keep AGC gain from getting stuck at its minimum value, the occurrence of a limit cycle may be detected, and

AGC gain increased until the cycle ceases. From that point on the normal mode of adaptation can be resumed.

8.4 NON-DATA-AIDED ADAPTIVE EQUALIZATION

Adaptation schemes are often but not always data-aided. By analogy to our AGC treatment we commence with a non-data-aided appetizer before moving on to the data-aided main dish. NDA adaptation schemes come in two types:

1. *Self-recovering* (also called *self-training* or *blind*) schemes. Here the data is unknown but the clock is recovered correctly. As a result of this synchronism, standard decision-directed adaptation schemes of the type discussed in Section 8.6 will normally start up correctly when data is binary [53, 50]. Special schemes are needed in other cases [72, 35, 11]; a related approach is described in [78]. A weak spot of blind equalization is that synchronism is required for all possible channel conditions. This may not be easy with a clock recovery scheme that is by necessity non-data-aided.

2. *Asynchronous schemes.* Here the clock needs not be recovered correctly. A direct implication is that only amplitude-related channel imperfections can be corrected. As a rule, the number of adapted parameters is small. A typical example is slope correction for metallic cables [77], [12] and digital recording.

Example 8.15 In Section 8.2.5 (Fig. 8.7) we have encountered an analog equalizer that is suited for joint gain and slope control in recording channels. Fig. 8.25 shows a closed-loop adaptation circuit that is completely based on the output $\tilde{d}$ of this equalizer. Phase distortion is assumed absent, i.e. the first derivative y' of the prefilter output y is not used. Linear combination of y and its second derivative y'' permits joint gain and slope control.

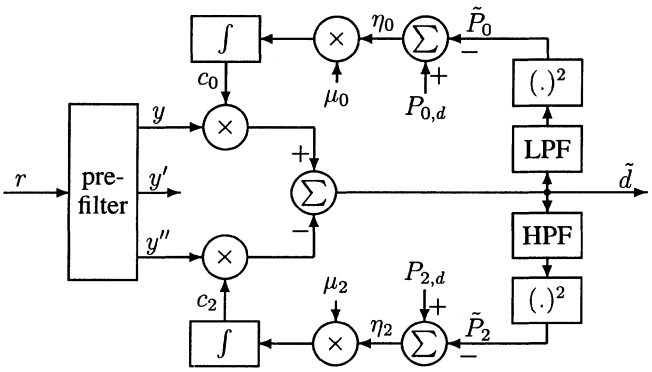


Figure 8.25: Non-data-aided closed-loop adaptation circuit for the analog equalizer of Fig. 8.7.

The desired output d of the equalizer is well defined, and so is its amplitude spectrum. The spectrum of the actual output $\tilde{d}$ depends on c_0 and c_2, with c_2 affecting high frequencies more heavily than c_0. This suggests that we adjust c_2 in order to force the power in a high-frequency portion of the spectrum to a desired value $P_{2,d}$, and adjust c_0 to force power in another range to a desired value $P_{0,d}$. This is what the circuit of Fig. 8.25 strives for. It is essentially a 'stereo' version of the closed-loop AGC of Fig. 8.9, and retains most of its properties.

The characteristics of LPF and HPF along with the desired signal d determine $P_{0,d}$ and $P_{2,d}$ (see Problem 8.25). Choice of both filters is an engineering compromise that involves such aspects as steady-state ISI under different channel conditions, gradient noise, required and attainable implementation accuracies, and filter complexity. Very simple filters may do as long as the range of gains and slopes that must be handled is not too extreme. For example, the HPF might be a first-order differentiator and the LPF might not be needed at all. Some of these aspects are addressed in Problem 8.25 and in Appendix 8A. Although the appendix addresses only the single-loop case, the approach and most of the conclusions carry over to the situation at hand.

In principle, both control loops will interact with each other, i.e. c_0 and c_2 will converge in a coupled manner. Decoupling may improve convergence properties and is possible with a weighing network. These issues are addressed for data-aided adaptation in the sections ahead.

This example illustrates how amplitude distortion can be undone by casting the amplitude spectrum at the output of the equalizer adaptively into a predefined mould. This approach is described in a more general form in [9]. It is inherently unsuited for correction of phase distortion and has most of the drawbacks of the closed-loop NDA AGC, but in return offers two compelling advantages:

1. Erroneous decisions and sampling phase errors do not come into play. As a result, adaptation is highly robust, and startup problems are nonexistent.

2. Circuitry can be simple and lean on power consumption when implemented in the analog domain.

Amplitude distortion may also be corrected in an open-loop fashion. We again illustrate this for the analog derivative equalizer.

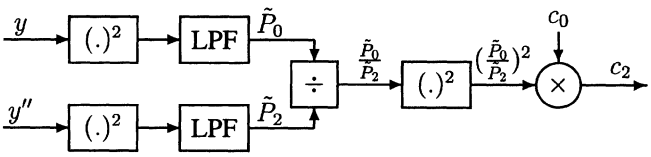


Figure 8.26: Open-loop slope adjustment circuit for the analog equalizer of Fig. 8.7.

Example 8.16 Fig. 8.26 shows an open-loop circuit for slope correction with the equalizer of Fig. 8.25. The circuit is based on the fact that the power P_0 of y relative the power P_2 of y'' is a measure of the required slope correction. In cases where much slope correction is needed, y will contain mainly low frequencies, little power will remain after twofold differentiation, and P_0/P_2 will be large. The opposite applies when little correction is needed. The exact amount of correction that must be applied for a given ratio P_0/P_2 can be determined analytically when

a channel model is available. In other cases experiments may be needed. A typical correction function is shown in Fig. 8.26. The circuitry that is shown will only determine the ratio of c_0 and c_2, i.e. the required degree of slope correction. Gain correction, i.e. adaptation of the absolute value of c_0, can be done in any of the previously discussed ways.

The behavior of this scheme is similar to that of the open-loop AGC of Section 8.3. Added disadvantage with respect to the scheme of Fig. 8.25 is increased sensitivity to implementation inaccuracies in the control path; potential advantage is that (analog) implementation may be still simpler.

8.5 ZERO-FORCING EQUALIZER ADAPTATION

In the zero-forcing AGC, a single parameter is controlled in order to adjust one system feature (viz. the system gain). Any mismatch is forced to zero. In this section we extend this approach to adaptive equalization. Here a total of N parameters is to be adjusted. Ideally, all ISI is forced to zero. However, since there are infinitely many ISI terms (related to past, current and future bits), this is, in general, intrinsically impossible, and some compromise must be made.[3] This is fundamentally different from the ZF AGC, where perfect adjustment is essentially always possible. To expose the implications of this difference, we first give a review of ZF equalization with a *finite number of free parameters*. This extends the treatment of Section 5, in which an infinitely long equalizer was assumed. The step towards adaptation is then straightforward and very similar to that for the ZF AGC.

ZF adaptation is usually motivated by its comparative simplicity [49, 66]. Its widespread use testifies to this advantage [9, 59]. Treatments of ZF adaptation in textbooks are nevertheless rare, perhaps because performance is perceived as poor. In this section we will see that ZF has various arguments going for it. For example, prior knowledge about the channel characteristics can be hardwired into the loop in order to achieve low ISI, a high adaptation efficiency and full coefficient decoupling. Also, adaptation properties are rather insensitive to certain variations of channel and circuit parameters. MMSE adaptation, as treated in Section 8.6 and in a large fraction of the adaptation literature, lags behind ZF in several of these respects.

8.5.1 ZF equalization with a finite number of parameters

ZF equalization has traditionally been used with transversal equalizers but is, in fact, applicable to any equalizer that contains a linear combiner. A case in point is the analog derivative equalizer of Fig. 8.7. We first consider a Class IV system that uses this equalizer, and then extend the treatment to other target responses and equalizer structures.

In Fig. 8.27, a data sequence a_k is applied to a channel with impulse response $h(t)$ and additive noise $n(t)$. The analog equalizer of Fig. 8.7 operates on the replay signal

[3] A direct consequence is that the adjective 'zero-forcing' is, strictly speaking, incorrect. In line with common practice we shall nevertheless stick to it.

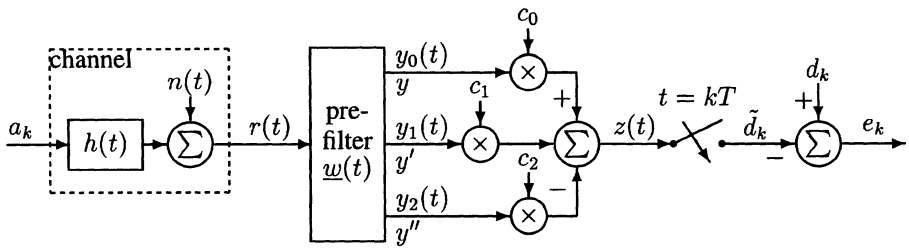


Figure 8.27: Data transmission system with the analog equalizer of Fig. 8.7.

$r(t)$ in order to produce an output $z(t)$ that is desired to be Class IV at the decision instants $t = kT$. Decision variables $\tilde{d}_k$ are compared to a reference Class IV sequence d_k in order to obtain an error sequence e_k that can serve as a basis for adaptation of the $N = 3$ equalizer coefficients c_0, c_1 and c_2.

Let $w_i(t)$ be the impulse response of the ith prefilter branch, so that $y_i(t) = (r * w_i)(t)$ for $i = 0, \ldots, N-1$. These tap signals can be split into data and noise components according to

$$y_i(t) = \sum_{k=-\infty}^{\infty} a_k (h * w_i)(t - kT) + (n * w_i)(t). \tag{8.21}$$

The sampled output $\tilde{d}_k$ of the equalizer is

$$\tilde{d}_k = \sum_{i=0}^{N-1} y_i(kT) c_i = \underline{y}_k^T \underline{c}, \tag{8.22}$$

where $\underline{y}_k \triangleq [y_0(kT), \ldots, y_{N-1}(kT)]^T$ is a column vector of length N that characterizes the tap signals at the sampling instants $t = kT$, and $\underline{c} \triangleq [c_0, \ldots, c_{N-1}]^T$ is a similar vector of tap values (the superscript ^{T} indicates transposition).

In practice the sampled symbol responses $(h * w_i)(kT)$ up until the prefilter all have a finite duration. That is, they all vanish outside some span S so that $(h * w_i)(kT) = 0$ for $k \notin S$, irrespective of i. Let $S_P \triangleq \{P_l, \ldots, P_u\}$ be a given span of length $P \triangleq P_u - P_l + 1$. Then the $N \times P$ matrix

$$\mathbf{F} \triangleq \begin{pmatrix} (h * w_0)(P_l T) & \cdots & (h * w_0)(0) & \cdots & (h * w_0)(P_u T) \\ \vdots & & \vdots & & \vdots \\ (h * w_{N-1})(P_l T) & \cdots & (h * w_{N-1})(0) & \cdots & (h * w_{N-1})(P_u T) \end{pmatrix} \tag{8.23}$$

contains all nontrivial samples of all responses $(h * w_i)(kT)$ if S_P encompasses S. Conversely, one or more nontrivial samples may not be covered by $\mathbf{F}$ if S_P does not encompass S. Typically $P_l \leq 0$ and $P_u > 0$. The span S_P is a design freedom whose role will become apparent later on. For the time being we will assume that S_P indeed covers S

so that all nontrivial samples $(h * w_i)(kT)$ are collected in $\mathbf{F}$. With the aid of (8.21) we can then split $\underline{y}_k$ into data and noise components according to

$$\underline{y}_k = \mathbf{F}\underline{a}_k + \underline{u}_k \tag{8.24}$$

where $\underline{a}_k \triangleq [a_{k-P_l}, \ldots, a_{k-P_u}]^T$ is a data vector of length P and $\underline{u}_k$ is a vector of length N that characterizes the noise components of the tap signals. The equivalent discrete-time vector model implied by (8.22) and (8.24) is shown in Fig. 8.28. The system is linear and has an overall system response q_k that determines a *system response vector* $\underline{q} = [q_{P_l}, \ldots, q_{P_u}]^T$. Clearly $\underline{q} = \mathbf{F}^T\underline{c}$. Since S_P encompasses S, $\underline{q}$ will cover all nontrivial samples of q_k, i.e. $q_k = 0$ for $k \notin S_P$.

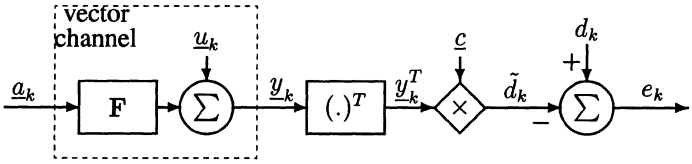


Figure 8.28: Discrete-time vector model equivalent to the continuous-time system of Fig. 8.27.

Fig. 8.28 does not exhibit the vector structure of d_k. As discussed in Chapter 6, d_k is determined by a *target response* g_k according to $d_k = (a * g)_k$. For Class IV this response has $\mathcal{D}$-transform $1 - \mathcal{D}^2$ so that $d_k = a_k - a_{k-2}$. In the sequel we will characterize g_k by a *target response vector* $\underline{g} \triangleq [g_{P_l}, \ldots, g_{P_u}]$. In practice the extent of g_k is invariably smaller than the span S and since S_P covers S, $\underline{g}$ will contain all nontrivial samples g_k. In order for all ISI to vanish it is necessary that the system response q_k fully coincides with the target response g_k, i.e. that the error response $\Delta_k \triangleq q_k - g_k$ vanishes for all k. Since S_P covers S, it suffices to require that $\underline{q} = \underline{g}$, i.e. that the *system response error vector*

$$\underline{\Delta} \triangleq \underline{g} - \underline{q} = \underline{g} - \mathbf{F}^T\underline{c} \tag{8.25}$$

is the all-zero vector of length P. We may influence $\underline{q}$ and thereby $\underline{\Delta}$ via the coefficient vector $\underline{c}$. Elimination of all ISI requires that all P components of $\underline{\Delta}$ are forced to zero. The equalizer has N degrees of freedom to accomplish this. In the usual event that $N < P$, it may be intrinsically impossible to force all ISI to zero, and a compromise must be struck. Two possibilities, namely partial zero-forcing and minimization of ISI power, are discussed below.

For future reference we note that the error e_k may be expressed in terms of the above quantities as

$$e_k = d_k - \underline{a}_k^T\mathbf{F}^T\underline{c} - \underline{u}_k^T\underline{c} = \underline{a}_k^T[\underline{g} - \underline{q}] + \zeta_k = \underline{a}_k^T\underline{\Delta} + \zeta_k \tag{8.26}$$

where $\zeta_k \triangleq \underline{u}_k^T\underline{c}$ is the noise component of $\tilde{d}_k$ and e_k. Clearly e_k consists purely of noise ζ_k if and only if $\underline{\Delta}$ is the all-zero vector, i.e. if and only if all ISI components are forced to zero.

The model of Fig. 8.28 is applicable to any linear system in which the equalizer contains a weighing network that combines two or more internal signals into an equalizer output. Examples include the analog equalizer of Fig. 8.7 and the transversal equalizer of Fig. 8.5, in which the linear combiner operates on successive taps of a delay line.

Exercise 8.7 Assume that the analog equalizer in Fig. 8.27 is replaced by a T-spaced transversal filter with N taps that is preceded by fixed prefilter. Show that the equivalent discrete-time model of Fig. 8.28 continues to apply. How are the channel matrix $\mathbf{F}$ and the noise vector $\underline{u}$ defined in this case? Repeat this exercise for a fractionally-spaced equalizer with tap spacing T/L for some integer $L \geq 1$.

Restriction of S_P For simplicity we have assumed so far that S_P covers S and thereby also all nontrivial coefficients g_k. We will soon identify S_P as a design freedom that defines the span across which the adaptation algorithm attempts to minimize ISI. In some cases it is desirable for S_P to be so small that some of the nontrivial components of the system response q_k and/or the target response g_k are no longer captured in $\underline{q}$ and $\underline{g}$, respectively. A direct implication is that the vector $\underline{\Delta}$ no longer describes all ISI components but only the ones within S_P. Expressions (8.24) and (8.26) remain valid in this case provided only that we accommodate the remaining ISI components (i.e. those outside S_P) in $\underline{u}_k$ and thereby in ζ_k.

Exercise 8.8 Re-express $\underline{u}_k$, e_k and ζ_k for this situation.

Partial zero-forcing (PZF) According to this classical approach [48, 49], the N degrees of freedom of the equalizer are used to force N consecutive components Δ_k toward zero. Remaining components are not constrained. In the adopted notation, this is equivalent to choosing $P = N$. The index P_l of the first component Δ_{P_l} is a design freedom; the span $S_P = \{P_l, \ldots, P_l + N - 1 = P_u\}$ is then fully determined. The channel matrix $\mathbf{F}$ is now square ($N \times N$) and is usually invertible. The set of equations $\underline{\Delta} = \underline{g} - \mathbf{F}\underline{c} = \underline{0}$ has solution $\underline{c} = \mathbf{F}^{-1}\underline{g}$. Main disadvantage of this approach is that even when P_l is selected optimally, some components Δ_k that are not included in $\underline{\Delta}$ may deviate significantly from zero, i.e. residual ISI outside S_P may be large. Nevertheless, for full-response T-spaced transversal equalizers it can be shown that PZF minimizes the so-called *peak distortion* or *peak eye closure* whenever the eye before the equalizer is already open [48, 49]. Peak distortion is closely related to bit error probability at high signal-to-noise ratios.

Minimization of ISI power (MMSI) As $\underline{c}$ traverses the N-dimensional space of possible equalizer settings, $\underline{\Delta}$ traverses an N-dimensional subspace of the P-dimensional space of possible error vectors. As a rule, this subspace will not include the origin $\underline{\Delta} = \underline{0}$, i.e. elimination of all ISI within S_P is intrinsically impossible. Instead, we may seek to minimize the L_2-norm $||\underline{\Delta}||^2$ of the error vector. For uncorrelated data this corresponds to minimization of all ISI power within S_P; ISI outside this span remains unconstrained. In geometrical terms, we must select the point in the P-dimensional

subspace with minimum Euclidian distance $||\underline{\Delta}||$ to the origin. In this point, the error vector $\underline{\Delta} = \underline{g} - \mathbf{F}^T\underline{c}$ is perpendicular to the N-dimensional subspace spanned by $\mathbf{F}^T\underline{c}$, i.e. $(\mathbf{F}^T\underline{c})^T(\underline{g} - \mathbf{F}^T\underline{c}) = 0$ [76]. The corresponding optimum tap vector $\hat{\underline{c}}$ and error vector $\hat{\underline{\Delta}}$ are

$$\hat{\underline{c}} = (\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F}\underline{g} \tag{8.27}$$

and

$$\hat{\underline{\Delta}} = (\mathbf{I} - \mathbf{F}^T(\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F})\underline{g}, \tag{8.28}$$

where $\mathbf{I}$ is the $P \times P$ identity matrix.

Minimization of *total* ISI power necessitates that S_P encompasses S. The corresponding value of P may be large. As P is decreased, minimum ISI power within the span S_P will decrease, but additional ISI power outside S_P will emerge so that total ISI power increases. Complexity of the equalizer depends only on N and is independent of P. Conversely, ZF adaptation circuitry that controls the equalizer has a complexity that grows with P (see Section 8.5.2). For this reason it is of interest to know the smallest acceptable span S_P. In practice, a length P on the order of the channel memory length M will often do.

Exercise 8.9 Argue that PZF may be viewed as a special case of MMSI that emerges when P is lowered to N.

Exercise 8.10 Consider a channel/prefilter combination with a memory length of L symbol intervals that is followed by a T-spaced equalizer with N taps. Show that a value $P = N + L$ suffices to minimize total ISI power.

8.5.2 ZF adaptation

In the ZF AGC of Fig. 8.14, the AGC value is formed by an integrator whose input is basically the cross-product of the error e and a reference sequence $x \triangleq \mu d$. The structure of the ZF AGC is easily extended to multi-parameter adaptation (Fig. 8.29). The ZF adaptation loop of Fig. 8.29 has a tap vector $\underline{c}$ that is formed by a bank of N integrators, whose input is a cross-product $e\underline{x}$ where $\underline{x}$ is a vector of N reference sequences $[x_0, \ldots, x_{N-1}]^T$. This reference vector is obtained by applying the data sequence a_k to a bank of N filters with vectorial impulse response $\underline{w}_k \triangleq [w_k^0, \ldots w_k^{N-1}]^T$. This response is a design freedom that can be used to optimize adaptation properties. (The functions $\underline{w}_k$ and w_k^i should not be confused with the continuous-time impulse responses $\underline{w}(t)$ and $w^i(t)$ of the prefilter in Fig. 8.27.) The ZF AGC of Fig. 8.14 is a special case of the loop of Fig. 8.29 with $N = 1$, $x_k^0 = \mu d_k$ and $w_k^0 = \mu g_k$.

At moment k, the ith integrator in Fig. 8.29 has input

$$z_k^i = e_k(a * w^i)_k = \sum_{j \in S_P} w_j^i(e_k a_{k-j}). \tag{8.29}$$

In this equation it is assumed that all weighing networks have span S_P, i.e. that $w_k^i = 0$ for $k \notin S_P$, irrespective of i. We may apparently regard x_k^i as a weighted sum of P

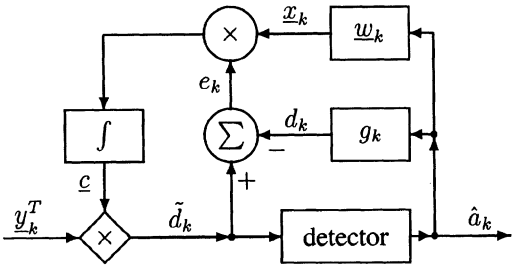


Figure 8.29: Basic structure of adaptive ZF equalizer.

cross-correlations $e_k a_{k-j}, j \in S_P$. For each integrator there are P weights w_k^i, and all weights together may be combined into an $N \times P$ matrix

$$\mathbf{M} \triangleq \begin{pmatrix} w_{P_l}^0 & \cdots & w_0^0 & \cdots & w_{P_u}^0 \\ \vdots & & \vdots & & \vdots \\ w_{P_l}^{N-1} & \cdots & w_0^{N-1} & \cdots & w_{P_u}^{N-1} \end{pmatrix}. \tag{8.30}$$

We arrive at the equivalent loop model of Fig. 8.30.

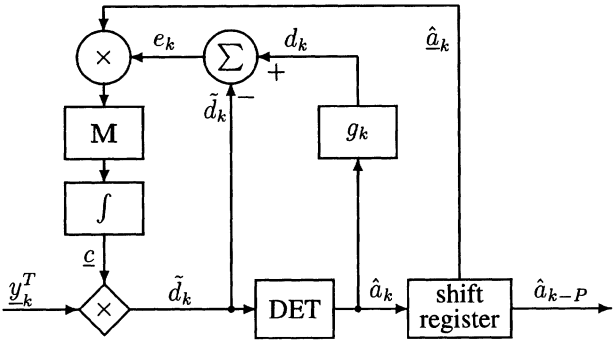


Figure 8.30: Alternative version of the loop of Fig. 8.30.

In Fig. 8.30, the data vector $\hat{\underline{a}}_k \triangleq [\hat{a}_{k-P_l}, \ldots, \hat{a}_{k-P_u}]^T$ is obtained from the detector output $\hat{a}_k$ via a shift register. The integrator bank is excited by the P cross-products $\hat{\underline{a}}_k e_k$ via a weighing network that is described by the matrix $\mathbf{M}$. The fundamental difference between the loops of Figs. 8.29 and 8.30 is that filtering and cross-correlation are reversed in order. This reversal does not affect loop operation but it does affect implementation. For example, cross-correlation of e_k with a discrete-valued data sequence a_{k-j} as in Fig. 8.30 tends to be easy, while correlation with a real-valued sequence x_k^i as in Fig. 8.29 can be cumbersome. In our development of ZF adaptation we will take the loop of Fig. 8.30 as a starting point. Before analyzing adaptation properties we point

out that the data vector $\underline{a}_k$ may contain *future* symbol decisions, depending on the adopted span S_P. In that case $\hat{\underline{a}}_k$ in Fig. 8.30 would contain symbols that have not yet been detected. This is clearly impossible. The solution is to admit a delay $\delta \geq P_l$ such that the shift register becomes causal. The error sequence e_k must be delayed accordingly. The net effect is a delay of δ symbol intervals in the control of $\underline{c}$. This delay does not affect the steady-state solution. It does, however, restrict stability ranges and attainable convergence rates (see Sections 8.3.7 and 11.3.2).

Steady-state solution In steady state, the integrator outputs $\underline{c}$ will all be finite (provided, of course, that the loop is stable). A direct implication is that the integrator inputs $\mathbf{M}\hat{\underline{a}}_k e_k$ cannot contain any DC. Thus the adaptation circuit will seek to force a weighted version $\mathbf{M}E[\hat{\underline{a}}_k e_k]$ of the cross-correlation $E[\hat{\underline{a}}_k e_k]$ toward zero. This cross-correlation is easily evaluated when decision errors are absent ($\hat{\underline{a}}_k = \underline{a}_k$) and ζ_k is uncorrelated with the data vector $\underline{a}_k$. With (8.26) we then find that

$$E[\hat{\underline{a}}_k e_k] = E[\underline{a}_k(\underline{a}_k^T \underline{\Delta} + \zeta_k)] = \mathbf{R}_a \underline{\Delta}, \tag{8.31}$$

where $\mathbf{R}_a \triangleq E[\underline{a}_k \underline{a}_k^T]$ is the data autocorrelation matrix. Unless stated otherwise we shall assume throughout this section that data is uncorrelated and has unit power, so that $\mathbf{R}_a = \mathbf{I}$ and $E[\underline{a}_k e_k] = \underline{\Delta}$. This means that the cross-product $\hat{\underline{a}}_k e_k$ as formed in Fig. 8.30 is in essence a random version of the system response error vector $\underline{\Delta}$. The loop suppresses this randomness and forces a weighted version $\mathbf{M}\underline{\Delta}$ of $\underline{\Delta}$ toward zero. Since $\underline{\Delta} = \underline{g} - \mathbf{F}^T \underline{c}$, the coefficient vector $\underline{c}$ will settle at a value such that $\mathbf{M}\mathbf{F}^T \underline{c} = \mathbf{M}\underline{g}$. Now $\mathbf{M}\mathbf{F}^T$ is an $N \times N$ matrix and is usually invertible. Consequently, $\underline{c}$ will converge towards the unique solution

$$\underline{c} = (\mathbf{M}\mathbf{F}^T)^{-1}\mathbf{M}\underline{g}. \tag{8.32}$$

This solution is similar in form to (8.27). For $\underline{c}$ to converge towards $\hat{\underline{c}}$, we can apparently use an adaptation matrix $\mathbf{M}$ that is proportional to the channel matrix $\mathbf{F}$.

Example 8.17 Let $\mathbf{M} = \mu\mathbf{F}$, where μ is a non-negative scaling factor. Then the bank of integrators has input $\mu\mathbf{F}\underline{a}_k e_k$. This means that e_k is correlated with the data component $\mathbf{F}\underline{a}_k$ of the tap vector $\underline{y}_k$ (compare (8.24)) and scaled by a factor μ. We will see in Section 8.6 that this approach is closely related to LMS adaptation.

The matrix $\mathbf{M} = \mu\mathbf{F}$ is not the only one that minimizes ISI power. Let $\mathbf{U}$ be an arbitrary invertible $N \times N$ matrix, and consider the adaptation matrix $\mathbf{M} \triangleq \mathbf{U}\mathbf{F}$. This matrix yields steady-state solution $\underline{c} = (\mathbf{U}\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{U}\mathbf{F}\underline{g} = (\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{U}^{-1}\mathbf{U}\mathbf{F}\underline{g} = \hat{\underline{c}}$ irrespective of $\mathbf{U}$. We can regard $\mathbf{U}$ as a design freedom that allows us to define loop gains and interaction between coefficients. It is possible to select $\mathbf{U}$ and thereby $\mathbf{M}$ such that all coefficients are decoupled and all N loop gains are equal. It is intuitively obvious that this is desirable. A motivation may be found in Appendix 8C. This appendix also proves the following Theorem.

Theorem 8.1 *Let* $\mathbf{M}$ *be such that*

$$\mathbf{M}\mathbf{F}^T = K_t\mathbf{I}, \tag{8.33}$$

where K_t is a positive constant. Then there is no cross-coupling between coefficients and all loop gains are equal to K_t.

With slight abuse of terminology we call a loop that fulfills (8.33) *fully decoupled* or *orthogonalized.* The orthogonalized MMSI loop with gain K_t has adaptation matrix

$$\mathbf{M} = K_t(\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F}. \tag{8.34}$$

This matrix satisfies (8.33), i.e. the loop is indeed orthogonalized and has gain K_t. It minimizes ISI power because it is of the form $\mathbf{UF}$, with $\mathbf{U} = K_t(\mathbf{F}\mathbf{F}^T)^{-1}$. It is worth noting that the orthogonalized MMSI loop is completely determined by the channel matrix $\mathbf{F}$. Since $\underline{g}$ does not come into play, the employed reception technique is of no concern. This suggests that we may view the weighing network $\mathbf{M}$ as *a priori* knowledge about the channel that is hardwired into the loop.

Of course, if $\mathbf{F}$ were exactly known, then adaptation would not be needed. In practice, the actual channel matrix $\mathbf{F}$ will vary to a certain extent. The matrix $\mathbf{M}$ is best based on the worst-case channel conditions that are expected. This guarantees minimum ISI power under worst-case conditions, at the expense of a possible increment in other cases. In practice, a well-designed ZF loop will perform satisfactorily across a considerable range of channel conditions.

Example 8.18 In Exercise 8.9 we have seen that MMSI degenerates into PZF for $P = N$. In that case $\mathbf{F}$ is an $N \times N$ matrix that is usually invertible. By (8.34), the orthogonalized PZF loop has adaptation matrix $\mathbf{M} = K_t[\mathbf{F}^T]^{-1}$. This loop is illustrated in Fig. 8.31 for a full-response T-spaced equalizer. Here the desired sequence d_k is just the transmitted data sequence a_k, and the error sequence is formed by subtracting the decision variables $\tilde{a}_k = \tilde{d}_k$ from the decisions $\hat{a}_k$. A delay of P_l symbol intervals in the error path allows P_l pre-cursive ISI-terms to be constrained (see remarks following Fig. 8.30). The remaining $N - P_l$ degrees of freedom are spent on the amplitude q_0 of the detected bit and the first $P_u = N - P_l - 1$ post-cursive ISI-terms. ISI outside the span $S_P = \{P_l, \ldots, P_u\}$ is not constrained and may become significant. The central tap position is uniquely defined (Problem 8.28). A longer equalizer (i.e. an increment of N) will lower this residual ISI, but this remedy has various disadvantages (see Section 8.2.4).

With a discrete-time input sequence, circuitry of Fig. 8.31 will normally be digital. Sign inversions suffice to produce the cross-products $e_{k-P_l}\hat{a}_{k-j}$ when a_k is binary. To simplify hardware, coefficients of the weighing network $\mathbf{M}$ can be strongly quantized. This is typically possible without serious performance degradations. Most of the circuitry becomes analog if sampling is postponed until after the transversal filter.

Replacement of $\mathbf{M}$ by another invertible matrix will induce coupling but will still lead to the PZF solution, provided that the loop remains stable. Of particular interest is the scaled identity matrix $\mathbf{M} = \mu\mathbf{I}$, for which all cross-couplings in the weighing network disappear. This yields the classical T-spaced ZF equalizer, which takes center stage in the few textbooks that devote attention to ZF adaptation, such as [48] and [66].

8.5.3 Properties of the orthogonalized MMSI loop

In effect, this loop consists of N independent AGCs that are operating in parallel. All have the same time constant $\tau = 1/K_t$, and are stable for $K_t < 2$ in the absence

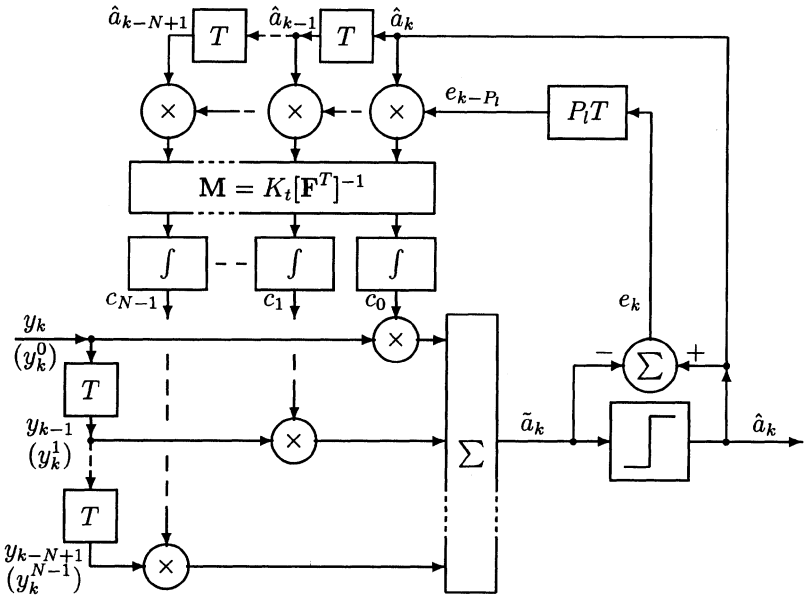


Figure 8.31: Adaptive T-spaced full-response equalizer. Adaptation involves an orthogonalized partial zero-forcing (PZF) loop.

of implicit or explicit loop delays (see Section 8.3.7 for a review of the effect of such delays). Each sub-loop contributes the same amount of gradient noise to $\tilde{d}_k$. In practice these contributions will add in power, i.e. the total gradient noise power is N times as large as for a single AGC.

Let us denote by e_k the actual error sequence (including gradient noise), and by $\dot{e}_k$ the error sequence excluding gradient noise that would arise if coefficient fluctuations were absent. On top of filtered channel noise, $\dot{e}_k$ generally contains residual ISI, both within and outside the span S_P that is exploited for adaptation. Gradient noise in $\tilde{d}_k$ and e_k has power

$$\sigma_{\text{gr}}^2 \simeq \frac{N}{\alpha} B_l \varepsilon \tag{8.35}$$

where $B_l = K_t/(2 - K_t)$ denotes the noise bandwidth of a single sub-loop and ε denotes the power of $\dot{e}_k$. The loss factor $\alpha \leq 1$ represents, loosely speaking, the fraction of all data power that is within the span S_P. A more accurate definition is given in appendix 8D for the single-tap case ($N = 1$). For a well-designed loop, α will not be much smaller than 1 because S_P will cover most of the memory length of the channel. Expression (8.35) generalizes the corresponding expression for the ZF AGC (8.6). The misadjustment $M \triangleq \sigma_{\text{gr}}^2/\varepsilon$ due to gradient noise amounts to NB_l/α. For loop gains of practical interest we have $B_l \simeq K_t/2$, and we may attribute an efficiency factor

$$\gamma \triangleq \frac{1}{M\tau} \simeq \frac{2\alpha}{N} \tag{8.36}$$

to the loop. This factor decreases in proportion to the number of adjusted parameters.

For the sake of simplicity we have applied a somewhat heuristic reasoning to arrive at (8.35). A more accurate analysis validates this expression subject to certain assumptions and approximations that are usually justifiable [90]. Appendix 8D, which is devoted to the single-tap case $N = 1$, gives a flavor of the approach.

Exercise 8.11 Assume that a 16-tap equalizer is controlled by means of an orthogonalized MMSI loop that exploits all data power and is dimensioned to have a time constant τ of 100 symbol intervals. How much does the predetection signal-to-noise ratio drop due to gradient noise? What time constant τ is attainable if we insist that this degradation be at most 0.1 dB?

Example 8.19 Consider a bipolar $(1 - \mathcal{D})$ partial-response system as in example 8.13, with the AGC replaced by a T-spaced transversal equalizer having $N = 15$ taps.[4] Then MMSI equalization necessitates a span $S_P = \{0, \ldots, N\}$ (see Exercise 8.10). The corresponding orthogonalized $(N+1) \times N$ adaptation matrix $\mathbf{M}$ is determined in Problem 8.31. The weighing network characterized by $\mathbf{M}$ is rather complicated for $P = 15$ because all $N(N+1)$ weights are nonzero. By comparison, the classical PZF loop due to Lucky has an $N \times N$ network characterized by $\mathbf{M} = \mu\mathbf{I}$, i.e. only N weights are nonzero. Learning curves of these two loops are compared in Fig. 8.32. As in Example 8.13, both loops are dimensioned for a nominal steady-state EMSE of 0.5 dB, and initial equalizer settings ($c_7 = 1.25$, $c_i = 0$ for $i \neq 7$) are such that the initial MSE is around -8 dB. Since all ISI can be suppressed for the channel at hand even with a single equalizer tap, MSE floors for ZF and MMSI are equal and coincide with those of Fig. 8.21 (-17 dB).

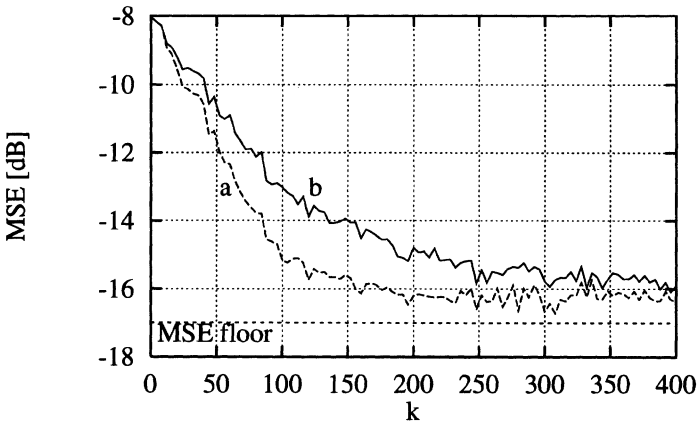


Figure 8.32: Learning curves for two adaptation loops with a nominal steady-state EMSE of 0.5 dB. a. Orthogonalized minimum mean-square ISI (MMSI); b. Coupled partial zero-forcing (PZF).

Due to its coupled nature, the PZF loop converges significantly slower than the orthogonalized MMSI loop. Both loops are at least 15 times slower than the AGCs of Example 8.13

[4]Application of such an equalizer may be motivated by *a priori* unknown excursions of the channel about the nominal $1 - \mathcal{D}$ characteristics. In this example we restrict attention to this nominal channel.

(compare Fig. 8.21). This penalty is a direct consequence of having $N = 15$ adjustable parameters rather than only a single one (compare the efficiency factor of (8.10) with that of (8.36) for $\alpha = 1$). This suggests that no more adaptivity be used than strictly needed (see Section 8.2.4 for further considerations in this context).

The use of $16 \times 15 = 240$ weights instead of only 15 is not an attractive way to achieve the speed advantage of Fig. 8.32. A more practical approach is to implement only the largest weights in order to retain most of the advantage of orthogonalization at greatly reduced cost. Even so, orthogonalization will be most practical when only a few taps are to be adjusted. The next example deals with such a situation.

Example 8.20 With an appropriate low-pass prefilter, the derivative equalizer of Fig. 8.7 can be used for partial-response Class IV $(1 - \mathcal{D}^2)$ reception on magnetic recording channels. The continuous-time system response $q(t)$ is ideally $1 - \mathcal{D}^2$ when sampled at the proper instants (Fig. 8.33).

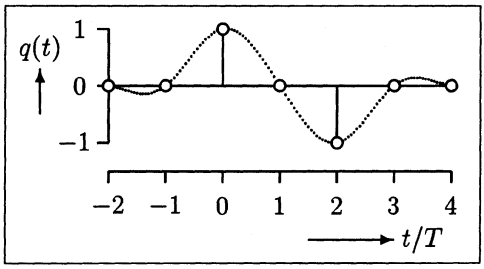


Figure 8.33: Ideal Class IV system response $q(t)$ and its sampled version $q_k = g_k$.

Let us assume that channel plus prefilter introduce no phase distortion. In that case the first derivative y' of the prefilter needs not be used (see Section 8.2.5). Linear combination of y and its second derivative y'' permits joint gain and slope control. An erroneous system gain manifests itself in that q_0 and q_2 deviate from the desired values $g_0 = 1$ and $g_2 = -1$, respectively. An erroneous slope causes the pulse to become too slim or too fat. As a result, especially q_{-1} and q_3 will deviate from their desired values $g_{-1} = 0$ and $g_3 = 0$; to a smaller extent, q_{-2} and q_4 will also change. Finally, an erroneous sampling phase will cause especially q_1 to deviate from the desired value $g_1 = 0$. This reasoning suggests that the span $S_P = \{-2, \ldots, 4\}$ contains most of the available control information. If the prefilter is designed properly, then minimization of ISI within this span should ensure that ISI outside it is small as well under normal operating conditions. Further restriction of S_P would normally increase total ISI. Because of the guaranteed symmetry of $q(t)$, however, it suffices to base control on the right (or left) half of the pulse in order to ensure that the entire pulse is equalized correctly. This means that we can take $S_P = \{2, 3, 4\}$ for joint slope and gain control. The price of the resulting simplification of adaptation circuitry is that at least half of all data power is not exploited for adaptation. This doubles adaptation time constants for a given excess mean-square error (compare (8.36) with $\alpha \simeq 0.5$). An implementation is shown in Fig. 8.34.

The weighing network has three inputs and two outputs. The underlying 3×2 matrix $\mathbf{M}$ is designed on the basis of (8.34) for worst-case channel conditions, i.e. for the largest possible normalized information density t_{50}/T. Smaller densities do not greatly increase steady-state

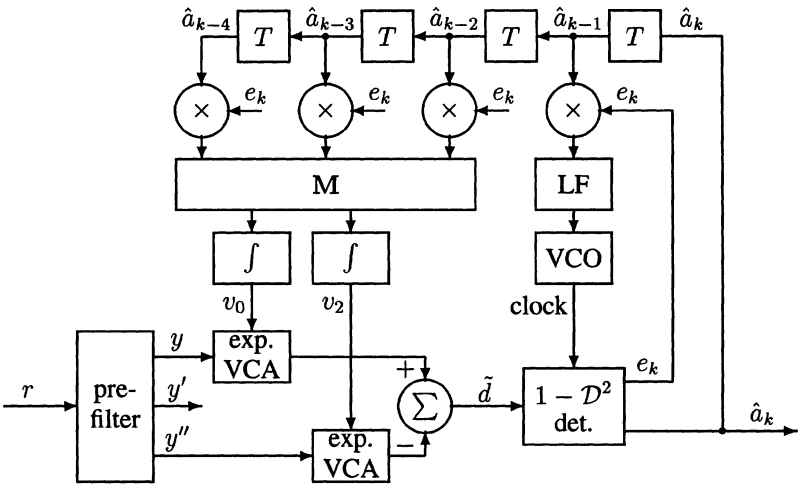


Figure 8.34: Baseline topology of MMSI equalizer for Class IV reception on magnetic recording channels. The equalizer structure is that of Fig. 8.7.

ISI and adaptation speeds. Conversely, adaptation speed and EMSE would normally depend quite heavily on the system gain, which is quite unpredictable. To eliminate this dependence (which, just as for the ZF AGC, is in principle linear), exponential VCAs are used instead of linear VCAs in Fig. 8.34 (see Section 8.3.2 and Problem 8.12 for an explanation).

The cross-correlation between e and $\hat{a}_{k-1}$ is used for timing recovery (this is the ZF technique of Example 10.13 and Section 11.7). It is interesting to note that control information for such different system parameters as gain, slope and sampling phase can be extracted in as uniformly a manner as in Fig. 8.7. This uniformity translates into a restriction of required circuit design effort.

The use of exponential VCAs permits certain topology changes to be accounted for within the weighing network. To illustrate this possibility, let us note that it will generally be desirable to normalize signal levels early on in the system, i.e. before the prefilter in Fig. 8.34. This means that we would like to use v_0 and v_2 only for slope correction, and add a VCA with control parameter β for gain control before the prefilter. Denote the new values of v_0 and v_2 by v_0' and v_2'. Clearly $c(v_0) = c(\beta)c(v_0')$ and $c(v_2) = c(\beta)c(v_2')$, i.e. $v_0 = \beta + v_0'$ and $v_2 = \beta + v_2'$.

The ratio of the weighing factors of y and y'', i.e. $c(v_0')/c(v_2') = c(v_0' - v_2')$, defines the degree of slope correction. To be able to control this ratio with one knob, we hardwire v_2' to be inverse of v_0', i.e. $v_2' = -v_0'$. This leaves v_0' and β to be determined. One easily checks that $\beta = (v_0 + v_2)/2$ and $v_0' = (v_0 - v_2)/2$, i.e. the new control parameters are a linear transformation of the original ones. This transformation can be incorporated in the linear weighing network. All that is needed to account for the new topology is a new set of weighing factors (Fig. 8.35). The new weighing matrix $\mathbf{M}'$ retains the uncoupled nature of adaptation across the topology change.

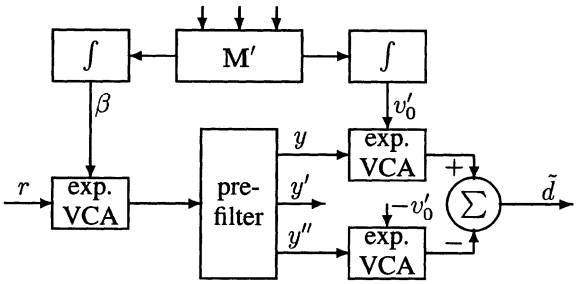


Figure 8.35: Alternative topology for joint VCA and equalizer adaptation based on transformed weighing network $\mathbf{M}'$.

8.6 MINIMUM MEAN-SQUARE ERROR EQUALIZER ADAPTATION

In the MMSE AGC of Section 8.3.5, a single parameter c is adjusted in order to minimize the error e between the AGC output and the desired input d of the detector. In this section we extend this approach to adaptive equalization. Here a total of N parameters is to be adjusted (Fig. 8.36).

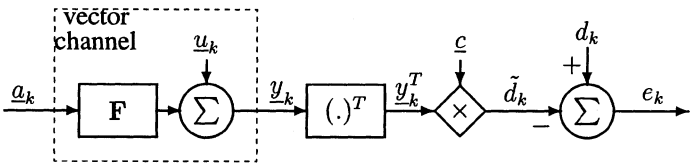


Figure 8.36: The discrete-time vector model that we have encountered earlier for ZF equalization remains applicable for MMSE equalization. The equalizer coefficient vector $\underline{c}$ is now selected in order to minimize the power ε of the error signal e_k.

The MMSE equalization problem is conveniently formulated in the same vector notation that was used for ZF (see Section 8.5). Of particular interest are the vectors $\underline{y}_k$ and $\underline{c}$ of length N that describe the tap signals and coefficients, respectively. Unlike ZF adaptation, details about the 'interior' of the channel (as represented by the matrix $\mathbf{F}$ and vectors $\underline{a}_k$ and $\underline{u}_k$) do directly not come into play. This 'interior' does, of course, leave its fingerprints on the equalizer characteristics (see Problem 8.34).

The error $e_k = d_k - \underline{y}_k^T \underline{c}$ is the difference between the desired equalizer output and the actual output. It has power

$$\varepsilon \triangleq E[e_k^2] = E[d_k^2 - 2d_k \underline{y}_k^T \underline{c} + \underline{c}_k^T \underline{y}_k \underline{y}_k^T \underline{c}] = P_d - 2\underline{p}^T \underline{c} + \underline{c}_k^T \mathbf{R}_y \underline{c}_k \tag{8.37}$$

where $P_d = E[d_k^2]$ is the power of d_k, $\underline{p} \triangleq E[d_k \underline{y}_k]$ is a vector of length N that describes the cross-correlation between desired signal and tap signals, and $\mathbf{R}_y \triangleq E[\underline{y}_k \underline{y}_k^T]$ is the

$N \times N$ autocorrelation matrix of the tap signals. To minimize ε, we set the gradient of ε with respect to $\underline{c}$ equal to zero. The gradient is

$$\underline{\nabla} = -2\underline{p} + 2\mathbf{R}_y\underline{c}. \tag{8.38}$$

The optimum tap vector $\hat{\underline{c}}$ is

$$\hat{\underline{c}} = \mathbf{R}_y^{-1}\underline{p}, \tag{8.39}$$

and the minimum mean-square error $\varepsilon_{\min}$ is

$$\varepsilon_{\min} = P_d - \underline{p}^T\hat{\underline{c}}. \tag{8.40}$$

8.6.1 Uniqueness of the MMSE equalizer

The solution of (8.39) and (8.40) becomes ill-defined when $\mathbf{R}_y$ is singular or nearly so. In that case many different equalizer settings all yield essentially optimum performance. By way of illustration we consider a transversal equalizer.

Example 8.21 For a T-spaced transversal equalizer, the N tap signals are delayed versions of the equalizer input y_k, and the matrix $\mathbf{R}_y$ has entries $\mathbf{R}_y(i,j) = E[y_{k-i}y_{k-j}] = E[y_k y_{k+i-j}]$, where the last step is valid whenever y_k is stationary. We consider three cases.

1. Let y_k be a sinusoid with unit power and angular frequency ω ($y_k = \sqrt{2}\sin\omega k$). The corresponding autocorrelation matrix $\mathbf{R}_y$ is studied in Problem 8.36. It is found that $\mathbf{R}_y$ is regular for $N = 2$ provided that $\omega \neq n\pi$. Conversely, $\mathbf{R}_y$ is singular irrespective of ω for $N \geq 3$, and MMSE equalizer settings are ill defined. A similar analysis is possible for a combination of sinusoids. For the equalizer to be well-defined, the number of equalizer taps should, in general, not exceed the number of spectral components (including DC) of the equalizer input signal.
2. Let y_k be a binary pulse sequence with a period of J symbol intervals and unit power according to $y_k = \sqrt{J}$ for $k = 0 \bmod J$ and 0 else. Then $E[y_k y_{k-i}] = 1$ for $i = 0, \pm J, \pm 2J, \ldots$ and $E[y_k y_{k-i}] = 0$ otherwise. Two cases may be distinguished.
   1. $J \geq N$: Here $\mathbf{R}_y = \mathbf{I}$. This is a regular matrix.
   2. $J < N$: Here $\mathbf{R}_y$ has off-diagonal components that are nonzero, and $\text{rank}(\mathbf{R}_y) < N$. Thus $\mathbf{R}_y$ is singular, i.e. the MMSE equalizer settings are ill defined.

   This suggests that the span of the equalizer should be smaller than the period of a periodic input signal.
3. Let y_k be uncorrelated with unit power. Then $\mathbf{R}_y = \mathbf{I}$ irrespective of N, i.e. $\mathbf{R}$ is regular irrespective of the number of equalizer taps.

In this example, MMSE equalizer settings are best defined when

1. The equalizer input is as random ('spectrally rich') as possible. In this respect, it is good engineering practice to use a scrambler in the transmitter so as to randomize data and avoid long runs of identical symbols.
2. The equalizer has no more taps than strictly needed.

These statements carry over to virtually any equalizer structure.

In many data receivers, adaptation is put on a decision-directed footing. This gives rise to a second source of non-uniqueness, notably for transversal equalizers. In a decision-directed receiver, the desired data sequence d_k is replaced by a sequence $\hat{d}_k$ that is derived from decisions taken by the receiver. If the equalizer is set optimally then decisions are normally correct so that $\hat{d}_k = d_k$. Assume by way of illustration that the equalizer is T-spaced and has infinitely many taps ($N = \infty$). If we shift all tap settings across K delay stages then the equalizer output remains identical except for an added delay of K symbol intervals. Since $\hat{d}_k$ also experiences a delay of K symbol intervals, the error e_k will remain small. There are, consequently, infinitely many optimum equalizer settings: one for each value of K, i.e. one for each position of the *central equalizer tap*. This non-uniqueness is of no concern when the equalizer is infinitely long, because all values of K will then yield identical performance. In practice, however, N is invariably finite and a shift across K symbol intervals causes some of the tap settings to 'disappear' beyond the span of the delay line. As a result, equalizer performance will deteriorate, though possibly not enough to cause decision errors, especially for small $|K|$. There is then a single optimum position for the central equalizer tap, and $N - 1$ positions that yield sub-optimum performance. To alleviate this problem it is obviously desirable to restrict N. Further relief is possible, for example, by constraining the non-central taps to be smaller in magnitude than the central one [85] (see also Example 8.6 and [68]).

To a larger or smaller extent, these considerations carry over to a variety of other equalizer structures, and again suggest that we take N no larger than strictly needed.

8.6.2 MMSE adaptation

The structure of the MMSE AGC of Fig. 8.17 is easily extended to multi-parameter adaptation (Fig. 8.37). The equalizer tap values are formed by a bank of N integrators. These are excited by N cross-products $\underline{y}_k e_k$ via a weighing network that can be described by an $N \times N$ adaptation matrix $\mathbf{M}$.

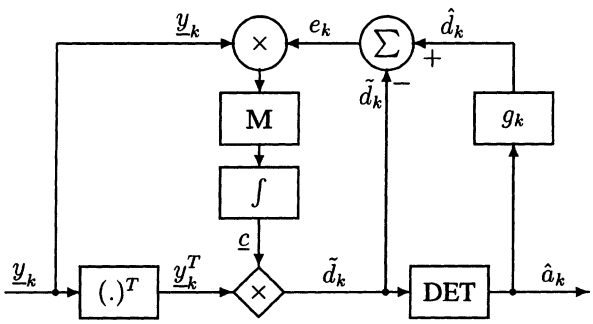


Figure 8.37: Basic structure of MMSE adaptation loop.

Steady-state solution In steady state, the integrator outputs $\underline{c}$ will all be finite (provided, of course, that the loop is stable), i.e. their inputs $\mathbf{M}\underline{y}_k e_k$ cannot contain any DC. If $\mathbf{M}$ is invertible then the adaptation circuit will seek to force the cross-correlation $E[\underline{y}_k e_k]$ toward zero. This cross-correlation is easily evaluated when decision errors are absent, so that $\hat{d}_k = d_k$. In that case

$$E[\underline{y}_k e_k] = E[\underline{y}_k(d_k - \underline{y}_k^T \underline{c})] = \underline{p} - \mathbf{R}_y \underline{c}. \tag{8.41}$$

Apart from a constant this is just the gradient of ε with respect to $\underline{c}$ (compare (8.38)). In effect, the loop continuously forms new gradient estimates and forces their average toward zero so as to minimize MSE. The discrete-time form of this adaptation scheme is usually called LMS (Least Mean Square) [88]. (Strictly speaking, it is only called LMS when the adaptation matrix is of the degenerate form $\mathbf{M} = \mu\mathbf{I}$; for other matrices it is 'similar to LMS' [88].) LMS minimizes MSE in an iterative fashion, driven by noisy gradient estimates. This approach is useful in many other applications besides adaptive equalization [88]. Just as in Section 8.5, the weighing network $\mathbf{M}$ provides design freedom that can be used to decouple adaptation. Problem 8.39 is concerned with an alternative decoupling technique.

Theorem 8.2 *Let the weighing network have adaptation matrix*

$$\mathbf{M} = K_t \mathbf{R}_y^{-1}, \tag{8.42}$$

where K_t is a positive constant. Then there is no cross coupling between coefficients and all loop gains are equal to K_t.

Proof: With (8.41), the integrator input is seen to have expected value

$$\mathbf{M}E[\underline{y}_k^T e_k] = \mathbf{M}(\underline{p} - \mathbf{R}_y \underline{c}) = \mathbf{M}\mathbf{R}_y(\mathbf{R}_y^{-1}\underline{p} - \underline{c}) = \mathbf{M}\mathbf{R}_y(\hat{\underline{c}} - \underline{c}). \tag{8.43}$$

This input is integrated to obtain the coefficient vector $\underline{c}$ (Fig. 8.38). The loop of Fig. 8.38 obviously becomes uncoupled when $\mathbf{M}\mathbf{R}_y = K_t\mathbf{I}$. In that case all sub-loops have gain K_t. □

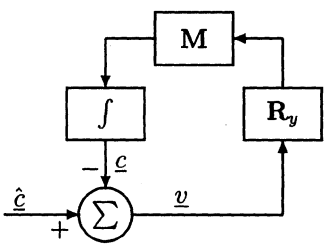


Figure 8.38: Model describing mean behavior of $\underline{c}$ for the MMSE adaptation loop of Fig. 8.37.

This theorem is the counterpart of Theorem 8.1 for ZF adaptation. A loop that satisfies (8.42) is said to be *fully decoupled* or *orthogonalized*. An orthogonalized loop essentially carries out a search based on Newtons method [36, 60, 88].

8.6.3 Properties of the orthogonalized MMSE loop

In effect, this loop consists of N independent MMSE AGCs that are operating in parallel. All have the same time constant $\tau = 1/K_t$, and are stable for $K_t < 2$ in the absence of implicit or explicit loop delays (see Section 8.3.7 for a review of the effect of loop delays). Each sub-loop contributes the same amount of gradient noise to $\tilde{d}_k$. In practice these contributions will add in power. Then the total gradient noise power σ_{gr}^2 is N times as large as for a single AGC as given by (8.18), i.e.

$$\sigma_{\text{gr}}^2 \simeq N B_l \varepsilon_{\min}, \tag{8.44}$$

where B_l is the noise bandwidth of the loop. This corresponds to a misadjustment $M = \sigma_{\text{gr}}^2/\varepsilon_{\min} = NB_l$. For loop gains of practical interest we have $B_l \simeq 2/K_t$, and we may attribute an efficiency factor

$$\gamma \triangleq \frac{1}{\tau M} \simeq \frac{2}{N} \tag{8.45}$$

to the loop. Loop efficiency decreases in proportion to the number of adjusted parameters. A comparison with (8.36) reveals that MMSE adaptation is at least as efficient as ZF adaptation. The efficiency of (8.45) cannot be bettered by any other adaptation technique [88, AppendixA].

Exercise 8.12 Assume that a 16-tap equalizer is controlled by means of a fully decoupled MMSE loop that is dimensioned to have a time constant τ of 100 symbol intervals. How much does the MSE increase due to gradient noise? What time constant τ is attainable if we insist that this degradation be at most 0.1 dB?

8.6.4 Coupled MMSE adaptation

The use of an $N \times N$ weighing network may be undesirable, especially when N is large. It is common practice to use a set of N identical weight factors μ instead, one for each tap. This corresponds to the classical LMS algorithm [88]. Here the adaptation matrix is $\mathbf{M} = \mu\mathbf{I}$, and the effective loop gain is $K_t = (\mu/N)P_{\underline{y}}$, where $P_{\underline{y}} = E[\underline{y}_k^T \underline{y}_k] = \text{tr}[\mathbf{R}_y]$ represents the total power of the tap input signals. A loop of this type carries out out a *steepest-descent* search. The coefficient vector $\underline{c}$ is continuously moved opposite to the gradient of e^2 with respect to $\underline{c}$. As a result, the error initially tends to decay rapidly, but because of coupling may take long to fully settle down.

Example 8.22 The classical LMS loop has integrator input $\mu e_k \underline{y}_k$. Except for noise, this is the same input as for the MMSI ZF loop of Example 8.17. The behavior of both loops will be virtually identical when noise is small.

Analysis of LMS can be moulded in a similar form as the AGC analysis (see Appendix 8E). As a result of coupling, coefficient convergence may slow down considerably in certain directions of the N-dimensional tap space. These are, crudely speaking, the directions in which little signal energy is present to drive the taps. As such, these directions do not contribute much to the MSE. For this reason convergence of the MSE doesn't slow down nearly as much (this effect appears to have been first noticed by

Ungerboeck [79]; a related phenomenon is discussed in the next section), and (8.45) remains a reasonable indicator of the efficiency of the algorithm whenever coupling is not too strong.

Example 8.23 In Example 8.18 we have considered a T-spaced full-response equalizer that is adapted under the control of an orthogonalized PZF loop (Fig. 8.31). Fig. 8.39 depicts the MMSE counterpart of this equalizer.

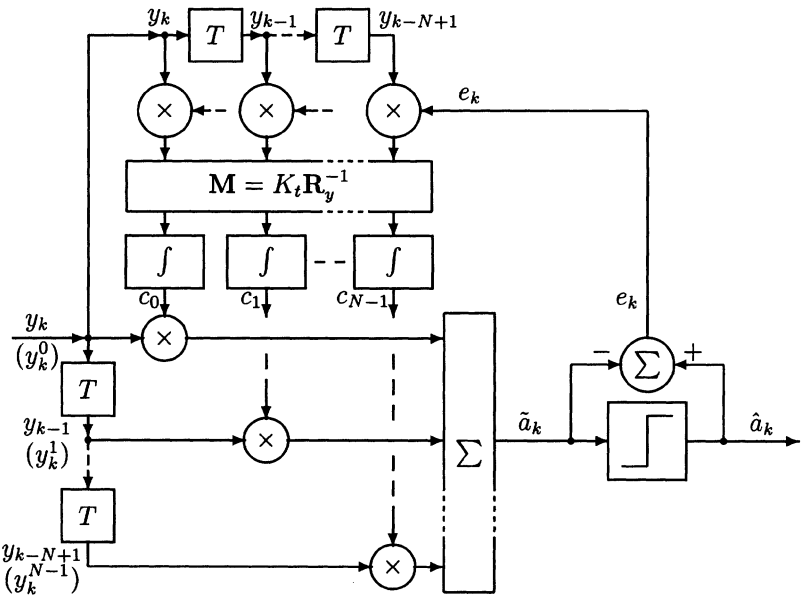


Figure 8.39: Adaptive T-spaced full-response equalizer. Adaptation involves an orthogonalized MMSE loop.

The tap signals y_{k-i} rather than delayed decisions $\hat{a}_{k-i}$ are now cross-correlated with the error signal. This will increase complexity of a digital implementation because y_k is a multi-bit signal, whereas decisions are typically binary. Unlike Fig. 8.31, the error e_k is not delayed. In practice a certain delay may be unavoidable for implementation reasons. In that case the tap signals must be delayed accordingly before being correlated. The decision-directed nature of the scheme causes the central tap position to be non-unique (see Section 8.6.1). Replacement of the adaptation network by N scalar weights μ ($\mathbf{M} = \mu\mathbf{I}$) leads to the classical full-response T-spaced LMS equalizer, which takes center stage in the adaptive equalization literature [66, 71].

Example 8.24 Let us again consider the $1 - \mathcal{D}$ system of Examples 8.13 and 8.19 when equipped with a 15-tap T-spaced equalizer. Fig. 8.40 shows learning curves for coupled and decoupled MMSE adaptation ($\mathbf{M} = \mu\mathbf{I}$ and $\mathbf{M} = K_t\mathbf{R}_y^{-1}$, respectively). As before, both loops are dimensioned for a nominal EMSE of 0.5 dB. The actually measured EMSE is slightly larger. MMSE settings of a 15-tap equalizer are not quite equal to the ZF settings for the channel at hand. As a result, the MSE floor in Fig. 8.40 is around 0.2 dB lower than that of Figs. 8.21 and 8.32.

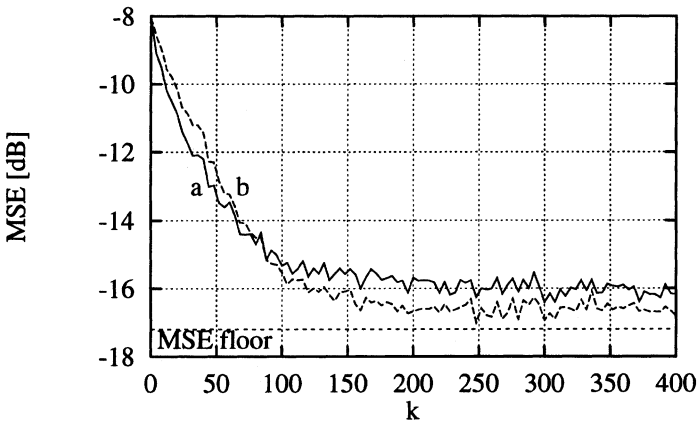


Figure 8.40: Learning curves for two MMSE adaptation loops with a nominal steady-state EMSE of 0.5 dB. a. Coupled; b. Orthogonalized.

The coupled loop initially converges more rapidly than the decoupled one, but takes much longer to fully settle down. Even so, the effect of decoupling is clearly smaller than for ZF (compare Fig. 8.32). This corroborates the heuristic argumentation above. In accordance with the efficiency estimates of (8.36) and (8.45), the decoupled ZF and MMSE loops have virtually identical convergence characteristics.

8.6.5 Blow-up of coefficients

Although MMSE adaptation is basically quite robust, it does exhibit some undesirable properties. One of these may show up when $\mathbf{R}_y$ is ill-conditioned. This is so, crudely speaking, when the equalizer input signal has very little or no content in a part of the spectrum (see also Section 8.6.1). The fractionally-spaced equalizer (FSE) is a typical case in point. The fundamental interval of this equalizer extends beyond the Nyquist frequency, and the upper portion of it is often more or less 'empty'. The equalizer transfer function C can be more or less arbitrary in this band because $0.C = 0$ for all C. At the same time, the MMSE loop cannot extract meaningful control information here because none is present. The net result is that C may float around freely without this being visible in the error signal. A particular danger is that $|C|$ becomes very large. In that case tap values may blow up beyond the range that can be represented in hardware. Small offsets in the loop, in fact, virtually guarantee that this will happen [14], as we have seen for the MMSE AGC (Section 8.3.7). There are essentially three possible cures:

1. Circuitry can be added that allows coefficients to saturate. This approach has at least one disadvantage: the more coefficients are saturated, the fewer degrees of freedom remain to manipulate the information-bearing portion of the spectrum. This may force coefficients to be reset from time to time.

2. Ideal integrators can be replaced by leaky ones. This will drive the transfer function of the equalizer toward zero in the concerned band, thereby avoiding overflow (see also Section 8.3.7).

3. Background noise can be added to the tap signals. This will regularize $\mathbf{R}_y$ and define the desired transfer function ($C = 0$) sufficiently accurately, possibly at some cost to MSE.

The above argumentation is rather heuristic but can be cast in a more rigorous form. We refer to [93], [32] and [81] for an analysis and for more details on the above cures.

8.6.6 Sign algorithm

Rather than the power of e we may seek to minimize the average amplitude $E[|e|]$ of e. Now

$$\frac{\partial E[|e|]}{\partial \underline{c}} = E[\mathrm{sgn}(e)\frac{\partial e}{\partial \underline{c}}] = -E[\mathrm{sgn}(e)\underline{y}_k]. \tag{8.46}$$

The first equality is due to the chain rule and the fact that $d|x|/dx = \mathrm{sgn}(x)$. We see that $E[|e|]$ may be minimized by forcing the cross-correlation between $\mathrm{sgn}(e)$ and $\underline{y}$ toward zero. The corresponding adaptation loop is depicted in Fig. 8.41. It is identical to the MMSE loop of Fig. 8.37 except for the sign operation on e_k. In practice the adaptation network normally degenerates into a set of N parallel scaling factors μ, one for each tap ($\mathbf{M} = \mu\mathbf{I}$).

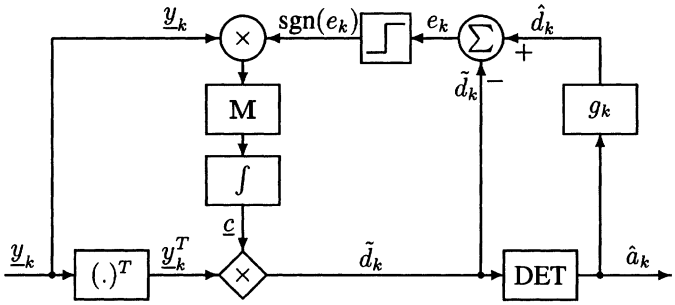


Figure 8.41: Adaptation loop based on the sign algorithm. This loop minimizes the average amplitude of the error signal e.

Multiplication of $\underline{y}_k$ by $\mathrm{sgn}(e_k)$ involves only sign reversals when done digitally, and is thereby significantly simpler than for the MMSE loop of Fig. 8.37. On the debit side, it takes much longer for the loop to eliminate large initial misadjustments. This is due to the fact that the magnitude of the misadjustment is no longer visible after the sign operator. To overcome this problem one sometimes resorts to time-varying loop gains [46, 51]. If loop gain is set for a given steady-state EMSE, then small misadjustments will be tracked at the same rate as for the MMSE loop. However, as misadjustments grow, rates of adaptation will lag ever further behind those of MMSE adaptation.

The attribute 'sign' (or *sign algorithm*) is used for any adaptation loop in which a sign operation takes place. Thus the MMSE loop of Fig. 8.37 would also become a 'sign'-loop if the components of either $\underline{y}_k$ or or the cross-product $\underline{y}_k e_k$ are quantized to one bit. For the ZF loop of Fig. 8.30, 'sign'-adaptation comes only in one taste when decisions $\hat{a}_k$ are binary; we will encounter this taste in Section 8.9. As a rule, stronger quantization within the loop will simplify digital implementation at the cost of degraded adaptation properties. For a more detailed analysis of sign algorithms we refer to [20], [15] and [83].

8.7 COMPARISON OF ZF AND MMSE ADAPTATION

Steady-state solution, adaptation properties Both ZF and MMSE adaptation loops contain a weighing network that determines adaptation properties. For simplicity this network is often cast in the form of a set of N uncoupled scaling factors, one for each equalizer tap. In that case adaptation generally becomes coupled, and the ZF loop will act to force all ISI in a predefined span of length N toward zero. Total ISI and thereby performance will typically be worse than for MMSE adaptation.

For both loops, prior knowledge about the channel can be hardwired into the network in order to decouple adaptation for one reference channel. For ZF, the network can be enlarged to $P \times N$ with $P > N$ in order to minimize ISI power within a span of length P. For sufficiently large P, steady-state performance can approach that of MMSE adaptation if noise is not too large.

MMSE loops have an efficiency factor γ on the order of $2/N$ (see Section 8.3.4 for a definition and interpretation of γ). Decoupled ZF loops can have the same efficiency for sufficiently large P. Here coupling may degrade γ significantly.

ZF convergence properties depend linearly on the channel characteristics. Conversely, MMSE properties depend quadratically on amplitude and do not depend on phase at all. For ZF, the dependence on the amplitude characteristics can in principle be eliminated via an exponential voltage-controlled amplifier as in Example 8.20. For MMSE this technique provides only partial relief (see Problem 8.14). A coarse AGC that precedes the loop may be used instead (this technique is not uncommon in voice-band modems [79]). Alternatively, loop gain may be adapted in response to input signal amplitude. The latter technique is called Normalized LMS [62, 7] and can be thought of as embedding an AGC within the loop. This yields, in essence, an 'adaptive adaptation loop', which requires judicious engineering.

Exercise 8.13 Consider an MMSE loop that is preceded by an AGC. Mention potential advantages and disadvantages of this structure relative to a loop using Normalized LMS. Can the AGC be data-aided?

Robustness MMSE loops intrinsically try to bring the MSE down. They may, however, converge to a local rather than global minimum (for example, to a wrong central tap position for a decision-directed transversal equalizer). Mazo [53] and Macchi [50] provide analyses of decision-directed MMSE equalizer startup. Another artefact of MMSE loops is that coefficients tend to blow up when the autocorrelation matrix of the tap signals is ill-conditioned. Saturation circuitry, integrator leakage and addition of noise constitute possible cures.

Decision-directed ZF loops tend to get stuck in unpredictable corners when the eye before the equalizer is not open at the start of adaptation [48] or when the initial central tap position is incorrect (Problem 8.28). This restricts their application to situations where a reasonable initial guess about the required equalizer settings can be made. Such a guess can be based on data-aided acquisition procedure or on prior knowledge about the channel characteristics.

Implementation MMSE loops are generally more complex than ZF loops. In the digital domain, the main difference is that the multi-bit error signal must be multiplied with N multi-bit tap signals, whereas in ZF P sign reversals (equivalent to multiplications with bits a_k) suffice. Furthermore, formation of the error signal inevitably takes some time, and the tap signals must be delayed accordingly. This is not easy in the analog domain, and is unnecessary for ZF.

Behavior of both loops in the face of nonidealities is fully analogous to that of the corresponding AGCs (see Section 8.3.7). A major difference between ZF and MMSE is that the MMSE loop can become very sensitive to DC offsets.

8.8 ADAPTIVE DETECTION

From Section 8.2.2 we recall that the basic problem in adaptive detection is one of signal estimation. Fig. 8.42, which recapitulates the core of Fig. 8.2, gives an outline.

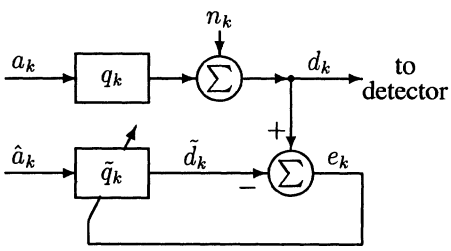


Figure 8.42: In adaptive detection, a filter impulse response $\tilde{q}$ is adapted in order for the filter output $\tilde{d}$ to approach as well as possible a given channel output d.

The adaptive filter has impulse response $\tilde{q}_k$ and is excited by data symbols $\hat{a}_k$ that are ideally correct. The filter output $\tilde{d}_k = (\hat{a} * \tilde{q})_k$ should approximate as well as possible a desired sequence d_k with data and noise component $(a * q)_k$ and n_k, respectively.

Filter adaptation is based on the error $e_k = d_k - \tilde{d}_k$. The basic aim is to minimize the power of e_k (MMSE). It should be noted that the adaptive filter cannot cancel the noise component n_k of d_k in any way (provided, of course, that data and noise are uncorrelated). This means that minimization of MSE is tantamount to minimization of total ISI power (MMSI).[5] This is unlike adaptive equalization, where MMSE and MMSI are different and lead to different adaptation schemes. In practice, both objectives are often equivalent to the elimination of all ISI within a certain span (PZF).

Example 8.25 Assume that the channel impulse response q_k has length N_c such that $q_k = 0$ for $k \notin \{0, \ldots, N_c - 1\}$ and that the adaptive filter is a T-spaced transversal filter with N taps such that $\tilde{q}_k = 0$ for $k \notin \{0, \ldots, N-1\}$. There are two possibilities:

1. $N \geq N_c$: Here the filter is long enough to be able to form a replica of the *entire* channel impulse response. This would eliminate *all* ISI from e_k, leaving only noise. There is clearly no way to make e_k any smaller, i.e. the PZF and MMSE/MMSI solutions coincide.

2. $N < N_c$: Here the filter is too short to model q_k completely, and e_k will contain residual ISI. A plausible strategy is to eliminate all ISI within the span of the filter, i.e. to choose $\tilde{q}_k = q_k$ for $k = 0, \ldots, N-1$ (PZF). This approach will minimize the power of e_k when data a_k is uncorrelated. Conversely, correlation of a_k induces correlation of ISI components and as a result it may become beneficial to allow some ISI within the span $\{0, \ldots, N-1\}$ in exchange for a bigger reduction outside it. Only in this case do the MMSI/MMSE and PZF solutions differ.

A vector model equivalent to the one of Fig. 8.42 is shown in Fig. 8.43. The adaptive filter is transversal and has length N. Its coefficients are collected in a vector $\underline{\tilde{q}} = [\tilde{q}_0, \ldots, \tilde{q}_{N-1}]^T$, and its output $\tilde{d}_k$ is $\tilde{d}_k = \underline{a}_k^T \underline{\tilde{q}}$ where $\underline{a}_k \triangleq [a_k, \ldots, a_{k-N+1}]^T$ is a data vector of length N. Similarly, the channel has output $d_k = \underline{a}_k^T \underline{q} + n_k$ where $\underline{q}_k^T = [q_0, \ldots, q_{N-1}]^T$ is a vector of length N. (The vector notation in previous sections is compatible with the present one if we put $P = N$.) In Case 1 in the above example, the $N - N_c$ last entries of this vector will be zero. In case 2, $N_c - N$ ISI components must be lumped into the noise n_k. The 'augmented' noise remains uncorrelated with the data vector $\underline{a}_k$ provided that data a_k is uncorrelated, as we assume for simplicity.

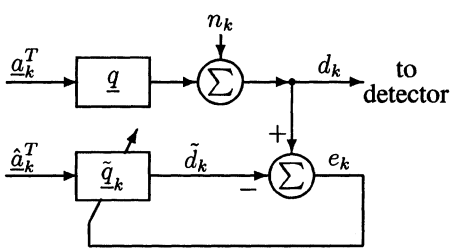


Figure 8.43: Vector model equivalent to the one of Fig. 8.42 for the situation that the adaptive filter is transversal.

[5] We designate any data component of e_k as ISI. The implicit assumption is that the detector is meant for an impulse response $\tilde{q}_k$.

For ZF adaptation, the loop must force all correlation between the error sequence e_k and the data vector $\underline{a}_k$ toward zero (see Section 8.5). For MMSE adaptation, the correlation between e_k and the vector of tap signals must vanish (see Section 8.6). In the case at hand, this vector is just the data vector $\underline{a}_k$, i.e. MMSE and ZF adaptation loops are the same.

Let us consider the MMSE criterion in some more detail. The error $e_k = d_k - \underline{a}_k^T \tilde{\underline{q}}_k$ has power $\varepsilon = E[e_k^2]$. To minimize ε with respect to $\tilde{\underline{q}}$, we must set the gradient of ε with respect to $\tilde{\underline{q}}$ equal to zero. This gradient is $\partial\varepsilon/\partial\underline{p} = E[2e_k \partial e_k/\partial\underline{p}] = -2E[e_k \underline{a}_k^T]$. The latter step is due to the fact that $e_k = d_k - \underline{a}_k^T \tilde{\underline{q}}_k$; since d_k and $\underline{a}_k^T$ do not depend on $\tilde{\underline{q}}$, $\partial e_k/\partial\tilde{\underline{q}}_k = -\underline{a}_k^T$. This confirms that ε is minimized when e_k and $\underline{a}_k$ have no cross-correlation.

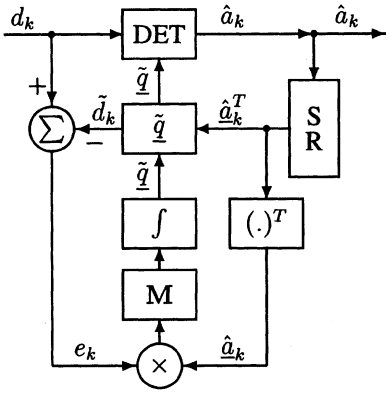


Figure 8.44: Adaptation loop for adaptive detection.

The adaptation loop of Fig. 8.44, a close relative of the one of Fig. 8.30, acts to eliminate all cross-correlation between e_k and $\underline{a}_k$. When data a_k is uncorrelated, optimum decoupling requires an adaptation matrix $\mathbf{M} = \mu\mathbf{I}$ (see Section 8.6), i.e. the adaptation network degenerates into a set of N parallel scaling factors. Since the tap signals have a fixed amplitude and a fixed correlation structure, loop properties are also completely fixed. All this is favorable to the situation in adaptive equalization.

Loop gain K_t is μP_a, where P_a is the power of a_k. For binary data $a_k \in \{-1, 1\}$ we have $P_a = 1$ and $K_t = \mu$. The loop has a time constant $\tau = 1/K_t$ (normalized in symbol intervals T). Noise n_k causes coefficient fluctuations which, in turn, induce gradient noise in e_k of power

$$\sigma_{\text{gr}}^2 \simeq \frac{P_n}{\gamma\tau}, \tag{8.47}$$

where P_n is the power of n and $\gamma = 2/N$ is the efficiency factor of the loop. A derivation of these properties may be found in Section 8.6.

Transversal filters are not the only detector structures that may need to be adapted. We consider three more structures.

8.8.1 Adaptive thresholds

In Fig. 8.45, which is a copy of Fig. 8.3, a noisy and scaled partial-response sequence $\tilde{b}_k$ is detected by means of a ternary threshold detector. For an explanation of this system we refer to Section 8.2.2; here we restrict attention to aspects that are relevant to threshold adaptation.

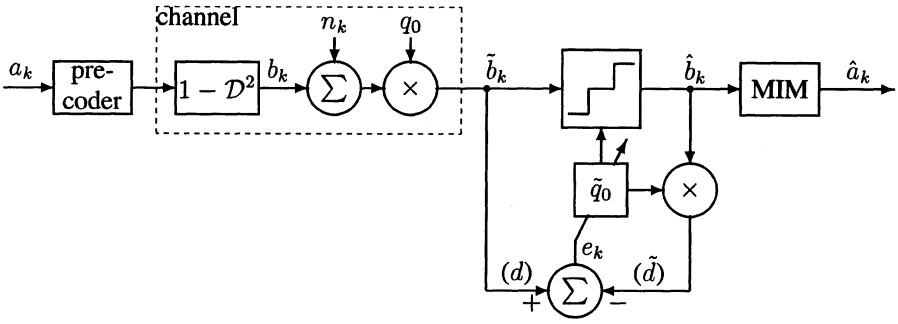


Figure 8.45: Ternary partial-response transmission system with precoder, bit-by-bit detector with adaptive threshold, and memoryless inverse mapping (MIM).

Nominal amplitude levels of $\tilde{b}_k$ are 0 and $\pm 2q_0$. Detection thresholds $\pm\tilde{q}_0$ are ideally located halfway between these levels, i.e. at $\pm q_0$. To approach this ideal, $\tilde{q}_0$ is adjusted on the basis of an error sequence $e_k = \tilde{b}_k - \tilde{q}_0\hat{b}_k$. In the absence of decision errors, this sequence may be written

$$e_k = (b_k + n_k)q_0 - \tilde{q}_0 b_k = (q_0 - \tilde{q}_0)b_k + n_k q_0. \tag{8.48}$$

This shows that e_k consists only of noise when the threshold is adjusted properly. Even when $\tilde{q}_0 \neq q_0$ it consists only of noise whenever $b_k = 0$. This reflects the fact that a ternary zero contains no meaningful amplitude information. To suppress this non-information we multiply e_k by b_k. This yields a sequence $\eta_k \triangleq e_k b_k = (q_0 - \tilde{q}_0)b_k^2 + n_k q_0 b_k$ that is zero half of the time (i.e. whenever $b_k = 0$). The other half of the time $b_k^2 = 4$, i.e. a negative value of η_k indicates that the threshold $\tilde{q}_0$ must be lowered and vice versa. This can be done with an integrator, which also suppresses noise. We arrive at Fig. 8.46.

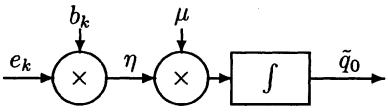


Figure 8.46: Control circuit for threshold adaptation in the system of Fig. 8.45.

In steady state, the output of $\tilde{q}_0$ of the integrator will be finite, and its input must be zero upon average. Thus the circuit of Fig. 8.46 attempts to eliminate any cross-correlation between e_k and b_k. This is exactly what is needed to minimize the power

of e_k. Having identified the circuit of Fig. 8.46 as the heart of an MMSE loop, we can refer back to the previous section for an overview of loop properties.

Exercise 8.14 Verify that the mean-square error $\varepsilon = E[e_k^2]$ is minimized when e_k and b_k have no cross-correlation.

8.8.2 Adaptive DC-compensation

DC offsets can emerge in the detection path of both analog and digital receivers (in the latter case the A-to-D converter at the input of the receiver tends to be a primary cause). Such offsets can severely degrade the bit-error rate, depending on their magnitude and on the type of detector used. The $1 - \mathcal{D}$ and $1 - \mathcal{D}^2$ Viterbi detectors, for example, are completely insensitive to DC-offsets, whereas the $1 + \mathcal{D}$ VD is highly sensitive (see Section 7C). A possible remedy for DC-offsets is a high-pass filter that suppresses the lowest frequencies. Such a filter, however, eliminates useful data energy and also introduces *baseline wander* or low-frequency ISI (see Section 4.3 for a description of this disturbance and its impact on receiver performance). The combined effect of DC-offsets and baseline wander may be undone by means of adaptive DC-compensation (Fig. 8.47).

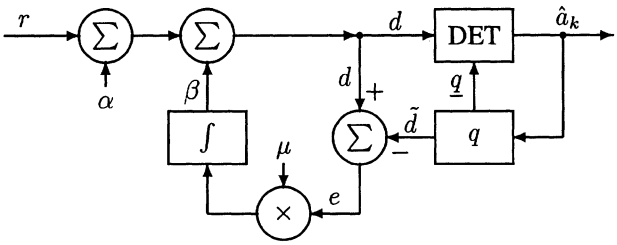


Figure 8.47: DC and baseline wander at the input of a detector may be eliminated with the aid of a loop for adaptive DC-compensation.

In Fig. 8.45, an *a priori* unknown DC offset α emerges in the detection path. We allow α to vary slowly in time so as to cover the effect of baseline wander. The output β of an integrator serves to compensate for α and is added just before the detector. A residual quantity of DC will remain whenever compensation is imperfect. This DC will also be present in the error sequence e_k, which is free from any data content once all other adaptive subsystems have converged. Hence e_k is a good indicator of the residual DC offset, an all that remains needed to close the loop is a scaling factor μ that determines the loop gain K_t. Loop properties are fully analogous to those in previous sections; ZF and MMSE interpretations are again possible.

Exercise 8.15 The actual compensation of DC-offset needs not involve any additions when done via the detection thresholds. Outline how this can be done. Can this approach be combined with that of Section 8.8.1?

For the sake of completeness we mention that baseline wander can also be handled with *quantized feedback* [86]. This is essentially an asynchronous form of decision-feedback equalization (see Chapter 6) that is specifically designed to compensate low-frequency post-cursive ISI. Complexity is typically lower than that of adaptive DC-compensation, but performance tends to be poorer.

8.8.3 Adaptive table look-up filters

In table look-up (TLU) filters, the filter outputs are stored in a tabular form, one for each possible input vector $\underline{a}_k$ (see Section 8.2.5). Filter outputs are read out directly and need not be computed. Arbitrary linear and nonlinear input-output relations can be handled. The TLU version of Fig. 8.42 is shown in Fig. 8.48.

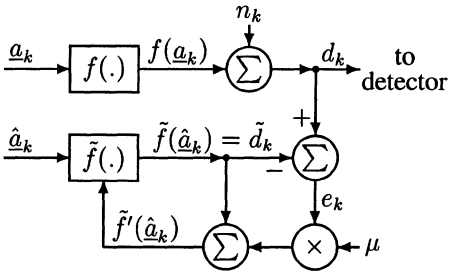


Figure 8.48: Identification of a nonlinear channel with an adaptive table look-up filter.

The desired sequence d_k now has the form $d_k = f(\underline{a}_k) + n_k$, where $f(\underline{a}_k)$ is an arbitrary function of $\underline{a}_k$. Linear functions are a special case; here $f(\underline{a}_k) = \underline{a}_k^T \underline{q}_k$ (compare Fig. 8.43). The TLU is modeled as a nonlinear function $\tilde{f}(\underline{a}_k)$ (compare Fig. 8.6, and note that the shift register in that figure is only needed to form the data vector $\underline{a}_k$). This function is desired to approximate $f(\underline{a}_k)$ as well as possible for every possible data vector $\underline{a}_k$. To this end, the error $e = d - \tilde{d} = f - \tilde{f} + n$ is computed, scaled by a factor μ, and added to the old estimate $\tilde{f}$ to obtain a new estimate $\tilde{f}'$ that replaces $\tilde{f}$ in the table. This is, in effect, a discrete-time integration, which serves to force the average correction term e toward zero. The result is that $\tilde{f}$ will approach f.

It should be noted that only one memory location is updated at every instant k, viz. the one that corresponds to the data vector $\underline{a}_k$ at hand. For uncorrelated binary data, this means that every memory location is, upon average, only updated once every 2^N symbol intervals. By comparison, all N tap values are updated every symbol interval in the transversal filter of Fig. 8.43. On the debit side, gradient noise due to these N updates adds in power, whereas in the TLU only a single parameter is updated. This constitutes only a single source of gradient noise. As a result of these two effects, the efficiency factor γ of the TLU is a factor $N/2^N$ worse than for the transversal filter, i.e.

$$\gamma = \frac{2}{N}\frac{N}{2^N} = 2^{1-N}. \tag{8.49}$$

For a more detailed comparison of adaptive transversal and TLU filters we refer to [28].

Exercise 8.16 Assume that data is binary and uncorrelated, that $N = 5$ and that the excess mean-square error due to gradient noise is to be no larger than 0.1 dB. What adaptation time constants τ are attainable with a transversal filter and with a TLU filter? Answer the same question for $N = 10$.

As N increases, complexity of the TLU filter rapidly becomes unmanageable, and γ rapidly becomes unacceptably small. Problems 8.44 and 8.45 are concerned with techniques to overcome this problem. Another technique, based on the use of Volterra kernels, is discussed in [1]. It is worth noting that all of these approaches can also be used to split a received data signal adaptively into data and noise components. This permits linear and nonlinear transfer characteristics as well as signal-to-noise ratios of the channel to be estimated in a comparatively simple and accurate fashion. All that is needed are digitized versions of the transmitted and received signals [6].

Exercise 8.17 Assume that the detector comprises an adaptive TLU filter, and that DC-offsets emerge in the detection path. Will a DC-correction loop help? Could it make matters worse?

8.9 IMPLEMENTATION EXAMPLE

To illustrate some of the foregoing concepts, we discuss in this section an automatic ZF equalizer that has in several respects been 'stripped to the bone' in order to achieve a data rate of 128 Mb/s with a modest amount of hardware (Fig. 8.49). The equalizer is used in a professional digital video tape recorder according to the D2 standard [59]. Recorders of this type employ the Miller-Squared (M^2) modulation code. This is a binary code with rate $1/2$, i.e. the channel bit rate $1/T = 128$ Mb/s is twice as large as the input bit rate. M^2 has a minimum run-length of 2 symbol intervals and is DC-free (see Example 4.39 for more details).

The equalizer is of the full-response type and is meant to handle slow gain and slope variations of the channel, as induced, for example, by exchange of one brand of tape by another. Because of the high data rate, slope equalization is done in the analog domain by means of a cosine filter according to [44]. This is essentially a transversal filter with a tap spacing of about T seconds and a symmetric impulse response $(-c, 1, -c)$, where c defines the degree of slope correction. Automatic gain control is not used. Instead, a threshold $\tilde{q}_0$ is adapted that defines the system gain. This is a rudimentary form of adaptive detection (see Section 8.8). The output $\tilde{a}$ of the equalizer is nominally equal to $+\tilde{q}_0$ or $-\tilde{q}_0$ at the ideal sampling instants. At these instants, a sign operation yields bit decisions $\hat{a}_k$, and logic comparison of $\tilde{a}$ with $\tilde{q}_0$ yields the sign of the error signal provided that $\hat{a}_k = +1$; this sign is invalid when $\hat{a}_k = -1$. The two bits of information thus produced suffice for control of slope and gain when $\tilde{a}_k$ is DC-free.

Gain and slope control involve two greatly simplified ZF loops. The cost of these simplifications is a large but acceptable slowdown of adaptation. In Section 8.5.2 (eq. (8.31)) we have seen that the cross-correlation between e_k and a_{k-i} is a measure of the difference between the actual and desired values of the ith component q_i of the system

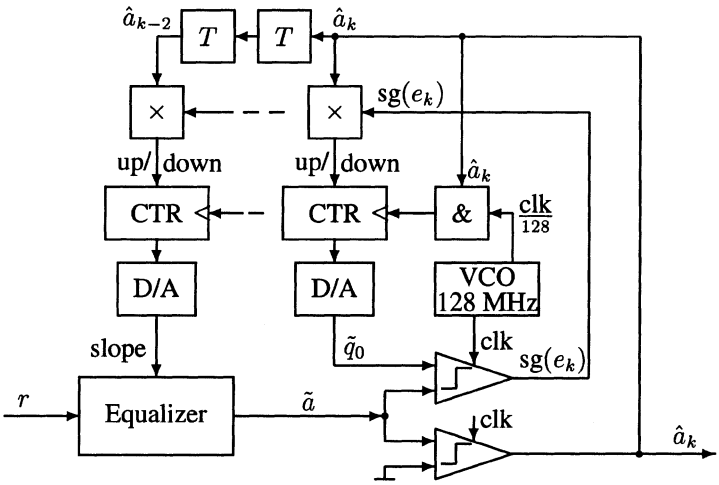


Figure 8.49: Conceptual model of automatic equalizer for digital magnetic video recording at 128 Mb/s. By permission from S. Mita [59].

response, provided that data a_k is uncorrelated. For a Nyquist-1-like system response, gain variations manifest themselves mainly in q_0, while slope errors are mainly visible in q_{-1} and q_1 and to a lesser extent in q_{-2} and q_2. The $d = 1$ constraint introduced by M^2 leaves q_{-1} and q_1 essentially free (see Example 5.7 and Section 5.5) and to profit from this freedom it is preferable to base slope control on q_{-2} and/or q_2. In the scheme of Fig. 8.49 only the latter component is used. Thus threshold adaptation and slope control are based on the cross-products $\text{sgn}(e_k)\hat{a}_k$ and $\text{sgn}(e_k)\hat{a}_{k-2}$, respectively. Formation of these products involves simple operations (logic comparison, bit delay, exclusive-or for bit multiplication) that are manageable at 128 Mb/s. Control information is only present in the DC components of the cross-products and this permits a sampling-rate reduction into a friendlier domain. Once every 128 symbol intervals T, the products are clocked into a digital up/down counter provided that $\text{sgn}(e_k)$ is valid, i.e. provided hat $\hat{a}_k = +1$. (In reality, downsampling is done via latches that have been omitted from Fig. 8.49 for simplicity.) The up-down counters in Fig. 8.49 perform the required integration. They are implemented cheaply in a microprocessor that is also used for control of push buttons, indicator lamps and the like. The counters are reset periodically, and their accumulated value q at the end of a period is compared with positive and negative thresholds q_- and q_+, respectively. If $q > q_+$ then the parameter that is controlled is decreased by one discrete step, if $q < q_-$ then it is increased.[6] Both slope and threshold are quantized rather coarsely so as to simplify D-to-A conversion and equalizer hardware.

This example illustrates that the 'standard' adaptation loops discussed earlier may be simplified substantially when adaptation speeds are not critical. The result, in this

[6]This variant of ZF adaptation finds its roots in work of Lucky [48].

case, is a compact and cost-effective automatic equalizer that functions satisfactorily at high data rates.

8.10 FURTHER READING

Automatic gain control is probably the earliest example of the use of adaptive techniques in electronic systems [64]. Properties of the AGC with exponential voltage-controlled amplifier were well understood by the early sixties [84]. An important later contribution is [63]. Reviews of amplitude control may be found in [55] and [57, Chapter 7].

The LMS algorithm was first described by Widrow and Hoff in 1960 [87], and has since become the cornerstone of a diverse array of applications, including echo cancellation, noise cancellation, predictive coding of speech, and antenna arrays (see [16] for an overview of these applications). LMS was not applied in equalization until the early seventies [27, 49, 65], well after the pioneering work of Lucky on zero-forcing equalizer adaptation [48]. Since then, great strides were made in the understanding of the general properties of LMS [88], of its behavior in adaptive equalization [79, 52, 31], and of digital implementation issues [29, 31, 20, 8]. The use of integrator leakage with LMS to temper coefficient growth for ill-conditioned inputs was proposed in '73 [93], and soon became a valued ingredient of adaptive fractionally-spaced equalizers [81, 32]. Early work in adaptive equalization dealt only with the T-spaced full-response equalizer. Later on, alternative filter structures were considered [37, 73, 40, 15] as well as other receiver types. Early contributions in the latter area include [26], [22], [67] and [80].

Most of the above subjects have been addressed in smaller or greater detail in this chapter. Omitted topics include:

1. Techniques for fast adaptation startup with the aid of a training sequence. Of particular interest are procedures that are based on cyclic sequences [61, 70, 58].

2. Gear-shift techniques. Here the adaptation parameter μ is not fixed, but is instead changed from a large initial value to a smaller steady-state value. The aim is to combine rapid initial convergence with low steady-state gradient noise levels. Evolution of μ over time can either be pre-programmed [71] or adaptive [39, 46, 51, 47].

3. Self-orthogonalizing algorithms. In Sections 8.5.2 and 8.6 we have discussed the use of a fixed decoupling matrix M to achieve optimum convergence properties even when filter inputs are correlated. Approaches of this kind have been proposed in the early seventies for use in partial-response systems [10, 60]. In 1974, Godard described an application of the Kalman filter to channel equalization [34] that was later identified as a recursive form of least-squares estimation [23]. In essence, recursive least-squares (RLS) algorithms dynamically estimate

the optimum decoupling matrix M from the filter inputs, and are as such self-orthogonalizing [30]. Potential drawbacks are a high computational complexity and poor numerical properties [14]. In spite of a large and continuing research effort, these two drawbacks have not yet been overcome to an extent that permits widespread application of RLS. Various textbooks (such as [66, 41, 19]) give a good overview of the state of the art in this field.

4. Decision-directed equalizer startup. This is a difficult subject, and published studies of it are rare [53, 50].

5. Digital implementation issues. Aspects of these issues are treated in various papers [8, 14, 15, 20, 29, 31], [71, 82].

6. Analog implementation issues. Ventures into this important realm are rare to say the least. Among the exceptions is [85].

Adaptive signal processing has grown into a field that cannot possibly be covered in a single book, let alone in a single chapter. A further study may be based on textbooks such as [2], [4], [19], [38], [41] and [89]. For a review of adaptive equalization with an emphasis on LMS-like algorithms we recommend the excellent tutorial by Qureshi [71].

APPENDIX 8A

Analysis of the non-data-aided closed-loop AGC of Fig. 8.9

To keep derivations manageable we restrict attention to the tracking mode of operation. Here c fluctuates around an expected value $\bar{c}$ according to $c = \bar{c} + v$, where v is the deviation of c from $\bar{c}$. To determine $\bar{c}$ we note that the integrator input will vanish upon average, whence $E[x^2] = \bar{c}^2 P_r = P_d$. Correspondingly

$$\bar{c} = \sqrt{\frac{P_d}{P_r}}, \tag{8A.1}$$

just as for the open-loop AGC of Fig. 8.8. We shall analyze AGC properties for the situation that time is discrete (a similar analysis is possible for the continuous-time case). Here the error sequence $\eta_k = P_d - x_k^2 = P_d - r_k^2 c_k^2$ determines the update of c_k according to

$$c_{k+1} = c_k + \mu\eta_k = c_k + \mu(P_d - r_k^2 c_k^2). \tag{8A.2}$$

Now $c_k = \bar{c} + v_k$. Upon subtracting $\bar{c}$ from both sides of (8A.2) we get the recursion $v_{k+1} = v_k + \mu(P_d - r_k^2 c_k^2)$. In the tracking mode of operation, deviations v will remain small, whence $c_k^2 = (\bar{c} + v_k)^2 \simeq \bar{c}^2 + 2\bar{c}v_k$. Hence to a good approximation

$$v_{k+1} = v_k + \mu(P_d - r_k^2(\bar{c}^2 + 2\bar{c}v_k)) = (1 - 2\mu\bar{c}r_k^2)v_k + \mu(P_d - r_k^2\bar{c}^2). \tag{8A.3}$$

Stability, mean tracking rate From (8A.3), the expected value of v_k evolves according to $E[v_{k+1}] = (1-2\mu\bar{c}P_r)E[v_k] + \mu(P_d - P_r\bar{c}^2)$. As a consequence of the definition of v_k, this mean deviation should vanish as $k \to \infty$ whenever the algorithm is stable. This is only possible when the second term of the last right hand side is zero. Hence $P_d = P_r\bar{c}^2$, in accordance with (8A.1). In order for the algorithm to be stable, the first term of the last right hand side should decrease in magnitude as a function of k. This is so whenever $\mu\bar{c}P_r < 1$, i.e. whenever $\mu\sqrt{P_rP_d} < 1$. The steady-state tracking rate amounts to $20\log_{10}(1 - 2\mu\bar{c}P_r) = 20\log_{10}(1 - 2\mu\sqrt{P_rP_d}) \simeq -40\mu\sqrt{P_rP_d}/\ln 10$ dB/iteration, where the approximation applies when the algorithm is well within the stability range. This corresponds to a total loop gain $K_t = 2\mu\sqrt{P_rP_d}$.

Steady-state coefficient fluctuations By (8A.3),

$$\begin{aligned} v_{k+1}^2 &= (1 - 4\mu\bar{c}r_k^2 + 4\mu^2\bar{c}^2r_k^4)v_k^2 + \mu(1 - 2\mu\bar{c}r_k^2)(P_d - r_k^2\bar{c}^2)v_k \\ &\quad + \mu^2(P_d^2 - 2P_dr_k^2\bar{c}^2 + r_k^4\bar{c}^4). \end{aligned} \tag{8A.4}$$

In steady state, the expected value of v_k^2 no longer depends on k, i.e. $E[v_{k+1}^2] = E[v_k^2] \triangleq \sigma_v^2$. To be able to evaluate the expected value of (8A.4), we assume that r_k and v_k have no cross-correlation.[7] Furthermore, we denote $E[r_k^4] = \kappa E[r_k^2]^2 = \kappa P_r^2$ where $\kappa \geq 1$ is the *kurtosis* of r_k. In particular for binary and Gaussian r_k, we have $\kappa = 1$ and $\kappa = 3$, respectively (see Problem 8.6). Upon using (8A.1) and the fact that $E[v_k] = 0$ in (8A.4), we find that

$$\sigma_v^2 = (1 - 4\mu\sqrt{P_rP_d} + 4\mu^2\kappa P_rP_d)\sigma_v^2 + \mu^2P_d^2(\kappa - 1). \tag{8A.5}$$

Correspondingly

$$\sigma_v^2 = \frac{\mu P_d^2(\kappa - 1)}{4\sqrt{P_rP_d} - 4\mu\kappa P_rP_d} = \frac{\mu P_d^2(\kappa - 1)}{4\sqrt{P_rP_d}}\frac{1}{1 - \mu\kappa\sqrt{P_rP_d}}. \tag{8A.6}$$

This shows that coefficient fluctuations vanish completely when r_k is binary, i.e. when $\kappa = 1$. Typically the algorithm will be dimensioned to be well within the stability range. In that case $\mu\kappa\sqrt{P_rP_d} \ll 1$, whence $\sigma_v^2 \simeq \mu P_d^2(\kappa - 1)/[4\sqrt{P_rP_d}]$.

Exercise 8.18 Determine the efficiency factor $\gamma = P_n/(\tau\sigma_{\text{gr}^2})$ of the algorithm, where τ is the adaptation time constant (expressed in symbol intervals T) and $\sigma_{\text{gr}}^2 = P_r\sigma_v^2$ is the gradient noise variance.

[7] This so-called *independence assumption* is usually made in analyses of this type [89, 41]. It is justified *a posteriori* by the fact that the theoretical results agree well with practice.

APPENDIX 8B

Effect of a nonlinear voltage-controlled amplifier on adaptation properties

Figs. 8.16 and 8.19 pertain to AGCs with linear VCA. For a nonlinear VCA with gain control characteristic $c(v)$ we arrive at the model of Fig. 8B.1.a. This model pertains to the ZF DA AGC but becomes valid for the MMSE DA AGC if we replace c^* by $\hat{c}$.

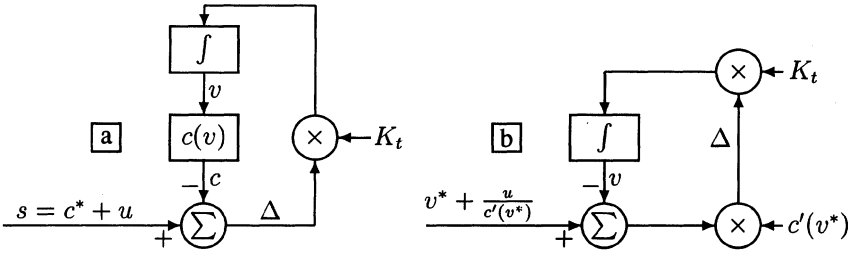


Figure 8B.1: Simplified loop model of AGC with nonlinear voltage-controlled amplifier (a.) and linearized version (b).

The integrator output v is passed through the nonlinearity $c(v)$ which defines the AGC gain c. In steady state, the error Δ will be small, and c will fluctuate about c^*. For simplicity we assume that $c(v)$ is monotonous and non-decreasing. In that case v will fluctuate about a unique value v^* for which $c(v^*) = c^*$. Since fluctuations are small, we may linearize $c(v)$ around $v = v^*$. This yields

$$c(v) \simeq c(v^*) + (v - v^*)c'(v^*) = c^* + (v - v^*)c'(v^*) \tag{8B.1}$$

where $c'(v^*)$ is the derivative of $c(v)$ in $v = v^*$. The error Δ becomes

$$\Delta = c^* + u - c(v) \simeq c'(v^*)[v^* + \frac{u}{c'(v^*)} - v]. \tag{8B.2}$$

This leads to Fig. 8B.1.b. Total loop gain has increased by a factor $c'(v^*)$. Gradient noise has decreased by the same factor. As a result, the quality factor γ of the algorithm has not changed, i.e. a given adaptation time constant τ still yields the same misadjustment M and EMSE as with a linear VCA.

APPENDIX 8C

Decoupled ZF adaptation

In this appendix we examine the consequences of coefficient decoupling on the convergence of the coefficient vector $\underline{c}$ and the system response error vector $\underline{\Delta}$. As $\underline{c}$ converges towards its expected steady-state value $\underline{c}^*$, $\underline{\Delta}$ converges towards a response $\underline{\Delta}^*$

that defines steady-state ISI. As the deviation $\underline{c}' \triangleq \underline{c} - \underline{c}^*$ vanishes, so does $\underline{\Delta}' \triangleq \underline{\Delta} - \underline{\Delta}^*$. We will show that full coefficient decoupling causes both $\underline{c}'$ and $\underline{\Delta}'$ to decay in a desirable manner. For simplicity we restrict attention to the average trajectories $\overline{\underline{c}}'$ and $\overline{\underline{\Delta}}'$ of $\underline{c}'$ an $\underline{\Delta}'$, and we take time to be discrete (the continuous-time analysis is completely analogous). In that case the integrator input is the increment of the output $\underline{c}$, i.e. $\underline{c}_{k+1} - \underline{c}_k = \mathbf{M}\underline{a}e_k$ (see Fig. 8.30). Now $\underline{a}e = \underline{\Delta}$ upon average (see (8.31) with $\mathbf{R}_a = \mathbf{I}$), while $\underline{\Delta} = \underline{g} - \mathbf{F}^T\underline{c}$. Hence

$$\overline{\underline{c}}_{k+1} - \overline{\underline{c}}_k = \mathbf{M}\overline{\underline{\Delta}}_k, = \mathbf{M}\underline{g} - \mathbf{M}\mathbf{F}^T\overline{\underline{c}}_k, \tag{8C.1}$$

where the overbar denotes averaging across data and noise ensembles. For $\mathbf{M}\mathbf{F}^T = K_t\mathbf{I}$ this becomes

$$\overline{\underline{c}}_{k+1} - \overline{\underline{c}}_k = -K_t\overline{\underline{c}}_k + \mathbf{M}\underline{g}. \tag{8C.2}$$

This is a set of N uncoupled first-order recursions, i.e. there is no longer any coupling between coefficients. The left hand side vanishes as $k \to \infty$, and so must the right hand side, i.e. $K_t\underline{c}^* = \mathbf{M}\underline{g}$. In terms of $\overline{\underline{c}}' = \overline{\underline{c}} - \underline{c}^*$ we have the recursion $\overline{\underline{c}}'_{k+1} - \overline{\underline{c}}'_k = -K_t\overline{\underline{c}}'_k$. This shows that the coefficient deviation $\overline{\underline{c}}'$ decays exponentially according to

$$\overline{\underline{c}}'_k = (1 - K_t)^k\overline{\underline{c}}'_0. \tag{8C.3}$$

The rate of decrement is the same in all directions and is independent of the initial deviation $\overline{\underline{c}}'_0$. It is easy to verify that this would not be so if coefficients were coupled. In that case coefficient vectors might decay much more slowly in certain directions than in other ones.

We next focus on $\underline{\Delta}$. Convergence of $\underline{\Delta}$ is perhaps more interesting than that of $\underline{c}$ because $\underline{\Delta}$ determines ISI and thereby transmission quality. We recall that $\underline{\Delta} = \underline{g} - \mathbf{F}^T\underline{c}$, so that $\underline{\Delta}^* = \underline{g} - \mathbf{F}^T\underline{c}^*$. Consequently $\underline{\Delta}' = -\mathbf{F}^T\underline{c}'$. Premultiplication of (8C.3) by $-\mathbf{F}^T$ shows that

$$\overline{\underline{\Delta}}'_k = (1 - K_t)^k\overline{\underline{\Delta}}'_0. \tag{8C.4}$$

Hence $\underline{\Delta}$ decays in the same desirable manner as $\underline{c}$. As a result, ISI decreases and transmission quality improves at a constant rate irrespective of the initial value of $\underline{\Delta}$.

APPENDIX 8D

Behavior of single-tap ZF MMSI loop

For $N = 1$ the equalizer has only a single tap and degenerates into an AGC. In this appendix we analyze this AGC and compare it to the ZF DA AGC of Section 8.3.4. Just as in Fig. 8.13, we assume that the system response q_k is a scaled version of the desired response g_k according to $q_k = \lambda g_k$, where λ is the gain of the channel. Channel noise n_k is scaled by λ before ending up in the equalizer input y_k. Without loss of generality we take g_k to be causal such that $g_k = 0$ for $k \notin \{0, \ldots, M_c - 1\}$, where M_c denotes

the memory length of g_k and should not be confused with the misadjustment M or the adaptation matrix $\mathbf{M}$. We can then take the span S_P to start at zero ($S_P = \{0, \ldots, P-1\}$). For $P \geq M_c$, $\underline{g}$ covers all nonzero components of g_k, i.e. $\check{d}_k \triangleq \underline{a}_k^T \underline{g}$ coincides with the desired sequence $d_k = (a * g)_k$, and the same applies to their powers $P_{\check{d}}$ and P_d. Conversely, $\check{d}_k \neq d_k$ and $P_{\check{d}} < P_d$ when $P < M_c$.

For $N = 1$, matrices $\mathbf{F}$ and $\mathbf{M}$ become row vectors of length P. Specifically, $\mathbf{F} = \lambda \underline{g}^T$, whence

$$\mathbf{F}\mathbf{F}^T = \lambda^2 \underline{g}^T \underline{g} = \lambda^2 P_{\check{d}} \quad \text{and} \quad \mathbf{M} = K_t(\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F} = \frac{K_t}{\lambda P_{\check{d}}} \underline{g}^T. \tag{8D.1}$$

Thus the integrator in Fig. 8.30 has input

$$\mathbf{M}\underline{a}_k ek = \frac{K_t}{\lambda P_{\check{d}}} \underline{g}^T \underline{a}_k e_k = \frac{K_t}{\lambda P_{\check{d}}} \check{d}_k e_k. \tag{8D.2}$$

The cross-product $\check{d}_k e_k$ is similar to the one on which adaptation is based in Fig. 8.14. Two cases are of interest:

1. $P \geq M_c$: Here $\check{d}_k = d_k$ and $P_{\check{d}} = P_d$. Upon identifying the adaptation parameter μ in Fig. 8.14 as $K_t/(\lambda P_d)$ we conclude that the loop at hand is equivalent to the ZF DA AGC of Fig. 8.3.4.

2. $P < M_c$: Here $e_k \check{d}_k \neq e_k d_k$, i.e. there is no equivalence. Gradient noise will increase. In order to analyze gradient noise we distinguish between the actual error sequence e_k including gradient noise, and the steady-state error sequence $\dot{e}_k$ that would arise if coefficient fluctuations (and hence gradient noise) were absent. For the expected steady-state AGC gain $c^* = 1/\lambda$, $\dot{e}_k$ consists purely of noise n_k, i.e. the power ε of $\dot{e}_k$ will be equal to the noise power P_n. As a result of $\dot{e}_k$, the integrator input will contain noise

$$\nu_k = \mathbf{M}\underline{a}_k \dot{e}_k = \frac{K_t}{\lambda P_{\check{d}}} \check{d}_k \dot{e}_k \tag{8D.3}$$

where the second equality is due to (8D.2). In steady state $\check{d}_k$ and $\dot{e}_k$ will be approximately uncorrelated. Then ν_k is white and has variance $\sigma_\nu^2 = K_t^2 \varepsilon / [\lambda^2 P_{\check{d}}]$. Coefficient fluctuations v_k at the integrator output are caused by ν_k and are fed back with a loop gain K_t. As a result, v_k has variance

$$\sigma_v^2 = \frac{B_l}{K_t^2} \sigma_\nu^2 = \frac{B_l}{\lambda^2 P_{\check{d}}} \varepsilon \tag{8D.4}$$

where $B_l = K_t/(2 - K_t) \simeq K_t$ denotes the noise bandwidth of the loop. Gradient noise is caused by coefficient fluctuations. To see how, we write the decision variable $\tilde{d}_k$ as $\tilde{d}_k = \lambda(d_k + n_k)(c^* + v_k)$ where $c^* = 1/\lambda$ denotes the expected steady-state AGC gain. Clearly

$$\tilde{d}_k \simeq d_k + n_k + \lambda d_k v_k, \tag{8D.5}$$

where we have neglected the noise×noise term $n_k v_k$. The last term in (8D.5) is gradient noise of variance

$$\sigma_{\text{gr}}^2 = \lambda^2 P_d \sigma_v^2 = B_l \frac{P_d}{P_{\check{d}}} \varepsilon. \tag{8D.6}$$

As P is decreased below M_c, adaptation is no longer based on all available data power, but only on a fraction $\alpha = P_{\check{d}}/P_d$ of it. It this loss factor that shows up here.

Exercise 8.19 Assume that d_k is a Class IV partial-response sequence $d_k = a_k - a_{k-2}$ and that $\check{d}_k = a_k$, i.e. that gain control is only based on the current data symbol ($P = 1$). How large is the loss factor α? How does the excess mean-square error compare with that of an optimum AGC (for which $\alpha = 1$) if both loops have a time constant τ of 50 symbol intervals?

APPENDIX 8E

Properties of the MMSE adaptation loop

This appendix largely follows the line of thought laid down by Widrow [88]. Elementary knowledge of matrix algebra (eigenvalues, eigenvectors, orthonormal modal decomposition, traces) is assumed.

Rate of convergence We first consider the mean coefficient trajectories, whose evolution is governed by the model of Fig. 8.38. Coupling depends on the overall loop matrix $\mathbf{MR}_y$. For simplicity we assume that $\mathbf{M}$ is symmetric and positive definite. The matrices $\mathbf{M} = K_t \mathbf{R}_y^{-1}$ and $\mathbf{M} = \mu \mathbf{I}$ that were discussed in Section 8.6 both fall into this category. As a result of this assumption, $\mathbf{MR}_y$ becomes symmetric and positive definite. For this reason it can be decomposed as

$$\mathbf{MR}_y = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{-1} \tag{8E.1}$$

where $\mathbf{Q}$ is the orthonormal modal matrix of $\mathbf{MR}_y$ and $\mathbf{\Lambda}$ is its diagonal matrix of eigenvalues: $\mathbf{\Lambda} = \text{diag}[\lambda_0, \ldots, \lambda_{N-1}]$. The trace of this matrix coincides with that of $\mathbf{MR}$. Since eigenvalues are all nonnegative, we can upper bound the largest eigenvalue $\lambda_{\max}$ by

$$\lambda_{\max} < \text{tr}[\mathbf{MR}_y]. \tag{8E.2}$$

Since $\mathbf{Q}$ is orthonormal, $\mathbf{Q}^T = \mathbf{Q}^{-1}$. We now apply a coordinate transformation to $\underline{c}$, $\hat{\underline{c}}$ and $\underline{v}$. Specifically, we define a transformed coefficient vector $\underline{c}' \triangleq \mathbf{Q}^{-1}\underline{c}$, and we define $\hat{\underline{c}}'$ and $\underline{v}'$ analogously. By linearity of the system of Fig. 8.38, we can surround the integrator by a matrix $\mathbf{Q}$ and its inverse $\mathbf{Q}^{-1}$ without affecting loop properties. We are also free to include a cascade $\mathbf{Q}^{-1}\mathbf{Q}$ in the reference terminal. This leads to Fig. 8E.1.

The matrix $\mathbf{Q}$ appears in both inputs of the summator and can be shifted across it. By (8E.1), the overall loop matrix $\mathbf{Q}^{-1}\mathbf{MR}_y\mathbf{Q}$ is a diagonal matrix $\mathbf{\Lambda}$. We arrive at the decoupled model of Fig. 8E.2, which is entirely cast in terms of transformed vectors.

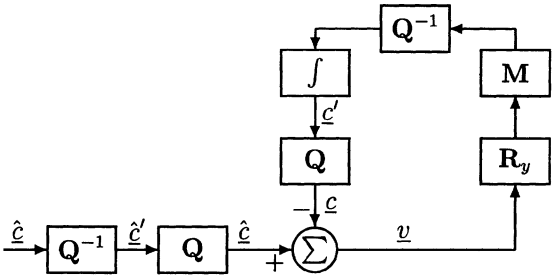


Figure 8E.1: Loop model with matrices $\mathbf{Q}$ and $\mathbf{Q}^{-1}$. This model is equivalent to that of Fig. 8.38.

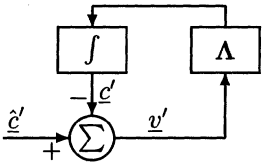


Figure 8E.2: Simplified version of Fig. 8E.1, equivalent to the loop of Fig. 8.38.

Since Λ is diagonal, there is no cross coupling between the transformed coefficients. The N uncoupled subloops have gains $\lambda_0, \ldots, \lambda_{N-1}$ that are, in general, unequal. As a result, some of the transformed coefficients converge more rapidly than other ones. The entire loop will be stable when all subloops are stable. The most critical one has loop gain $\lambda_{\max}$ and is stable in the discrete-time case when $\lambda_{\max} < 2$. By (8E.2), a sufficient condition for stability is $\mathrm{tr}[\mathbf{M}\mathbf{R}_y] < 2$. Given $\mathbf{M}$, this yields a condition on the powers of $y_k^0, \ldots, y_k^{N-1}$. The average loop gain is

$$\overline{K_t} = \frac{1}{N}\sum_{i=0}^{N-1} \lambda_i = \frac{1}{N}\mathrm{tr}[\Lambda] = \frac{1}{N}\mathrm{tr}[\mathbf{M}\mathbf{R}_y] \tag{8E.3}$$

and characterizes the average rate at which coefficients decay. The smallest eigenvalue $\lambda_{\min}$ defines the slowest mode of coefficient convergence. It was found by Ungerboeck [79] that the MSE decreases at a rate that is, in general, better characterized by the average eigenvalue than by $\lambda_{\min}$ (see Section 8.6.3 for a heuristic explanation). This justifies the definition of a time constant $\tau \triangleq 1/\overline{K_t}$ of the entire loop.

Steady-state EMSE Gradient noise arises in the cross-product $\underline{z} \triangleq e_k \underline{y}_k$. The average of this cross-product is the true gradient that the loop forces to zero. In steady state it has managed to do so, and $\underline{z}$ consists purely of gradient noise with autocovariance

$$E[\underline{z}\underline{z}^T] = E[\underline{y}\underline{y}^T e_k^2] = E[\underline{y}\underline{y}^T]E[e_k^2] \simeq \mathbf{R}_y \varepsilon_{\min}. \tag{8E.4}$$

The second equality is due to the fact that the loop suppresses any correlation between e and $\underline{y}$; the approximation applies because $\varepsilon \simeq \varepsilon_{\min}$ in steady state whenever K_t is not

too large. Gradient noise adds into the loop just before the weighing matrix $\mathbf{M}$. For simplicity we restrict attention to LMS, which has adaptation matrix $\mathbf{M} = \mu\mathbf{I}$. We then obtain Fig. 8E.3.a (compare with Fig. 8.38, which describes only the average trajectory of $\underline{c}$). By assumption, $\underline{z}$ is uncorrelated over time. Although this assumption may not always hold, it does lead to theoretical results that agree well with practice [88].

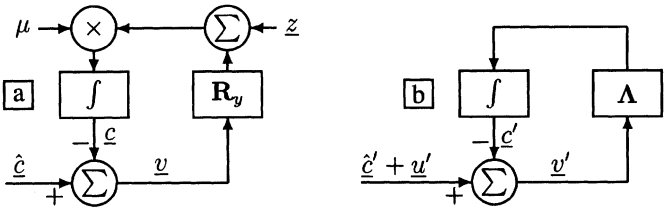


Figure 8E.3: Loop model with gradient noise $\underline{z}$ and degenerate adaptation matrix $\mathbf{M} = \mu\mathbf{I}$ (a), and equivalent model in primed coordinates (b).

We now refer noise back to the reference terminal (this involves premultiplication by $\mathbf{R}_y^{-1}$) and subsequently adopt primed coordinates as in Fig. 8E.2. This leads to Fig. 8E.3.b. The input contains noise $\underline{u}' \triangleq \mathbf{Q}^{-1}\mathbf{R}_y^{-1}\underline{z}$ that is uncorrelated over time and has autocovariance

$$E[\underline{u}'\underline{u}'^T] = \mathbf{Q}^{-1}\mathbf{R}_y^{-1}E[\underline{z}\underline{z}^T]\mathbf{R}_y^{-1}\mathbf{Q} = \mathbf{Q}^{-1}\mathbf{R}_y^{-1}\mathbf{Q}\varepsilon_{\min} = \mu\mathbf{\Lambda}^{-1}\varepsilon_{\min}. \tag{8E.5}$$

The last step follows from (8E.1) and the fact that $\mathbf{MR} = \mu\mathbf{R}_y$. Due to the coordinate transformation, the loop is now uncoupled. We can distinguish N loop gains $\lambda_0, \ldots, \lambda_{N-1}$ with associated noise bandwidth $B_{l,i} \simeq \lambda_i/2$. The ith component of $\underline{u}'$ is filtered by the ith loop; the corresponding component of $\underline{v}'$ has has variance $\sigma_{v_i}^2 = B_{l,i}\sigma_{u_i}^2 = B_{l,i}\mu\lambda_i^{-1}\varepsilon_{\min} \simeq \mu\varepsilon_{\min}/2$. This shows that all transformed coefficient deviations $\underline{v}_i'$ have the same variance.[8] Knowledge of these variances suffices to compute the excess MSE. By (8P.4), a deterministic coefficient deviation $\underline{v}$ increases $\varepsilon_{\min}$ by an amount $\underline{v}^T\mathbf{R}_y\underline{v}$. In our case $\underline{v}$ is random, and $\varepsilon_{\min}$ is increased by the gradient noise power

$$\sigma_{\text{gr}}^2 = E[\underline{v}^T\mathbf{R}_y\underline{v}] = \frac{1}{\mu}E[\underline{v}'^T\mathbf{\Lambda}\underline{v}']. \tag{8E.6}$$

Upon exploiting the diagonal structure of $\mathbf{\Lambda}$ we conclude that

$$\sigma_{\text{gr}}^2 = \frac{1}{\mu}\sum_{i=0}^{N-1}\lambda_i\sigma_{v_i}^2 = \frac{1}{2}\sum_{i=0}^{N-1}\lambda_i = \frac{1}{2}\text{tr}[\mathbf{\Lambda}] = \frac{\mu}{2}\text{tr}[\mathbf{R}_y]. \tag{8E.7}$$

Just as in the uncoupled situation, we can attribute an efficiency $\gamma \triangleq \varepsilon_{\min}/(\tau\sigma_{\text{gr}}^2) = 2/N$ to the loop.

[8]More generally, it can be shown that $\underline{v}'$ has autocovariance matrix $E[\underline{v}'\underline{v}'^T] = (\mu/2)\varepsilon_{\min}\mathbf{I}$, i.e. the components of $\underline{v}'$ have no cross-correlation and equal variance.

PROBLEMS

8.1 Consider a system in which the equalizer and detector are both fully adaptive. Assume that adaptation is based on the error sequence e_k of Figs. 8.1 and 8.2. Argue that the equalizer and detector will always converge to the trivial solution $c = q = e_k = 0$.

8.2 Consider the system of Fig. 8.3. Show that the power of the normalized error sequence $e'_k \triangleq e_k/\tilde{q}_0$ is a measure of the predetection signal-to-noise ratio provided that decision errors are absent and $\tilde{q}_0 = q_0$. What happens if the threshold $\tilde{q}_0$ is too large or too small?

8.3 Assume that the detector in Fig. 8.2 is a Viterbi detector, and that possible system responses $\tilde{q}$ are such that single bit-errors prevail. How should the error sequence e_k be normalized in order to become a reliable performance indicator? What happens if for some reason multiple bit errors come to prevail?

8.4 The technique that was mentioned in Example 8.3 for extracting an error sequence e_k from within the VD has at least two potential problems:

1. The VD is good at making deferred decisions; it is not meant, however, to produce reliable decisions about the current data symbol. A similar remark applies to reliable selection of a *current* error sample.
2. At instants where a *wrong* error sample is selected, erroneous update takes place of the impulse response estimate $\tilde{q}$ that is used for the branch metric computations of *all* survivors (i.e. not just the one whose error sample has been erroneously selected). This gives rise to a similar degradation as in the VD with global decision feedback (see Section 7.7.1).

Formulate solutions to these two problems and argue how well they will work.

8.5 Determine the steady-state gain $\hat{c}$ of the open-loop AGC of Fig. 8.8 when operating on the channel of Fig. 8.13. Assume that data d is uncorrelated with the noise n and denote the power of d and n by P_d and P_n, respectively. What is the *signal suppression factor* $\beta \triangleq \lambda\hat{c}$? How large is the *bias* $1 - \beta$ when P_d is 20 dB larger than P_n? Can this bias degrade transmission quality significantly?

8.6 Let r_k be a stationary and uncorrelated sequence with zero mean, power $P_r \triangleq E[r_k^2]$ and *kurtosis* $\kappa \triangleq E[r_k^4]/P_r^2$. An estimate $\tilde{P}_r$ of P_r is formed by taking an average across N symbol intervals according to

$$\tilde{P}_r = \frac{1}{N}\sum_{k=0}^{N-1} r_k^2. \tag{8P.1}$$

1. Show that $\kappa = 1$ when r_k is binary, $\kappa = 2$ when r_k is a ternary partial-response sequence, and $\kappa = 3$ when r_k is Gaussian.
2. Show that $\tilde{P}_r$ is an unbiased estimate of P_r, i.e. that $E[\tilde{P}_r] = P_r$.

3. Show that the error $\tilde{P}_r - P_r$ has variance

$$E[(\tilde{P}_r - P_r)^2] = \frac{\kappa - 1}{N} P_r^2. \tag{8P.2}$$

4. Suppose that we wish to estimate $\tilde{P}_r$ with an error variance of 0.1 dB. How many samples N are needed when r_k is binary? The same question for ternary and Gaussian r_k.

8.7 Consider the open-loop AGC of Fig. 8.8 when operating on the channel of Fig. 8.13. Assume that the LPF forms a moving average across N symbol intervals as in Problem 8.6. Since the AGC settles fully in N symbol intervals, it has an effective time constant τ that is significantly smaller, say $\tau \simeq N/2$. Let $v = c - \hat{c}$ be the deviation of the actual AGC gain from its nominal value.

1. Use the results of Problem 8.6 to show that v has variance $\sigma_v^2 \simeq [(\kappa - 1)/(4N)].[P_d/P_r]$ provided that N is sufficiently large.
2. Due to fluctuations v, a *gradient noise* component vr arises in x that comes on top of the scaled channel noise $g\hat{c}n$. Compute the power σ_{gr}^2 under the assumption that v is statistically independent of r. Is this assumption reasonable?
3. Compute the misadjustment $M \triangleq \sigma_{\text{gr}}^2/P_n$, the excess mean-square error EMSE$\triangleq 1 + M$, and the efficiency factor $\gamma \triangleq 1/(M\tau)$.

8.8 Redo Problem 8.5 for the closed-loop non-data-aided AGC of Fig. 8.9. Is the result any different?

8.9 Consider the closed-loop non-data-aided AGC of Fig. 8.9 when operating on the channel of Fig. 8.13. Use the results of Appendix 8A to show that the AGC induces gradient noise at the detector input of variance $\sigma_{\text{gr}}^2 = P_r\sigma_v^2 \simeq K_t P_d(\kappa - 1)/8$. Make use of the fact that v_k and u_k are uncorrelated. How large are the misadjustment $M \triangleq \sigma_{\text{gr}}^2/P_n$, the excess mean-square error EMSE$\triangleq 1 + M$, and the efficiency factor $\gamma \triangleq K_t/M$?

8.10 Discuss the relative merits of envelope detection and square-law detection in the AGCs of Figs. 8.9 and 8.11. Pay attention in particular to signal bandwidths, required dynamic ranges, gain bias, and sensitivity to DC-offsets.

8.11 Consider the ZF DA AGC of Fig. 8.14. Assume that the linear VCA is replaced by a quadratic one such that the integrator output x determines AGC gain c according to $c = x^2$. Show that this leads to an effective loop gain $K_t = 2\mu P_d\sqrt{\lambda}$ that depends less heavily on the channel gain λ than for an AGC with linear VCA. What is the effect of a higher-order nonlinearity?

8.12 Assume that the linear VCA in Fig. 8.14 is replaced by an exponential one. Use a development akin to that of Section 8.3.2 to show that loop properties are now independent of the channel gain λ. Repeat Exercise 8.3 for this situation.

8.13 Prove (8.15).

8.14 Assume that the linear VCA in the MMSE DA AGC of Fig. 8.17 is replaced by an exponential one with gain-control characteristic $c(v) = \exp\{c\}$. Show that the loop has gain $K_t \simeq \mu\lambda P_d$ at high signal-to-noise ratios.

8.15 Consider the MMSE DA AGC of Fig. 8.17. Assume that we replace the linear voltage-controlled amplifier by a nonlinear one with characteristic $\tilde{d} = \alpha r/c$, where α is a constant.[9] Assume that noise is negligible and that c is already close to its ideal value $\hat{c}$ (hence the analysis is restricted to the *tracking* mode of AGC operation). Use a linearization as in Appendix 8A to show that the adaptation time constant τ is independent of the gain λ of the channel.

8.16 Prove (8.18).

8.17 Consider an L-level full-response data transmission system with detector input $\tilde{a}_k = \beta a_k + n_k$ where $a_k \in \{-(L-1), \ldots, -1, 1, \ldots, (L-1)\}$ is the transmitted data and n_k is additive noise of variance σ^2. The system gain β is determined by the applied AGC, and may deviate from the ideal value 1 as a result of gain bias. In this problem we are concerned with the effect of this bias on performance for the situation that a symbol-by-symbol detector with equally spaced decision thresholds is used. These thresholds are located at $0, \pm 2, \ldots, \pm(L-2)$.

1. Express the predetection SNR in terms of L and σ^2 for the situation that there is no gain bias ($\beta = 1$).
2. Assume that β becomes slightly smaller than unity. Which data levels are most likely to be detected erroneously? What is the corresponding predetection SNR? What is the SNR loss $\mathcal{L}$ with respect to $\beta = 1$?
3. Assume that the gain bias amounts to 1%, i.e. that $\beta = 0.99$. Quantify $\mathcal{L}$ for $L = 2, 4$ and 8.

8.18 In this problem we compare various AGCs for a ternary partial-response system with bit-by-bit detector according to Fig. 8.13. The partial-response sequence d_k is in the alphabet $\{-2, 0, 2\}$ and has power $P_d = 2$; noise n_k is Gaussian has power P_n.

1. Show that the kurtosis κ of the received sequence r_k must satisfy $\kappa \in [2, 3]$. Hint: First determine the kurtosis of the data and noise components of r_k.
2. An ideal AGC would normalize the system gain λc to 1 and induce no gradient noise. In that case the predetection signal-to-noise ratio would amount to $\rho = 1/P_n$. A practical AGC will induce an overall gain $\beta \triangleq \lambda c$ that may be smaller than 1, and will induce gradient noise that causes an excess mean-square error EMSE. Show that the bias and EMSE together lead to a predetection signal-to-noise ratio

$$\tilde{\rho} \simeq \frac{(2-\beta)^2}{\text{EMSE}} \frac{1}{P_n}. \tag{8P.3}$$

   Why is the approximation not an equality? Does $\mathcal{L}$ change when the detector is replaced by a Viterbi detector? If so, how?
3. The factor $\mathcal{L} \triangleq \rho/\tilde{\rho} = (2-\beta)^2/\text{EMSE}$ represents the loss in predetection signal-to-noise ratio induced by the AGC. Compute $\mathcal{L}$ as a function of the signal-to-noise ratio P_d/P_n for the AGCs of Table 8.1 when they are all dimensioned to have a time constant τ of 25 symbol intervals. How do the various AGCs rate at practical signal-to-noise ratios? Would you recommend a different time constant?
   Hint: Make use of the fact that EMSE$= 1 + M = 1 + 1/(\tau\gamma)$, where γ is the efficiency of the algorithm as in Table 8.1.

[9] This characteristic was first identified by Kevin D. Fisher.

8.19 Assume that we dimension the AGCs of Example 8.13 for an EMSE of 0.1 dB. How do the learning curves of Fig. 8.21 change? Does the relative performance of NDA versus DA AGCs improve? Answer the same questions for a decrement of the signal-to-noise ratio P_d/P_n to 10 dB.

8.20 Consider the discrete-time version of the ZF DA AGC of Fig. 8.14. Let d_k be a ternary partial-response sequence with alphabet $\{-2, 0, 2\}$, so that $P_d = 2$. Assume that a DC-offset of 0.2 is added to η. Determine the resulting bias in the steady-state system gain λc. Does this bias depend on λ?

8.21 Redo the previous problem for the MMSE DA AGC of Fig. 8.17. Assume for simplicity that noise is absent, and that λ can vary between 0.1 and 1, i.e. a range of 20 dB. Does the gain bias depend on λ? If so, what is the worst-case bias? Can this be tolerated?

8.22 This problem is concerned with the effect of DC offsets on the data-aided ZF and MMSEs AGCs when equipped with an exponential VCA. Repeat the analysis of Section 8.3.7 for this VCA type. Show that a large loop gain remains desirable, and that for both AGCs the qualitative dependence of bias on λ does not change relative to a linear VCA. Hint: Make use of the results of Problems 8.12 and 8.14.

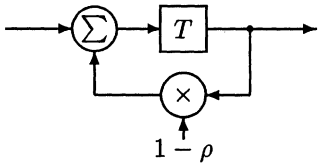


Figure 8P.1: Discrete-time leaky integrator.

8.23 Fig. 8P.1 shows a discrete-time leaky integrator with leakage factor ρ. Show that this integrator has normalized cut-off frequency $\Omega_c \simeq \rho/(2\pi)$ whenever $\Omega_c \ll 1$, and that the continuous-time results on integrator leakage of Section 8.3.7 all remain valid in this limiting discrete-time case.

8.24 Assume that a $1 + \mathcal{D}$ precoder is added at the transmitting end of the system of Example 8.14, and that an inverse precoder is included in the receiver. Show that the limit cycle for AGC startup can be detected at the output of the inverse precoder in a very simple manner. Generalize this technique to other partial responses.

8.25 Consider the closed-loop non-data-aided adaptation scheme of Fig. 8.25. Assume that the desired equalizer output d is Class IV and that the prefilter suppresses any content above the Nyquist frequency.

1. Determine the power spectral density of the desired equalizer output d.
2. Determine the total power of d and of its first derivative.
3. Assume that the LPF in the c_0-loop is absent and that the HPF in the c_1-loop is a first-order differentiator. Determine the desired powers $P_{0,d}$ and $P_{2,d}$.
4. A potential problem of the equalizer structure of Fig. 8.25 is that it cannot accurately compensate for slope distortion at very high densities. Specifically, the transfer magnitude of the equalizer can grow no faster than f^2 with frequency f, whereas the mag-

netic channel exhibits exponential roll-off. Argue that a slight change of $P_{2,d}$ with respect to the nominal value that was just computed may be desirable when very high densities are to be handled. What is the consequence at less extreme densities?

5. How do loop properties depend on the amplitude of r? Suggest a manner to eliminate this dependence.

8.26 Redo Exercise 8.7 for a fractionally-spaced equalizer with tap spacing LT/M, where L and M are positive integers such that $L < M$. Simplify your results for the situation $L = 1$. (The solution to this problem is closely related to the approach of [75].)

8.27 How do (8.32), (8.27), (8.33) and (8.34) change when data is correlated?

8.28 In this problem we are concerned with the uniqueness of the central tap position in the T-spaced ZF equalizer of Fig. 8.31. To keep the analysis simple we consider a received sequence r_k without ISI. In that case the eye at the output of the equalizer will be open whenever $c_n = 1$ for some $n \in \{0, \ldots, N-1\}$ and all other taps c_j ($j \neq n$) are zero. We shall refer to c_n as the *central tap* value and to n as the *central tap position*. In order for the equalizer output $\tilde{a}_k$ to be an *undelayed* estimate of a_k, we take r_k to be of the form $r_k = a_{k+n} + u_k$, where u_k is additive noise. (The results of the analysis remain valid for other delays of r with respect to a, but formulas become more complicated.) Decisions $\hat{a}_k$ are assumed to be correct. This requires that noise be sufficiently small and that equalizer settings be sufficiently close to the solution just mentioned.

1. Express the delayed error sequence e_{k-N_l} in terms of a, c and u.
2. What adaptation matrix $\mathbf{M}$ yields full coefficient decoupling for the channel at hand? Determine the expectation $E[e_{k-N_l} a_{k-j}]$ that governs the update of the jth coefficient c_j. Assume for simplicity that data a_k is uncorrelated and has unit power.
3. Show that there is only a single central tap position $n \in \{0, \ldots, N-1\}$ for which $E[e_{k-N_l} a_{k-j}]$ depends on c_j. Argue that this position leads to coefficient convergence, while all other central tap positions lead to a random coefficient drift.

8.29 Assume that the weighing matrix $\mathbf{M}$ in Fig. 8.31 is replaced by a scaled identity matrix $\mathbf{M} = \mu\mathbf{I}$. For what channel does the loop become orthogonalized? Would coupling be strong for channels with heavy ISI?

8.30 How does Fig. 8.31 change for an MMSI loop? How does this change affect complexity?

8.31 Consider the bipolar $(1 - \mathcal{D})$ system of Example 8.19 with a T-spaced transversal equalizer having N taps. Let S_P be the smallest possible span that permits MMSI adaptation, i.e. $S_P = \{0, \ldots, N\}$ (see Exercise 8.10).

1. Determine the $(N+1) \times N$ channel matrix $\mathbf{F}$ for this situation.
2. Determine the $N \times N$ autocorrelation matrix $\mathbf{F}\mathbf{F}^T$. Verify that this matrix has a symmetric inverse with components $(\mathbf{F}\mathbf{F}^T)^{-1}_{i,j} = (i+1)(N-j)/(N+1)$ for $j \geq i$. (Further properties of $\mathbf{F}\mathbf{F}^T$ (eigenvalues, eigenvectors) are derived in [10].)
3. Determine the $(N+1) \times N$ MMSI adaptation matrix $\mathbf{M} = K_t(\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F}$. Are all components of $\mathbf{M}$ nonzero?

8.32 Consider the set of all initial equalizer settings that yield an MSE of -8 dB in the system of Example 8.19. Are the corresponding orthogonalized MMSI learning curves very different from each other? Answer the same question for coupled PZF adaptation.

8.33 Consider the system of Example 8.20. Assume that gain variations are so large that a cascade of two identical VCAs is needed to normalize signal levels before the prefilter in Fig. 8.35. Redesign the adaptation matrix $\mathbf{M}'$ to account for this extended topology.

8.34 In Fig. 8.36, the data vector $\underline{a}_k = [a_{k-P_l}, \ldots, a_{k-P_u}]^T$ has length $P = P_u - P_l + 1$, and is multiplied by an $N \times P$ matrix $\mathbf{F}$ that describes the symbol responses of the system up to the equalizer taps. In this problem and following ones we will take the span $S_P = \{P_l, \ldots P_u\}$ sufficiently large to cover all emerging ISI. In that case the vector disturbance $\underline{u}_k$ describes the noise components of the tap signals and contains no ISI. Just as in Section 8.5, we take the desired sequence d_k to be a linearly transformed version of the data a_k according to $d_k = (a * g)_k$, where g_k is a prescribed target response. In vector notation we have $d_k = \underline{a}_k^T \underline{g}$, where $\underline{g} = [g_{P_l}, \ldots, g_{P_u}]^T$ is a vector of length P that contains all nonzero components of g_k (S_P must be large enough to ensure this).

1. Show that $\underline{p}$ and $\mathbf{R}_y$ may be expressed as $\underline{p} = \underline{g}_k^T \mathbf{R}_a \mathbf{F}^T$ and $\mathbf{R}_y = \mathbf{F}\mathbf{R}_a\mathbf{F}^T + \mathbf{R}_u$ where $\mathbf{R}_a = E[\underline{a}_k \underline{a}_k^T]$ is the $P \times P$ data autocorrelation matrix, and $\mathbf{R}_u \triangleq E[\underline{u}_k \underline{u}_k^T]$ is the $N \times N$ autocorrelation matrix of the noise component of the tap signals. How do these expressions simplify for uncorrelated data of unit power?
2. Reformulate the optimum tap vector $\hat{\underline{c}}$ and MMSE $\varepsilon_{\min}$ as given by (8.39) and (8.40) in terms of $\mathbf{F}$, $\mathbf{R}_u$ and $\underline{g}$.

8.35 Show that the gradient $\underline{\nabla}$ of (8.38) and the MSE ε of (8.37) may be expressed as $\underline{\nabla} = 2\mathbf{R}_y \underline{v}$ and

$$\varepsilon = \varepsilon_{\min} + \underline{v}^T \mathbf{R}_y \underline{v} \tag{8P.4}$$

where $\underline{v} \triangleq \underline{c} - \hat{\underline{c}}$ is the difference between the actual and optimum tap vectors.

8.36 Consider a T-spaced transversal equalizer with N taps as in Example 8.21. Let the equalizer input y_k be a sinusoid with unit power and angular frequency ω ($y_k = \sqrt{2}\sin\omega k$). In this problem we analyze the autocorrelation matrix $\mathbf{R}_y$ of the tap signal vector as a function of the equalizer length N. This matrix has components $\mathbf{R}_y(i,j) = E[y_{k-i}y_{k-j}]$. Since the equalizer input is deterministic rather than random, we must interpret the expectation as a time average.

1. Compute $E[y_{k-i}y_{k-j}]$. (Hint: Write $y_{k-i}y_{k-j}$ as a sum of cosines, and neglect all terms that DC-free.)
2. Determine $\mathbf{R}_y$ and $\det(\mathbf{R}_y)$ for an equalizer with $N = 2$ taps. When is $\mathbf{R}_y$ regular?
3. Determine $\mathbf{R}_y$ and $\det(\mathbf{R}_y)$ for $N = 3$. When is $\mathbf{R}_y$ regular?
4. Prove that $\det(\mathbf{R}_y) = 0$ irrespective of ω for all $N \geq 3$.

8.37 Consider the MMSE adaptation loop of Fig. 8.37. For what type of channel does the adaptation matrix $\mathbf{M} = \mu\mathbf{I}$ result in full coefficient decoupling? Does coupling become strong when ISI is heavy?

8.38 In Appendix 7C we have seen that performance of the Viterbi detector depends on weighted versions $e'_k = (e * w)_k$ of the error sequence e_k, where w_k is the impulse response of the weighing filter. For a $1 \pm \mathcal{D}^L$ VD, for example, this filter has impulse response $w_k = \delta_k \pm \delta_{k-L}$ (i.e. $e'_k = e_k \pm e_{k-L}$) whenever single bit-errors prevail. Modify the loop of Fig. 8.37 so that it minimizes the power of e' rather than that of e. Use the reasoning of Section 10.7.3 to simplify the resulting loop such that the only difference with that of Fig. 8.37 is

an additional filter in the e_k-branch. Show that this added filter must have impulse response $(w * w_-)_k$ where '$_-$' denotes time-reversal.

8.39 Consider a stationary vector sequence $\underline{x}$ with autocorrelation matrix $\mathbf{R}_x = E[\underline{x}\underline{x}^T]$. This sequence is applied to an $N \times N$ weighing network that is characterized by a matrix $\mathbf{Q}$. The output of the network is a vector sequence $\underline{y}$.

1. Determine the autocorrelation matrix $\mathbf{R}_y = E[\underline{y}\underline{y}^T]$.
2. How must $\mathbf{Q}$ be designed in order for the components of $\underline{y}$ to be orthonormal (i.e. for $\underline{y}$ to have autocorrelation matrix $\mathbf{R}_y = \mathbf{I}$)?
3. Explain how a weighing network of this type can be used to decouple adaptation with a standard MMSE adaptation loop (i.e. one with adaptation matrix $\mathbf{M} = \mu\mathbf{I}$). Would the same network also decouple ZF adaptation? This technique is due to Chang [10].

8.40 Follow the approach of Section 8.5 to design a partial zero-forcing (PZF) adaptation loop for the system of Fig. 8.43. Verify that this loop is just the one of Fig. 8.44.

8.41 Elaborate the loop of Fig. 8.44 for adaptation of the feedback detector of Fig. 8.4.

8.42 Express the loop gain K_t of the threshold adaptation loop according to Figs. 8.45 and 8.46 in terms of μ.

8.43 Express the loop gain K_t of the DC-compensation loop of Fig. 8.47 in terms of μ.

8.44 Digital recording systems can be hampered by severe linear ISI along with nonlinearities that are mainly restricted to interaction between the current bit and a few neighboring bits. The hybrid filter of Fig. 8P.2 matches this ISI structure in a rather natural way.

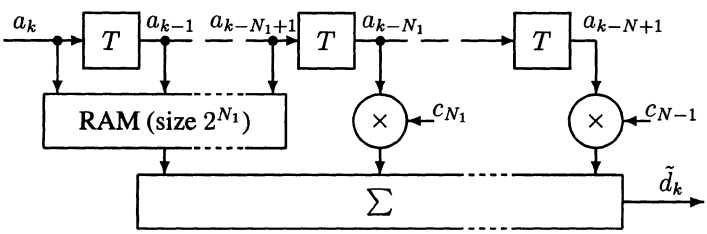


Figure 8P.2: Hybrid filter with table-look-up and transversal part.

The first section is a table look-up (TLU) filter that can handle any interaction between N_1 successive bits $a_k, \ldots, a_{k-N_1+1}$, and the remaining part is a transversal filter with $N_2 \triangleq N - N_1$ taps that serves to handle the remaining ISI, which is assumed linear. With this structure, a much smaller table is needed than for a TLU filter of memory length N, and adaptation properties are better. Even so, all principal ISI components can still be handled.

1. Assume that the filter is embedded in the configuration of Fig. 8.42. Outline how the table (RAM) and the coefficients of the transversal filter can be jointly adapted.
2. Assume that that the adaptation subloops of both filter parts are dimensioned to have equal adaptation time constant τ. Do they contribute the same amount of gradient noise to $\tilde{d}_k$ and e_k? If not, what is the difference?

3. Assume that both subloops are dimensioned in order for both filter parts to contribute equal amounts of gradient noise. How do their time constants relate?
4. Modify the filter of Fig. 8P.2 so that it becomes suitable as a feedback filter for the feedback detector of Fig. 8.4. What further changes are necessary to enable the filter to perform adaptive DC-compensation?

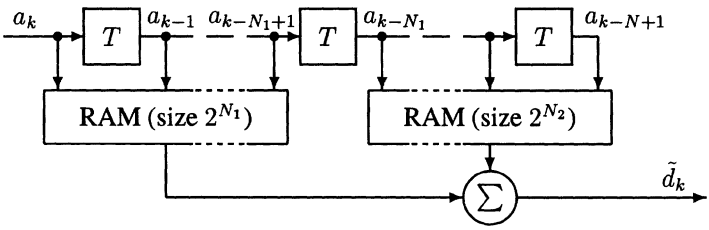


Figure 8P.3: Filter structure with two disjunct table look-up filters.

8.45 Fig. 8P.3 depicts a filter structure that consists of two table look-up (TLU) filters which operate on disjunct portions of the data vector $\underline{a}_k$. The TLU filter outputs are summed to obtain the total filter output $\tilde{d}$.

1. Can this filter handle arbitrary nonlinear functions of the data vector $\underline{a} = [a_k, \ldots, a_{k-N_1+1}]^T$? If not, then what functions can be handled?
2. Compare the required table size of this filter to that of a single TLU filter of memory length N.
3. Assume that the filter of Fig. 8P.3 is embedded in the configuration of Fig. 8.42. Outline how both TLU filters can be joinly adapted.
4. Assume that both tables are equally large and that their adaptation sub-loops are dimensioned to have identical time constants. Determine the efficiency factor γ for the complete adaptive filter and compare it to that of a single adaptive TLU filter of memory length N.
5. Can the structure of Fig. 8P.3 compensate for DC-offsets? Are there potential problems in this regard?

8.46 Repeat the derivation of Appendix 8A for a continuous-time AGC. Are the stability conditions more or less strict than for a discrete-time AGC?

REFERENCES

[1] O. Agazzi, D.G. Messerschmitt and D.A. Hodges, 'Nonlinear Echo Cancellation of Data Signals', *IEEE Trans. Commun.*, Vol. COM-30, pp. 2421-2433, Nov. 1982.

[2] S.T. Alexander, *Adaptive Signal Processing, Theory and Applications*. Vienna: Springer Verlag, 1986.

[3] C.T. Beare, 'The Choice of the Desired Impulse Response in Combined Linear-Viterbi Algorithm Equalizers', *IEEE Trans. Commun.*, Vol. COM-26, No. 8, pp. 1301-1307, Aug. 1978.

[4] M.G. Bellanger, *Adaptive Digital Filters and Signal Analysis*. New York and Basel: Marcel Dekker, Inc., 1987.

[5] J.W.M. Bergmans, S.A. Rajput and F.A.M. van de Laar, 'On the Use of Decision Feedback for Simplifying the Viterbi Detector', *Philips J. Res.*, Vol. 42, No. 4, pp. 399-428, 1987.

[6] J.W.M. Bergmans, S. Mita and M. Izumita, 'Characterization of Digital Recording Channels', *Philips J. Res.*, Vol. 44, No. 1, pp. 57-96, 1989.

[7] N.J. Bershad, 'Analysis of the Normalized LMS Algorithm with Gaussian Inputs', *IEEE Trans. Acoust., Speech, Signal Process.*, Vol. ASSP-34, pp. 793-806, Aug. 1986.

[8] C. Caraiscos and B. Liu, 'A Roundoff Error Analysis of the LMS Adaptive Algorithm', *IEEE Trans. Acoust., Speech, Signal Process.*, Vol. ASSP-32, pp. 34-41, Feb. 1984.

[9] J.K. Chamberlain, F.M. Clayton, H. Sari and P. Vandamme, 'Receiver Techniques for Microwave Digital Radio', *IEEE Communications Magazine*, Vol. 24, No. 11, pp. 43-54, Nov. 1986.

[10] R.W. Chang, 'A New Equalizer Structure for Fast Start-Up Digital Communication', *Bell Syst. Tech. J.*, Vol. 50, pp. 1969-2014, July-Aug. 1971.

[11] G. Cherubini, 'Nonlinear Self-Training Adaptive Equalization for Partial-Response Systems', *IEEE Trans. Commun.*, Vol. COM-42, No. 2/3/4, pp. 367-376, Feb./March/Apr. 1994.

[12] 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. Commun.*, Vol. 44, No. 6, pp. 675-685, June 1996.

[13] R.D. Cideciyan, F. Dolivo, R. Hermann, W. Hirt and W. Schott, 'A PRML System for Digital Magnetic Recording', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 38-56, Jan. 1992.

[14] J.M. Cioffi, 'Limited-Precision Effects in Adaptive Filtering', *IEEE Trans. Circuits Syst.*, Vol. CAS-34, No. 7, pp. 821-833, July 1987.

[15] T.A.C.M. Claasen and W.F.G. Mecklenbräuker, 'Comparison of the Convergence of Two Algorithms for Adaptive FIR Digital Filters', *IEEE Trans. Circuits Syst.*, Vol. CAS-28, No. 6, pp. 510-518, June 1981.

[16] T.A.C.M. Claasen and W.F.G. Mecklenbräuker, 'Adaptive Techniques for Signal Processing in Communications', *IEEE Communications Magazine*, Vol. 23, No. 11, pp. 8-19, Nov. 1985.

[17] G. Clark, S.K. Mitra and S.R. Parker, 'A Unified Approach to Time- and Frequency-Domain Realization of FIR Adaptive Digital Filters', *IEEE Trans. Acoust., Speech, Signal Proces.*, Vol. ASSP-31, No. 5, pp. 1073-1083, Oct. 1983.

[18] C.F.N. Cowan, S.G. Smith and J.H. Elliott, 'A Digital Adaptive Filter Using a Memory-Accumulator Architecture: Theory and Realization', *IEEE Trans. Acoust., Speech, Signal Process.*, Vol. ASSP-31, No. 3, pp. 541-549, June 1983.

[19] C.F.N. Cowan and P.M. Grant, *Adaptive Filters*. Englewood Cliffs, New Jersey: Prentice-Hall, 1985.

[20] D.L. Duttweiler, 'Adaptive Filter Performance with Nonlinearities in the Correlation Multiplier', *IEEE Trans. Acoust., Speech, Signal Proces.*, Vol. ASSP-30, pp. 578-586, Aug. 1982.

[21] A.W.M. van den Enden and N.A.M. Verhoeckx, *Discrete-Time Signal Processing - An Introduction.* Hemel Hempstead, Hertfordshire (UK): Prentice-Hall, 1989.

[22] D.D. Falconer and F.R. Magee, Jr., 'Adaptive Channel Memory Truncation for Maximum-Likelihood Sequence Estimation', *Bell Syst. Tech. J.*, Vol. 52, pp. 1541-1562, Nov. 1973.

[23] D.D. Falconer and L. Ljung, 'Application of Fast Kalman Estimation to Adaptive Equalization', *IEEE Trans. Commun.*, Vol. COM-26, pp. 1439-1446, Oct. 1978.

[24] S.A. Fredricksson, 'Joint Optimization of Transmitter and Receiver Filters in Digital PAM Systems with a Viterbi Detector', *IEEE Trans. Inform. Theory*, Vol. IT-18, No. 2, pp. 200-210, Mar. 1976.

[25] B. Friedlander, 'Lattice Filtes for Adaptive Processing', *Proc. IEEE*, Vol. 70, No. 8, pp. 829-867, Aug. 1982.

[26] D.A. George, R.R. Bowen and J.R. Storey, 'An Adaptive Decision Feedback Equalizer', *IEEE Trans. Commun. Technol.*, Vol. COM-19, pp. 281-293, June 1971.

[27] A. Gersho, 'Adaptive Equalization of Highly Dispersive Channels', *Bell Syst. Tech. J.*, Vol. 48, pp. 55-70, Jan. 1969.

[28] P.J. van Gerwen, N.A.M. Verhoeckx and T.A.C.M. Claasen, 'Design Considerations for a 144 kbit/s Digital Transmission Unit for the Local Telephone Network', *IEEE J. Select. Areas Commun.*, Vol. SAC-2, No. 2, pp. 314-323, Mar. 1984.

[29] R.D. Gitlin, J.E. Mazo and M.G. Taylor, 'On the Design of Gradient Algorithms for Digitally Implemented Adaptive Filters', *IEEE Trans. Circuit Theory*, Vol. CT-20, pp. 125-136, Mar. 1973.

[30] R.D. Gitlin and F.R. Magee, Jr., 'Self-Orthogonalizing Algorithms for Accelerated Convergence of Adaptive Equalizers', *IEEE Trans. Commun.*, Vol. COM-25, pp. 666-672, July 1977.

[31] R.D. Gitlin and S.B. Weinstein, 'On the Required Tap-Weight Precision for Digitally-Implemented Adaptive Mean-Squared Equalizers', *Bell Syst. Tech. J.*, Vol. 58, pp. 301-321, Feb. 1979.

[32] R.D. Gitlin, H.C. Meadors, Jr., and S.B. Weinstein, 'The Tap-Leakage Algorithm: An Algorithm for the Stable Operation of a Digitally Implemented, Fractionally-Spaced Adaptive Equalizer', *Bell Syst. Techn. J.*, Vol. 61, pp. 1817-1939, Oct. 1982.

[33] R.D. Gitlin and H.C. Meadors, Jr., 'Center-Tap Tracking Algorithms for Timing Recovery', *AT&T Techn. J.*, Vol. 66, pp. 63-78, Nov./Dec. 1987.

[34] D.N. Godard, 'Channel Equalization Using a Kalman Filter for Fast Data Transmission', *IBM J. Res. Develop.*, Vol. 18, pp. 267-273, May 1974.

[35] D.N. Godard, 'Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems', *IEEE Trans. Commun.*, Vol. COM-28, pp. 1867-1875, Nov. 1980.

[36] L.J. Griffiths and P.E. Mantey, 'Iterative Least-Squares Algorithm for Signal Extraction', in *Proc. Second Hawaii Int. Conf. System Sciences*, Western Periodicals Co., pp. 767-770, 1969.

[37] L. Guidoux, 'Egalisateur Autoadaptif a Double Echantillonage', *L'onde Electronique*, Vol. 55, pp. 9-13, Jan. 1975.

[38] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, New Jersey: Prentice-Hall, 1986.

[39] R.W. Harris, D.M. Chabries and F.A. Bishop, 'A Variable Step Adaptive Filter Algorithm', *IEEE Trans. Acoust. Speech, Signal Proces.*, Vol. ASSP-34, No. 2, pp. 309-316, Apr. 1986.

[40] N. Holte and S. Stueflotten, 'A New Digital Echo Canceller for Two-Wire Subscriber Lines', *IEEE Trans. Commun.*, Vol. COM-29, No. 11, pp. 1573-1581, 1981.

[41] M.L. Honig and D.G. Messerschmitt, *Adaptive Filters: Structures, Algorithms, and Applications*. Boston: Kluwer Academic Publishers, 1984.

[42] C.R. Johnson, 'Adaptive IIR Filtering: Current Results and Open Issues', *IEEE Trans. Inform. Theory*, Vol. IT-30, No. 2, pp. 237-250, March 1984.

[43] P. Kabal, 'The Stability of Adaptive Minimum Mean Square Error Equalizers Using Delayed Adjustment', *IEEE Trans. Commun.*, Vol. COM-31, pp. 430-432, Mar. 1983.

[44] T. Kameyama, S. Takanami and R. Arai, 'Improvement of Recording Density by Means of Cosine Equalizer', *IEEE Trans. Magn.*, Vol. MAG-12, No. 6, pp. 746-748, Nov. 1976.

[45] H. Kobayashi, 'Application of Probabilistic Decoding to Digital Magnetic Recording Systems', *IBM J. Res. Develop.*, pp. 64-74, Jan. 1971.

[46] C.P. Kwong, 'Dual Sign Algorithm for Adaptive Filtering', *IEEE Trans. Commun.*, Vol. COM-34, No. 12, pp. 272-1275, Dec. 1986.

[47] R.H. Kwong and E.W. Johnston, 'A Variable Step Size LMS Algorithm', *IEEE Trans. on Signal Processing*, Vol. 40, No. 7, pp. 1633-1642, July 1992.

[48] R.W. Lucky, 'Automatic Equalization for Digital Communications', *Bell Syst. Tech. J.*, Vol. 44, pp. 547-588, April 1965.

[49] R.W. Lucky, J. Salz and E.J.Weldon, Jr., *Principles of Data Communication*. New York: McGraw-Hill, 1968.

[50] O. Macchi and E. Eweda, 'Convergence Characteristics of Self-Adaptive Equalizers', *IEEE Trans. Inform. Theory*, Vol. IT-30, pp. 161-176, Mar. 1984.

[51] V. J. Mathews, 'Performance Analysis of Adaptive Filters Equipped with the Dual Sign Algorithm', *IEEE Trans. on Signal Processing*, Vol. 39, No. 1, pp. 85-91, Jan. 1991.

[52] J.E. Mazo, 'On the Independence Theory of Equalizer Convergence', *Bell Syst. Tech. J.*, Vol. 58, No. 5, pp. 963-993, May-June 1979.

[53] J.E. Mazo, 'Analysis of Decision-Directed Equalizer Convergence', *Bell Syst. Tech. J.*, Vol. 59, No. 10, pp. 1857-1876, Dec. 1980.

[54] J.G. McWhirter, 'RLS Minimization Using a Systolic Array', *Proc. SPIE*, Vol. 43, pp. 415-431, 1983.

[55] D.V. Mercy, 'A Review of Automatic Gain Control Theory', *The Radio and Electronic Engineer*, Vol. 51, No. 11/12, pp. 579-590, Nov/Dec. 1981.

[56] D.G. Messerschmitt, 'Design of a Finite Impulse Response for the Viterbi Algorithm and Decision Feedback Equalizer', in *Proc. IEEE Int. Conf. Communications, ICC-74* (Minneapolis, MN, June 17-19, 1974).

[57] H. Meyr and G. Ascheid, *Synchronization in Digital Communications - Volume I*. New York: Wiley, 1990.

[58] A. Milewski, 'Periodic Sequences with Optimal Properties for Channel Estimation and Fast Start-Up Equalization', *IBM J. Res. Develop.*, Vol. 27, pp. 426-431, Sept. 1983.

[59] S. Mita, H. Ide and Y. Inagaki, 'A Simplified Automatic Equalizer for Digital VTR', *Journal of the Institute of Television Engineers of Japan*, Vol. 44, No. 6, pp. 728-735, June 1990 (in Japanese).

[60] K.H. Mueller, 'A New, Fast-Converging Mean-Square Algorithm for Adaptive Equalizers with Partial-Response Signaling', *Bell Syst. Tech. J.*, Vol. 54, No. 1, pp. 143-153, Jan. 1975.

[61] K.H. Mueller and D.A. Spaulding, 'Cyclic Equalization - A New Rapidly Converging Equalization Technique for Synchronous Data Communication', *Bell Syst. Tech. J.*, Vol. 54, pp. 369-406, Feb. 1975.

[62] J.I. Naguno and A. Noda, 'A Learning Method for System Identification', *IEEE Trans. Autom. Contr.*, Vol. AC-12, pp. 282-287, June 1967.

[63] J.E. Ohlson, 'Exact Dynamics of Automatic Gain Control', *IEEE Trans. Commun.*, Vol. COM-22, No. 1, pp. 72-75, Jan. 1974.

[64] B.M. Oliver, 'Automatic Volume Control as a Feedback Problem', *Proc. IRE*, Vol. 36, pp. 466-473, Apr. 1948.

[65] J.G. Proakis and J.H. Miller, 'An Adaptive Receiver for Digital Signaling through Channels with Intersymbol Interference', *IEEE Trans. Inform. Theory*, Vol. IT-15, pp. 484-497, July 1969.

[66] J.G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1983 (2nd ed. 1989).

[67] S.U.H. Qureshi and E.E. Newhall, 'An Adaptive Receiver for Data Transmission Over Time-Dispersive Channels', *IEEE Trans. Inform. Theory*, Vol. IT-19, pp. 448-457, July 1973.

[68] S.U.H. Qureshi, 'Adjustment of the Position of the Reference Tap of an Adaptive Equalizer', *IEEE Trans. Commun.*, Vol. COM-21, pp. 1046-1052, Sept. 1973.

[69] S.U.H. Qureshi, 'Timing Recovery for Equalized Partial-Response Systems', *IEEE Trans. Commun.*, Vol. COM-24, pp. 1326-1331, Dec. 1976.

[70] S.U.H. Qureshi, 'Fast Start-Up Equalization With Periodic Training Sequences', *IEEE Trans. Inform. Theory*, Vol. IT-23, pp. 553-563, Sept. 1977.

[71] S.U.H. Qureshi, 'Adaptive Equalization', *Proc. IEEE*, Vol. 73, No. 9, pp. 1349-1387, Sep. 1985.

[72] Y. Sato, 'A Method of Self-Recovering Equalization for Multi-Level Amplitude Modulation', *IEEE Trans. Commun.*, Vol. COM-23, pp. 679-682, June 1975.

[73] E.H. Satorius and S.T. Alexander, 'Channel Equalization Using Adaptive Lattice Algorithms', *IEEE Trans. Commun.* (Concise Paper), Vol. COM-27, pp. 899-905, June 1979.

[74] R.C. Schneider, 'An Improved Pulse-Slimming Method for Magnetic Recording', *IEEE Trans. Magn.*, Vol. MAG-11, No. 5, pp. 1240-1241, Sept. 1975.

[75] C.A. Siller, Jr., and W. Debus, 'Decision-Directed Fractionally-Spaced Equalizer Control Using Time-Domain Interpolation', *IEEE Trans. Commun.*, Vol. COM-39, No. 2, pp. 182-186, Feb. 1991.

[76] G. Strang, *Linear Algebra and its Applications*, 1976, p. 107, p. 221, p. 224.

[77] R.A. Tarbox, 'An Automatic Equalizer for Digital Repeatered Lines', *Proc. IEEE*, Vol. 57, pp. 363-364, March 1969.

[78] J.K. Tugnait, 'Blind Estimation of Digital Communication Channel Impulse Response', *IEEE Trans. Commun.*, Vol. COM-42, No. 2/3/4, pp. 1606-1616, Feb./March/Apr. 1994.

[79] G. Ungerboeck, 'Theory on the Speed of Convergence in Adaptive Equalizers for Digital Communication', *IBM J. Res. Develop.*, Vol. 16, pp. 546-555, Nov. 1972.

[80] G. Ungerboeck, 'Adaptive Maximum-Likelihood Receiver for Carrier-Modulated Data-Transmission Systems', *IEEE Trans. Commun.*, Vol. COM-22, pp. 624-636, May 1974.

[81] G. Ungerboeck, 'Fractional Tap Spacing and Consequences for Clock Recovery in Data Modems', *IEEE Trans. Commun.*, Vol. COM-24, pp. 856-864, Aug. 1976.

[82] N.A.M. Verhoeckx, H.C. van den Elzen, F.A.M. Snijders and P.J. van Gerwen, 'Digital Echo Cancellation for Baseband Data Transmission', *IEEE Trans. Acoust., Speech, Signal Process.*, Vol. ASSP-27, No. 6, pp. 768-781, Dec. 1979.

[83] N.A.M. Verhoeckx and T.A.C.M. Claasen, 'Some Considerations on the Design of Adaptive Filters Equipped with the Sign Algorithm', *IEEE Trans. Commun.*, Vol. COM-32, No. 3, pp. 258-266, March 1984.

[84] W.K. Victor and M.H. Brockmann, 'The Application of Linear Servo Theory to the Design of AGC Loops', *Proc. IRE*, Vol. 48, pp. 234-238, Feb. 1960.

[85] J.O. Voorman, P.J. Snijder, P.J. Barth, and J.S. Vromans, 'A One-Chip Automatic Equalizer for Echo-Reduction in Teletext', *IEEE Trans. Cons. Electr.*, Vol. CE-27, No. 3, pp. 512-529, 1981.

[86] F.D. Waldhauer, 'Quantized Feedback in an Experimental 280-Mb/s Digital Receiver for Coaxial Transmission', *IEEE Trans. Commun.*, Vol. COM-22, Jan. 1974.

[87] B. Widrow and M.E. Hoff, Jr., 'Adaptive Switching Circuits', in *IRE WESCON Conf. Rec.*, pt. 4, pp. 96-104, Aug. 1960.

[88] B. Widrow, J.M. McCool, M.G. Larimore and R. Johnson, Jr., 'Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter', *Proc. IEEE*, Vol. 64, No. 8, pp. 1151-1162, Aug. 1976.

[89] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, New Jersey: Prentice-Hall, 1985.

[90] H.W. Wong-Lam, J.W.M. Bergmans and K.D. Fisher, 'Analysis of Zero-Forcing Adaptation Algorithms', submitted to *IEEE Trans. Commun.*, 1995.

[91] R.W. Wood and D.A. Peterson, 'Viterbi Detection of Class IV Partial Response on a Magnetic Recording Channel', *IEEE Trans. Commun.*, Vol. COM-34, No. 5, pp. 454-461, May 1986.

[92] P. Xue and B. Liu, 'Adaptive Equalizer Using Finite-Bit Power-of-Two Quantization', *IEEE Trans. Acoust., Speech, Signal Process.*, Vol. ASSP-34, No. 6, pp. 1603-1611, Dec. 1986.

[93] C.L. Zahm, 'Applications of Adaptive Arrays to Suppress Strong Jammers in the Presence of Weak Signals', *IEEE Trans. Aerosp. Electron. Syst.*, Vol. AES-9, pp. 260-271, Mar. 1973.

9

BASICS OF TIMING RECOVERY

9.1 INTRODUCTION

So far we have assumed that the receiver knows the ideal decision instants to infinite precision. In practice this will not be true, and a local time-base must be established by means of a hierarchy of synchronization mechanisms. For carrier-modulated transmission systems this hierarchy starts with *carrier synchronization.* This concerns the generation of a reference carrier with a phase closely matching that of the modulated data signal. Next in line, and first for the baseband transmission systems considered here, is *timing recovery* (also called *clock recovery* or *bit synchronization*), which serves to recover a clock at the symbol rate that is phase-locked to the incoming data. The final task is *block synchronization*, sometimes subdivided into *word-*, *frame-* and *packet-*synchronization. Here the objective is to look for the start of a message or to break the data stream into constituent parts. This is usually accomplished by embedding predefined patterns into the data stream; these can be detected by means of correlation techniques. Carrier and block synchronization are beyond the scope of this text; we refer to [15], [62], [19], [58], [35, ch. 14] and [6], [42], [13, ch. 10], respectively, for a review. Timing recovery is the subject of this chapter and the two following ones.

To facilitate timing recovery, a clock signal may be transmitted separately from the data signal or be added to it at a low power level [65]. In many instances, however, the transmission of a separate clock requires additional facilities, power, or bandwidth. To avoid these inefficiencies, it is customary to recover the clock from the incoming data signal itself. This is referred to as *self-timing*. Our treatment of timing recovery is restricted to this approach. The potential of self-timing depends critically on the amount of timing information that is present in the incoming data signal. As data transmission becomes more efficient (in terms, for example, of modulation scheme, data constellation, excess bandwidth and SNR), the received waveform tends to look more irregular and noise-like. This reflects a decreased amount of timing information. At the same time, requirements on the accuracy of timing recovery tend to become increasingly severe. Both factors render timing recovery a more critical task, and have spurred the development of sophisticated schemes, some of which are described later on.

This chapter is concerned with basics of timing recovery. We discuss the timing recovery problem in some detail, introduce terminology, and discuss the role of code and channel (Section 9.2). We will see that most timing-recovery schemes take the form of a *phase-locked loop* (PLL), and more specifically of a *delay-tracking loop*. At the heart of the PLL is a *phase detector*, which serves to extract phase-error information from the incoming signal. For the PD of a delay-tracking loop we will use the more accurate term *timing-error detector* (TED). The remainder of the delay-tracking loop operates on the TED output in order to lock a local clock onto the incoming signal. The tracking capabilities of the loop are reviewed in Section 9.3. They depend heavily on the amount of control information that the TED manages to extract from the incoming signal per unit of time and SNR. This amount may be characterized by the *efficiency* of the TED (Section 9.3.3). The tracking mode of operation is preceded by a phase in which the loop acquires lock. This phase is discussed in Section 9.4. Many applications call for rapid acquisition. Section 9.5 describes various acquisition aids that are meant to improve acquisition speeds of the 'bare' timing-recovery loop, while Section 9.6 derives fundamental limits that govern these speeds.

After this introductory exposure to timing recovery, Chapter 10 presents a catalog of timing-recovery schemes that covers the optimum approach, viz. maximum-likelihood timing recovery, as well as a number of more practical approaches. The emphasis in Chapter 10 is on what sets distinct schemes apart, i.e. on how timing information is extracted from the incoming signal. The common trait of the schemes is that they all have the form of a PLL. Properties of PLLs are studied in a general setting in Chapter 11.

9.2 TIMING-RECOVERY SCHEMES

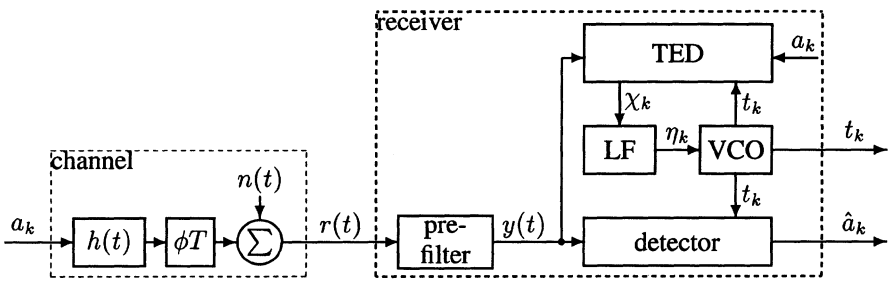


Figure 9.1: System model.

In Fig. 9.1, a data sequence a_k of data rate $1/T$ is applied to a channel with impulse response $h(t)$, additive noise $n(t)$, and an *a priori* unknown and possibly time-varying delay ϕ (expressed in symbol intervals T). The receiver operates on the received signal $r(t)$ to produce decisions $\hat{a}_k$ with respect to a_k as well as a clock signal that indicates the decision instants $t_k = (k + \psi)T$. The normalized sampling phase ψ must must closely approach ϕ in order for the detector to function properly. To achieve this, the instants t_k

are demarcated by a timing-recovery subsystem that takes the form of a phase-locked loop (PLL). This loop consists of a timing-error detector (TED), a loop filter (LF), and a voltage-controlled oscillator (VCO). The TED serves to produce an indication χ_k of the sampling phase error $\Delta \triangleq \phi - \psi$. PLL properties depend in part on the LF and VCO and are treated in Chapter 11.

The TED in Fig. 9.1 bases its output χ_k in part on the prefilter output $y(t)$; in some cases the prefilter input $r(t)$ or a signal inside the detector is used instead. The TED further uses the data sequence a_k and is therefore of the *data-aided* type. This data sequence may be available to the receiver in the form of a known *clock preamble* or *clock run-in pattern* that precedes the actual data, or in the form of decisions $\hat{a}_k$ once decisions have become sufficiently reliable. The latter mode of operation is called *decision-directed* (DD) and is especially useful to track delay variations once the PLL has locked up. Conversely, DD operation may be less suitable for the loop to acquire lock because of the comparatively high likelihood of decision errors during this phase. Clock run-in patterns, by comparison, do permit the actual data symbols to be used during acquisition. Unfortunately, such patterns are not always available. In that case the best way to go may involve the use of a *non-data-aided* (NDA) TED. By definition, NDA schemes perform timing recovery without reference to the transmitted data sequence. This may improve acquisition properties relative to DA schemes, but often yields poorer tracking properties.

9.2.1 Timing errors

In Fig. 9.1, the delay ϕ is modeled as fixed and arising at a single point in the system. Reality, however, is a bit more complicated.

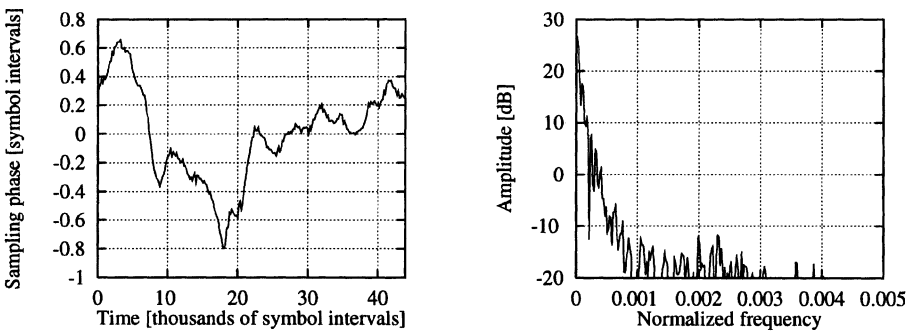


Figure 9.2: Sample function $\phi(t)$ measured in an experimental digital video tape recorder (left) and its amplitude spectrum (right).

Example 9.1 The left part of Fig. 9.2 shows a sample function $\phi(t)$ as measured for an experimental digital helical-scan video tape recorder with the aid of a crystal-controlled high-speed digitizer. The random behavior is due, among other factors, to slight fluctuations of the tape-to-head speed upon recording and replay. The peak-to-peak magnitude of $\phi(t)$ amounts to more

than one symbol interval T. This is far too much to be left uncompensated in the receiver. It takes several thousands of symbol intervals for ϕ to change significantly. This 'slowness' translates spectrally into a low-pass nature (Fig. 9.2, right). Components with a frequency above 0.1% of the bit rate are as good as absent and need not be tracked. This determines the required bandwidth of the PLL (see Chapter 11).

Exercise 9.1 In practice the average tape-to-head speeds upon recording and replay may be slightly different. How does this affect Fig. 9.2?

We can generally subdivide ϕ into deterministic and random components according to

$$\phi(t) = \phi_0 + \dot{\phi}\frac{t}{T} + \theta(t) \tag{9.1}$$

where ϕ_0 represents a static delay, $\dot{\phi}$ is a frequency offset normalized with respect to the signaling rate $1/T$, and $\theta(t)$ is the random component of ϕ. The PLL is required to track ϕ as well as possible. We will see in Chapter 11 that this is, in principle, possible to arbitrary precision for the deterministic components of ϕ. Conversely, a certain tracking error is generally inevitable for the random component $\theta(t)$ that we call *jitter* or *phase noise*.[1] The *jitter spectrum* $P_\theta(\Omega)$, i.e. the power spectral density of $\theta(t)$, is typically a low-pass function with a bandwidth much smaller than the signaling rate $1/T$ (compare Fig. 9.2.b). Components of $\theta(t)$ with a very low frequency compared to the signaling rate (in the order, say, of minutes, hours or days) are sometimes called *time-base wander* or just *wander*. It is typically not difficult to track wander accurately. Conversely, the remaining jitter components can be tracked only in as far as their frequency components are within the PLL bandwidth. This bandwidth, in turn, is restricted by noise-induced jitter, loop stability, and other considerations (see Chapter 11).

In the foregoing example we were deliberately vague about the precise point in the system at which delays and frequency offsets emerge. The following example reveals that this is permissible subject to mild conditions.

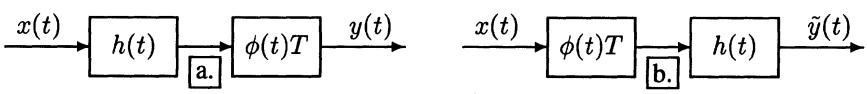


Figure 9.3: Two systems with a time-varying delay ϕ and a filter with impulse response $h(t)$.

Example 9.2 In Fig. 9.3.a, a signal $x(t)$ is applied to a filter with impulse response $h(t)$ and then delayed by a normalized delay $\phi(t)$. In Fig. 9.3.b the order of operations is reversed. Both configurations are linear but time-variant. As a result, the two outputs $y(t)$ and $\tilde{y}(t)$ are, in general, unequal. However, for delays $\phi(t)$ and impulse responses $h(t)$ of interest to us, the duration of $h(t)$ will be much smaller than the time that it takes for ϕ to change significantly. Then $y(t) \simeq \tilde{y}(t)$ with good approximation, i.e. we can think of distributed delays as if they were

[1] More generally, any random fluctuation of a delay is referred to as jitter or phase noise. Thus ψ and $\phi - \psi$, for example, can also exhibit jitter. In cases where confusion can arise we shall explicitly name the delay that is alluded to.

concentrated at a single point, just as in Fig. 9.1. In the sequel we will repeatedly make use of this fact without stating so explicitly.

9.2.2 Deductive versus inductive timing recovery

The detector in Fig. 9.1 has a continuous-time input. In practice one often uses a detector with a discrete-time input, preceded by a sampler that serves to demarcate the decision instants t_k. Timing-recovery schemes come basically in two configurations, depending on whether timing information is extracted before or after this sampler. *Deductive* schemes (Fig. 9.4) extract all timing information before the sampler, while *inductive* schemes (Fig. 9.5) extract timing information behind it.

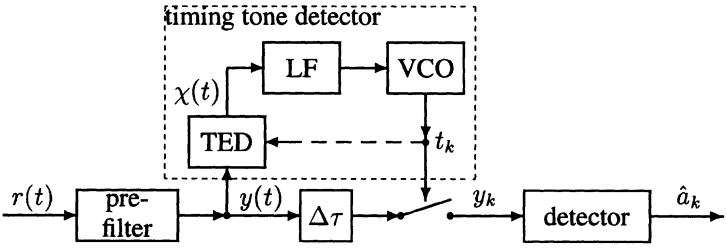


Figure 9.4: Deductive timing-recovery scheme. Timing information is extracted before the sampler.

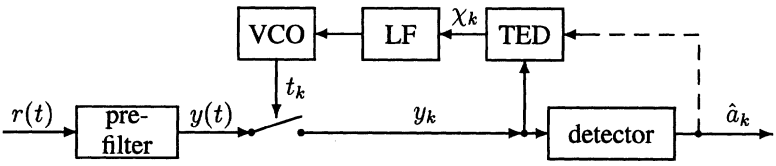


Figure 9.5: Inductive timing-recovery scheme. Timing information is extracted behind the sampler.

Deductive schemes are by necessity non-data-aided (NDA) and are typically - though not always - analog. One of their disadvantages is that unknown delay differences $\Delta\tau$ between the timing recovery and data paths, as caused, for example, by component tolerances and temperature fluctuations, cannot be detected and corrected. This problem is nonexistent in inductive schemes because both paths are merged up until the detector. Inductive schemes are typically digital and often decision-directed (DD). Some inductive schemes require an additional sampler or an increment of the sampling rate with respect to the symbol rate $1/T$.

In deductive schemes, the complete circuit that extracts the decision instants t_k from the incoming signal $y(t)$ is sometimes referred to as a *timing tone detector*. This detector typically contains a TED, LF and VCO as in Fig. 9.4, but other configurations are

also possible. Cases in point include the scheme of Section 10.4.6 and the one that is briefly discussed at the end of Section 10.4.4. Another example is illustrated in Fig. 9.6.

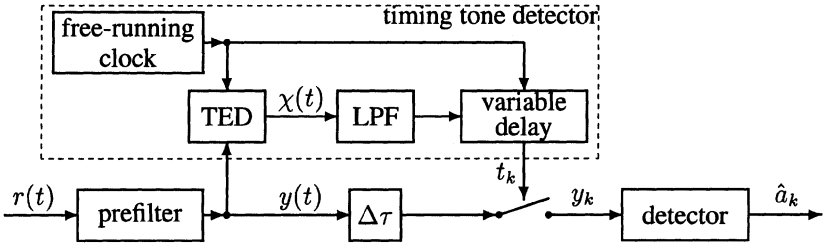


Figure 9.6: Open-loop deductive timing-recovery scheme.

The scheme of Fig. 9.6 contains a free-running clock whose frequency is nominally equal to the symbol rate $1/T$. The phase of the clock is compared with that of the incoming signal in a TED, whose output $\chi(t)$ is applied to a low-pass filter so as to suppress random disturbances. The LPF output is a measure of the relative delay of clock and received signal, and controls a variable-delay element so as to compensate for this relative delay. Clock, TED, LPF and variable-delay element together constitute the timing tone detector. Since any variation in these building blocks may affect the output t_k of this detector, it is essential that their characteristics be accurately known. This will generally favor a digital implementation.

By virtue of the periodicity of the clock signal, variable delays τ larger than one clock period T_c can be reduced modulo T_c without affecting the clock signal. In the same way a negative delay τ can also be replaced by a positive fractional delay, i.e. a delay between 0 and T_c seconds. It is, therefore, permissible for τ to decrease or increase continuously over time. This means that both positive and negative frequency offsets between $1/T_c$ and $1/T$ can, in principle, be handled.

Since only the relative delay of clock and received signal is of interest, it is allowable to replace the delay τ in the clock path by a delay $-\tau$ (or $T_c - \tau$ for reasons of causality) in the data path. The combination of that delay and the subsequent sampling operation can be conveniently approximated in the discrete-time domain with the aid of an interpolator, as we will see in Section 9.2.3. An all-digital timing recovery circuit along these lines is discussed in Section 10.4.6.

The scheme of Fig. 9.6 is said to be of the *open-loop* type because it contains no feedback loops.[2] The absence of feedback loops minimizes acquisition problems (see Section 9.4). Inductive schemes, by comparison, inherently contain a feedback loop and are therefore of the *closed-loop* type. The deductive scheme of Fig. 9.4 deserves both adjectives to a certain extent because it does contain a feedback loop, but one that is 'local' to the timing tone detector.

[2]Some of the earliest timing-recovery schemes are of the open-loop type [20], [31]. These schemes are meant for modulation codes such as FM that have a mid-bit transition every symbol interval (compare Fig. 4.44). Decision instants are derived from these transitions via a fixed delay (see also Problem 10.21). A modern variation on this theme, suitable for arbitrary modulation codes, is [5].

9.2.3 All-digital timing recovery

In Figs. 9.4 and 9.5, timing recovery is largely performed by analog means. At low data rates it is sometimes attractive to implement the receiver with the aid of a digital signal processor (DSP). The timing of such a processor is usually defined by a crystal and is then inevitably asynchronous with the transmitted data (Fig. 9.7).

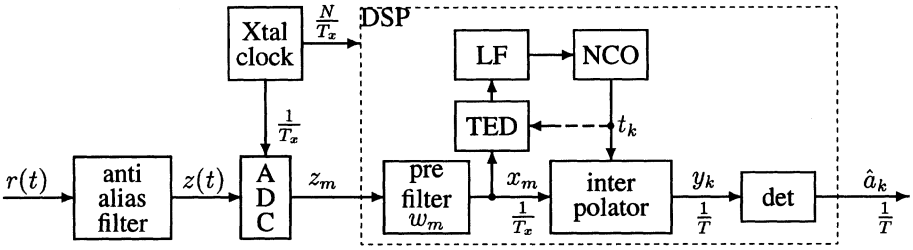


Figure 9.7: Almost-digital receiver with deductive timing recovery.

The analog-to-digital converter (ADC) is clocked at a sampling rate $1/T_x$ that is high enough to prevent aliasing. The processor runs at a multiple N/T_x of this sampling rate. Here N is an oversampling factor that determines the number of available instructions for each cycle of operation of the receiver.

Example 9.3 In the all-digital CD-receiver of [59], $1/T = 4.3218$ MHz nominally while $1/T_x = 8$ MHz. Hence $1/T_x$ is not much higher than $1/T$ and this is allowable because of the the fact that frequencies above 1 MHz are as good as absent in the replay signal.

The timing-recovery scheme of Fig. 9.7 is of the deductive type. All-digital deductive schemes were first suggested by Barton [7] and have been applied, for example, in the radio receiver of [4], in the digital compact cassette (DCC) system [33], and in the CD receiver of [59]. All-digital inductive schemes were first suggested by Miller [45].

The prefilter in Fig. 9.7 operates at the same rate $1/T_x$ as the ADC, while decisions $\hat{a}_k$ are only needed once every T seconds. Asynchronous-to-synchronous conversion is performed with the aid of an interpolator under the control of a numerically controlled oscillator (NCO). These two blocks are the digital counterparts of the sampler and VCO in Figs. 9.4 and 9.5. For simplicity the prefilter and interpolator can be combined into a single interpolating filter. Appendix 9A discusses the topology of Fig 9.7 to greater detail, and addresses the issue of interpolator design (see also [24, 18]).

9.2.4 Effect of data sequence on timing recovery

Uncoded binary data strings may contain long runs of ones or zeros. During such a run, no new timing information is generated. The receiver will make erroneous decisions if the run lasts too long for the old PLL estimates of clock frequency and phase to remain valid. As discussed to greater detail in Section 4.5, there are two main ways to avoid this.

1. Data may be randomized by means of a *scrambler* [55, 25]. This will make the probability of a long run of ones or zeros small, though not exactly zero. This approach does not require the introduction of redundancy and has found widespread application in data modems.

2. A modulation code may be used to limit the maximum run-length of ones and zeros [56, 57, 41]. Both *DC-free* and *run-length-limited* codes can be used to this end. Codes of these types are widely applied in digital recording systems. They do require the introduction of redundancy and are thereby more wasteful of bandwidth than scramblers. On the credit side, data transitions are certified to occur regularly. As an example, the compact disc (CD) system uses the EFM modulation code, which has a maximum run-length of 11 symbol intervals. With EFM, self-timing is quite practicable, even though the CD channel has little or no excess bandwidth.

During transmission startup, the actual user data may be preceded by a fixed *training sequence* or *preamble*. Preambles often contain a clock run-in pattern that is meant to facilitate clock acquisition. This pattern is typically periodic and invariably contains many transitions so as to convey much timing information. In Section 9.6 we will analyze what acquisition speeds are fundamentally attainable for a given data pattern.

9.2.5 Effect of excess bandwidth

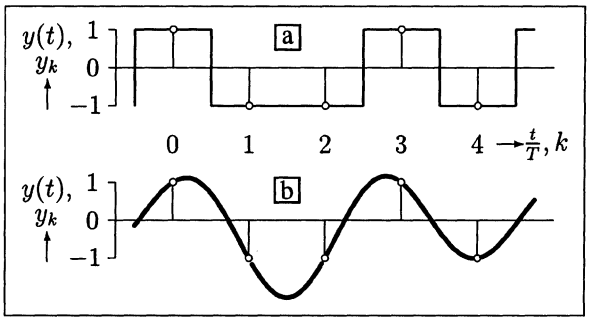


Figure 9.8: Binary data sequence carried by rectangular NRZ pulses (a) and Nyquist-1 pulses with negligible excess bandwidth (b).

Fig. 9.8 depicts two signals without intersymbol interference that carry identical information yet have different bandwidth. The NRZ signal has sharp transitions. These demarcate the borderline between adjacent bit cells without any ambiguity, and thereby convey much timing information. By contrast, the signal of Fig. 9.8.b has little excess bandwidth and is very much 'rounded'. The reader might not have managed to infer the required sampling instants from the waveform if the underlying data sequence had not been indicated in the graph. Somewhat similarly, timing-recovery schemes tend to

break down when excess bandwidth vanishes if they do not have side-information about the transmitted data sequence (i.e. if they are not data-aided).

For the NRZ signal, all timing information is concentrated at the single sampling phase where transitions occur. As bandwidth decreases, the signal gets 'rounded' and timing information gets more evenly spread across sampling phases (Fig. 9.8.b). In the limiting case of no excess bandwidth, the sampling theorem ascertains that no information whatsoever is lost with baud-rate sampling, irrespective of the sampling phase. A direct implication is that all sampling phases will convey the same amount of timing information. A particularly convenient choice is to base timing recovery on the same sampling phase that is used for data detection. This is what happens in all data-aided baud-rate schemes of Chapter 10.

In Section 10.2.3 we will show more formally that timing information resides in particular at high frequencies. Thus accurate timing recovery is intrinsically most difficult when excess bandwidth is small. Unfortunately, it is exactly in this situation that accurate timing-recovery tends to be most critical to detector performance.

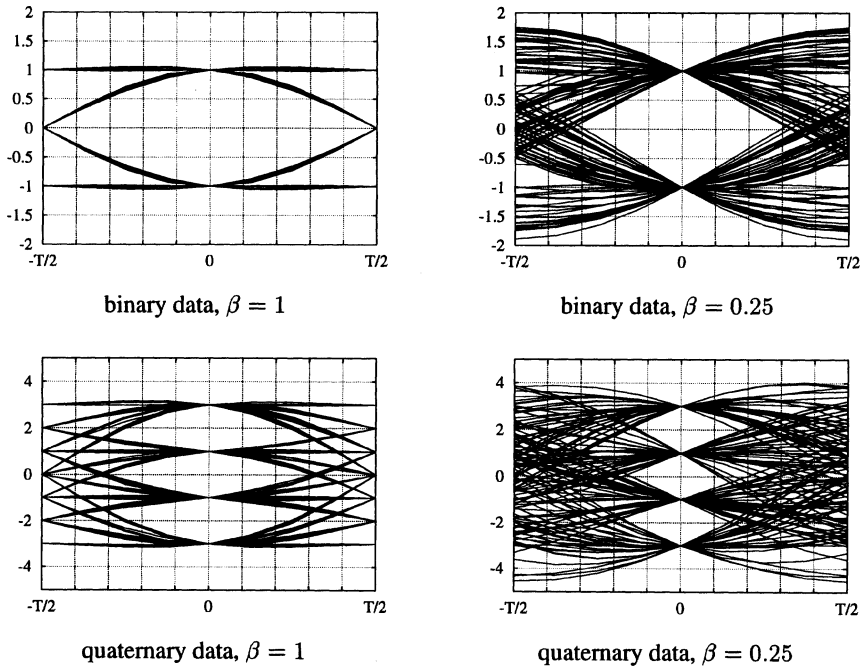


Figure 9.9: Eye patterns for a channel without noise that has been equalized into a raised cosine characteristic with roll-off parameter β.

Example 9.4 Fig. 9.9 depicts binary and quaternary eye patterns for a channel without noise that has been equalized into a raised-cosine characteristic. Decisions are ideally taken in the

middle of the 'eyes'. Timing errors cause the sampling instants to shift away from this ideal. Noise and intersymbol interference tend to lower the eye height.

The binary eyes are all considerably higher than their quaternary counterparts. This yields a larger margin against noise and ISI. Consequently, channel parameter variations can be larger, and accuracy requirements on the equalizer are less severe. Binary data also yields favorable eye widths and thereby a larger margin against timing errors.

Eye widths are a strong function of excess bandwidth. They are largest for $\beta = 1$ (i.e. for an excess bandwidth of 100%), but vanish as β approaches zero (see Problem 5.1), most rapidly so for quaternary data. Here transmission is most bandwidth-efficient, but requirements on the accuracy of timing recovery become extremely severe.

This example illustrates that accuracy requirements on timing recovery typically grow along with those on equalization. This explains in part why powerful data-aided timing-recovery schemes often supplement adaptive equalization techniques, which were discussed in Chapter 8. The interaction between both is subtle and requires judicious engineering (see Example 8.10). Example 9.4 may suggest that systems without excess bandwidth cannot withstand any timing errors. This is, however, not generally true. We give two counterexamples.

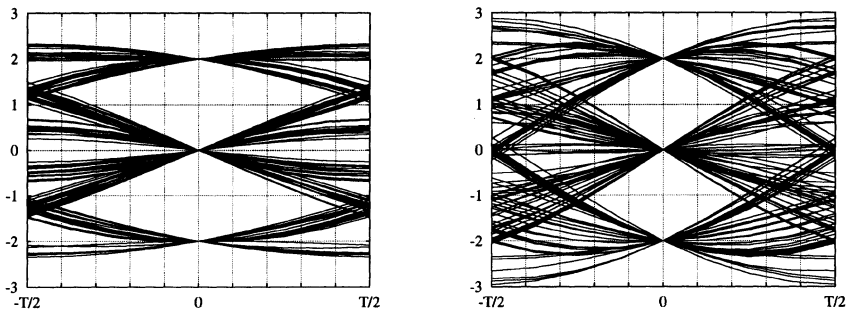


Figure 9.10: Eye patterns of duobinary (left) and Class IV (right) partial-response signals without excess bandwidth.

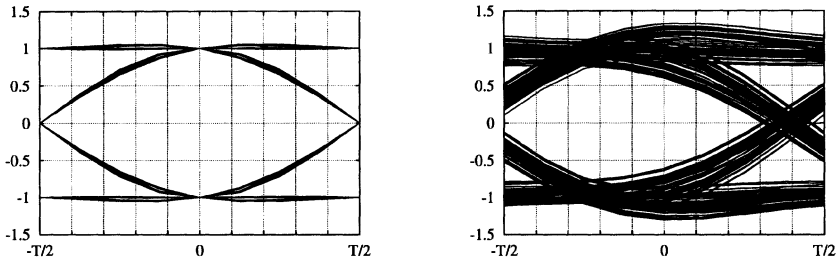
Example 9.5 Fig. 9.10 shows eye patterns of duobinary $(1+\mathcal{D})$ and Class IV $(1-\mathcal{D}^2)$ partial-response systems with binary data. Both systems have no excess bandwidth, yet eye widths are distinctly nonzero.

The effect of timing errors on partial-response systems is elaborated in detail in [34]. Performance consequences of timing errors for a Lorentzian recording system are studied in [10].

9.2.6 Effect of misequalization

Throughout the present and following chapter we assume for simplicity that the system is perfectly equalized. In practice, however, a certain degree of misequalization is generally unavoidable. This may affect the steady-state phase at which the timing-

recovery subsystem settles down, and may also increase jitter. Of particular concern is phase distortion. We illustrate this for a full-response system.



Ideal eye pattern (no phase distortion)

Actual eye pattern (with phase distortion)

Figure 9.11: Binary eye patterns of full-response system with 100% excess bandwidth.

Example 9.6 Let us consider a system with binary data and a system response that is nominally a raised cosine pulse with 100% excess bandwidth. The corresponding eye pattern is depicted in Fig. 9.11 (left). In Section 10.3 we will see that the sampling instants may be derived from the instants at which the eye crosses zero. This technique is ideally suited for the left eye pattern of Fig. 9.11 because all zero crossings occur exactly halfway between the ideal decision instants. This means that a jitter-free clock can be produced, and that the ideal decision instants can be found at a fixed delay of $T/2$ seconds from the zero crossings.

Let us now assume that the actual system exhibits phase distortion such that the right eye pattern of Fig. 9.11 emerges; the amplitude-frequency characteristics have not changed. Even though the excess bandwidth is still 100%, the zero crossings have become diffuse. A direct consequence is that jitter will arise; this jitter is called *pattern-dependent* because it is induced by the data. Even more seriously, the average location of the zero crossings is no longer exactly $T/2$ seconds away from the sampling phase at which the eye is most open. This means that the timing-recovery loop will settle down at a phase that is suboptimum for data detection. In Section 10.3 we will see that amplitude distortion is less cumbersome in both respects.

Similar effects occur with a variety of timing-recovery schemes, largely irrespective of the employed detection technique. Another illustration and further comments may be found in [53]. Problem 10.8 is concerned with a mathematical analysis for the maximum-likelihood timing-recovery scheme of Section 10.2.2. The arguments of Problem 10.8 can be extended to many other timing-recovery schemes.

9.3 TRACKING

Tracking properties of the timing-recovery loop of Fig. 9.1 depend heavily on the timing-error detector (TED). The present section is concerned with these properties, and more in particular with the clock jitter that is induced by the data and noise. For

the sake of brevity we are not concerned with open-loop configurations like the one of Fig. 9.6. The performance of such schemes is studied, for example, in [47] and [4].

9.3.1 Timing-error detectors

The objective of the TED is to provide an indication of the sampling-phase error Δ. The TED output χ_k is averaged out by the loop of Fig. 9.1. The expected value $E[\chi_k]$ constitutes the desired component of χ_k and may be written $E[\chi_k] = \rho(\Delta)$ (we recall that Δ is the sampling-phase error normalized in units T.) The function $\rho(\Delta)$ is said to be the *TED transfer characteristic* and is also called *timing function* or *S-curve*. An example is shown in Fig. 9.12.

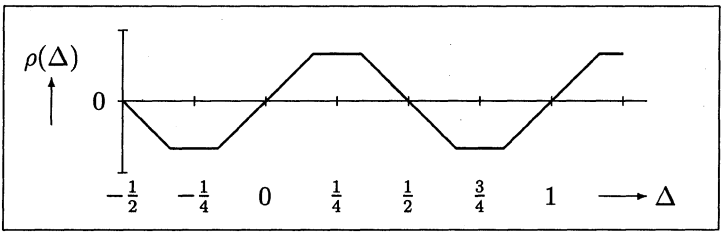


Figure 9.12: Example of a TED transfer characteristic.

The function of Fig. 9.12 is periodic in Δ and exhibits zero-crossings at $\Delta = 0, \pm\frac{1}{2}, \pm 1, \ldots$. The PLL acts to force χ_k, upon average, toward zero. To this end, the VCO sampling phase is advanced or retarded via the loop filter in proportion to χ_k. As a result, the sampling phase will settle at $\Delta = 0$ or at another zero crossing of ρ with a positive slope (i.e. $\Delta = \pm 1, \pm 2, \ldots$). All of these solutions are permissible. Their only difference is that a given bit is detected an integer number of bit intervals earlier or later. This ambiguity is irrelevant to data detection and is normally resolved prior to modulation decoding by a word synchronization mechanism. The spurious zero-crossings $\Delta = \pm\frac{1}{2}, \pm\frac{3}{2}, \ldots$ correspond to unstable equilibria at which the loop may dwell for some time. This phenomenon is called *hangup* or *false lock*.

TED transfer characteristics $\rho(\Delta)$ ideally have a zero crossing for $\Delta = 0$ as in Fig. 9.12. In practice this crossing can shift due to phase distortion of the channel, delay mismatches between clock and data paths, and various other artefacts. For the system to function properly it is necessary that these artefacts be sufficiently small to ensure that the loop will indeed settle down at or very near the correct sampling phase.

The slope K_d of the transfer characteristic in the origin ($\Delta = 0$) may be either positive or negative, as long as this polarity is known beforehand. A negative polarity must be compensated by an additional inversion elsewhere in the loop. Would $\rho(\Delta)$ constitute the only component of χ_k, then the PLL would come to rest at the ideal sampling phase. In practice, however, χ_k inevitably contains random disturbances that cause Δ to exhibit jitter. These are:

1. A component z_k that depends only on the data sequence a_k, and induces *pattern-dependent jitter* (also called *pattern noise*, *self-noise* or *systematic jitter*) of Δ_k.

2. A component u_k that is caused by noise $n(t)$ of the channel. Consequential jitter of Δ_k is said to be *noise-induced.* Normally u_k is linearly proportional to $n(t)$.

9.3.2 Pattern-dependent jitter

The desired TED output $\rho(\Delta)$ is quasi-static. To retain this component while suppressing jitter, the loop of Fig. 9.1 invariably has a bandwidth much smaller than the symbol rate. As a result, all disturbances will be rejected except for components near DC ($\Omega = 0$). Since the PSD of z_k is usually a smooth function of frequency, its value at DC will largely determine the amount of pattern-dependent jitter. In the tracking mode of operation (i.e. for small $|\Delta|$) it is often possible to ensure that z_k is DC-free [22] or that it vanishes altogether (see, for example, the deductive scheme of Section 10.3.4 with $\beta = 1$, the one of Section 10.4.4, and the error-based inductive schemes of Sections 10.6 and 10.7). In all of these cases, pattern-dependent jitter is bound to be small. For this reason we shall pay special attention to noise-induced jitter in what follows, and unless stated otherwise we shall neglect z_k.

9.3.3 Noise-induced jitter

Let us assume that the loop is in the tracking mode of operation and is functioning properly so that $|\Delta|$ is small. Then we may linearize the TED transfer characteristic around the operating point $\Delta = 0$ (Fig. 9.13).

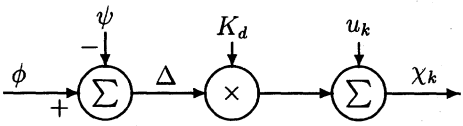


Figure 9.13: Linearized phase-domain TED model.

The TED compares the actual delay ϕ of the channel with the VCO phase ϕ (both ϕ and ψ are normalized in symbol intervals). It has gain K_d and introduces an additive disturbance u_k with power spectral density $\mathcal{U}(e^{j2\pi\Omega})$. Loop properties in Fig. 9.1 are determined by a loop gain of which the TED gain K_d forms a part. By way of illustration we consider a first-order loop.

Example 9.7 Fig. 9.14.a depicts the phase-domain model of a first-order discrete-time PLL. The integrator in Fig. 9.14.a models the frequency-to-phase conversion of the VCO of Fig. 9.1. The constant gain factor K_o accounts for the sensitivity of the VCO. The loop filter of Fig. 9.1 degenerates into a constant gain factor K_l. For a more detailed discussion of this model we refer ahead to Chapter 11.

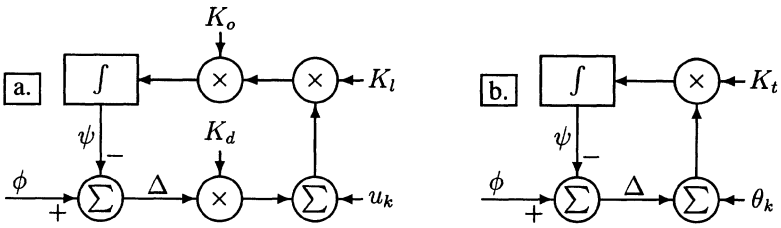


Figure 9.14: a. Phase-domain model of first-order discrete-time PLL. b. Simplified model with total open-loop gain $K_t = K_d K_l K_o$ and input-referred noise $\theta_k = u_k/K_d$.

The model may be simplified by lumping all gain factors into a total open-loop gain $K_t \triangleq K_d K_l K_o$. To this end we must refer u_k back towards the TED input. This involves a scaling factor $1/K_d$. The input-referred noise u_k/K_d is denoted θ_k. We arrive at Fig. 9.14.b. The loop of Fig. 9.14.b is analyzed in Section 11.3. Major properties are as follows.

Stability: The loop is stable if $0 < K_t < 2$. For a stable loop, the VCO phase ψ will settle down at the ideal value ϕ if noise θ_k is absent.

Loop bandwidth: The loop has a low-pass nature, i.e. slow variations of ϕ and θ_k are tracked by the VCO phase ψ while fast variations are rejected. Normally $K_t \ll 1$. Then the normalized -3 dB cut-off frequency Ω_c (i.e. the cut-off frequency relative to the symbol rate $1/T$) satisfies $\Omega_c \simeq K_t/(2\pi)$. This frequency determines the tracking speed of the PLL, i.e. the fastest variations of ϕ that the VCO can track.

Noise suppression: Noise components outside the pass-band of the loop are rejected. If $K_t \ll 1$, then only components of θ_k near DC will be passed, and the resulting jitter Δ_k has variance $\sigma_\Delta^2 \simeq B_l P_\theta(1)$ where B_l is the equivalent white-noise bandwidth of the loop and $P_\theta(1)$ is the power spectral density of θ_k at DC ($e^{j2\pi\Omega} = 1$). For $K_t \ll 1$ one has $B_l \simeq K_t/2$ and therefore

$$\sigma_\Delta^2 \simeq B_l P_\theta(1) \simeq \frac{K_t}{2} P_\theta(1). \tag{9.2}$$

For a given tracking speed (i.e. for a fixed value of K_t), jitter can be reduced only by lowering $P_\theta(1) = \mathcal{U}(1)/K_d^2$. This quantity is fully determined by the timing-error detector; the remainder of the loop does not come into play.

A similar role is played by $P_\theta(1)$ in second- and higher-order loops. In most cases u_k and θ_k are linearly proportional to the noise $n(t)$ of the channel. This causes $\mathcal{U}(1)$ and $P_\theta(1)$ to depend linearly on the power of $n(t)$. To arrive at an intrinsic figure of merit for the noise rejection by the the TED we must eliminate this dependence. To this end we will use the channel signal-to-noise ratio SNR as a scaling factor. We recall from Chapter 3 that SNR governs the matched filter bound, which sets an upper limit to the performance of the detector in Fig. 9.1. The *TED efficiency*

$$\gamma \triangleq \frac{1}{\text{SNR}} \frac{1}{P_\theta(1)} = \frac{1}{\text{SNR}} \frac{K_d^2}{\mathcal{U}(1)} \tag{9.3}$$

is, therefore, a measure of the relative ease of accurate timing recovery and reliable bit-detection. This measure is similar in concept to the efficiency of adaptive filters (see [63]). We will see later that γ tends to grow rapidly with the bandwidth of the channel; efficiencies are poorest when there is little or no excess bandwidth.

For future reference we recall that SNR is fully determined by the per-symbol spectral signal-to-noise ratio

$$\mathrm{SN}(\Omega) \triangleq \frac{1}{T}\frac{|H(\Omega)|^2}{\mathcal{N}(\Omega)} \tag{9.4}$$

where $H(\Omega)$ is the Fourier transform of $h(t)$ and $\mathcal{N}(\Omega)$ is the power spectral density of $n(t)$. Specifically

$$\mathrm{SNR} = \int_{-\infty}^{\infty} \mathrm{SN}(\Omega)d\Omega. \tag{9.5}$$

The variable Ω is a normalized measure of frequency, with $\Omega = 1$ corresponding to the signaling rate $1/T$.

It is worth noting that the variance of θ_k is sometimes used instead of $P_\theta(1)$ as a figure of merit for the TED (see, for example, [46]). This figure is equivalent to $P_\theta(1)$ when θ_k is uncorrelated, but becomes less significant to PLL properties when θ_k is correlated. Since γ does not account for pattern-dependent jitter, estimates of tracking performance that are based on γ alone can be too optimistic whenever this jitter is significant. We shall later encounter a few situations where this is so.

The following example illustrates how γ affects PLL properties. A more detailed discussion of these properties may be found in Chapter 11.

Example 9.8 Let us reconsider the first-order PLL of Fig. 9.14. By (9.2) and (9.3), noise-induced jitter Δ_k has variance

$$\sigma_\Delta^2 \simeq \frac{K_t}{2}P_\theta(1) = \frac{K_t}{2\gamma\mathrm{SNR}}. \tag{9.6}$$

A choice for K_t can be based, for example, on the requirement that RMS jitter σ_Δ should be restricted to a few percents of a symbol interval even at the poorest SNRs that permit somewhat reliable bit detection. For example, to achieve $\sigma_\Delta^2 = 10^{-3}$ at SNR = 10 dB with a realistic efficiency $\gamma = 1$, K_t should not be larger than 0.02. Hence Ω_c should not be larger than approximately 3.10^{-3}. This bandwidth will halve if K_t is halved.

Exercise 9.2 Consider two TEDs with efficiency γ and 2γ, respectively. Assume that the loop is dimensioned to achieve a given RMS jitter σ_Δ. How do the tracking speeds for the two TEDs compare? How do RMS jitter levels compare when the tracking speed is prescribed?

9.4 ACQUISITION

Before reliable bit-detection is possible the timing-recovery loop must first acquire lock.[3] The admissible duration of the acquisition phase depends heavily on the application.

[3]In the sections ahead we restrict attention to timing-recovery schemes in the form of a delay-tracking loop. Open-loop schemes as in Fig. 9.6 have a highly predictable acquisition behavior. Their acquisition

Example 9.9 Compact disks contain a continuous data stream with essentially no interruptions. During normal playback, acquisition will only be needed before the start of a music fragment and is then permitted to take in the order of a second, i.e. around 10^6 symbol intervals. It is easy to stay well below this bound [59].

Example 9.10 In hard disk drives, data is written in concentric *tracks* that are subdivided into *sectors* of about 512 bytes (4096 bits). Each sector is preceded by a *preamble* that serves, among other purposes, for the PLL to acquire lock. Preambles contain only some 100 bits so as to restrict preamble overhead to a few percents. Clock acquisition must, therefore, occur in well under 100 symbol intervals. This generally requires some form of *aided* acquisition (see Section 9.5).

Initially, the free-running frequency of the VCO usually differs from the data rate. Therefore the loop must first adjust the VCO frequency to approach that of the incoming signal. This acquisition stage is called *frequency pull-in* and its duration is called the *pull-in time*. The *pull-in range* (*capture range*, *acquisition range*) is the frequency range over which the loop can acquire lock. The second acquisition stage is *phase lock-in*. Here the VCO phase is matched to that of the incoming signal, and the loop no longer slips any cycles. (A cycle slip is said to occur when the VCO adds or drops a cycle of oscillation relative to the incoming signal.) The duration of the phase lock-in stage is called the *lock time* and is normally much smaller than the pull-in time. Phase lock-in is possible even with a small initial frequency error. The *direct lock range* (*seize range*) is the frequency range over which an unlocked loop can acquire lock without cycle slip. This range is typically on the order of the loop bandwidth [17, p. 60]. More details on these ranges and times are provided in Section 11.6.

The total acquisition time depends, among other factors, on the initial frequency and phase offsets between the incoming signal and the VCO, on the loop bandwidth, on the method that is used to extract timing information, and on the data pattern that is used for acquisition. We proceed to examine these factors.

Initial frequency and phase offsets Initial frequency offsets are due to uncertainties in both data rate and VCO frequency.

Example 9.11 In narrowband ISDN transmission as well as in voiceband modems, data rates are determined by a crystal and are therefore accurately known. In the Digital Compact Cassette (DCC) system, the average tape-to-head velocity is only accurate to within a few percents. The instantaneous velocity can be off by as much as tens of percents as a result of shocks and mechanical vibrations, and the data rate can vary accordingly.

Example 9.12 Crystal-based VCOs have an accurately determined free-running frequency, but are typically expensive and can take long to react to a change of the input voltage. Moreover, their actual frequency can usually not deviate much from the free-running frequency. Integrated oscillators, on the other hand, can be designed to go without these disadvantages, but require on-

---

time is largely determined by the total delay of TED, low-pass filter and interpolator, and is typically small. In any given application, this advantage must be weighted against various drawbacks (see Section 9.2.2).

chip resistors and/or capacitors whose values can easily have a spread of 10 percents. The free-running frequency of the VCO can vary accordingly if no external adjustments are permitted.

Most timing-error detectors have a periodic transfer characteristic, i.e. their output is essentially an estimate of the phase error modulo some period. Initial frequency offsets cause the phase error to traverse many such periods during acquisition. The resulting average TED output is small and contains little meaningful information to 'drive' the loop. This heuristic argumentation explains why the pull-in time tends to increase rapidly with the initial frequency offset between the data rate and VCO; the relation is often quadratic [17]. Many timing-recovery subsystems contain one or more acquisition aids that are aimed at alleviating this problem. Several of these aids are described in Section 9.5.

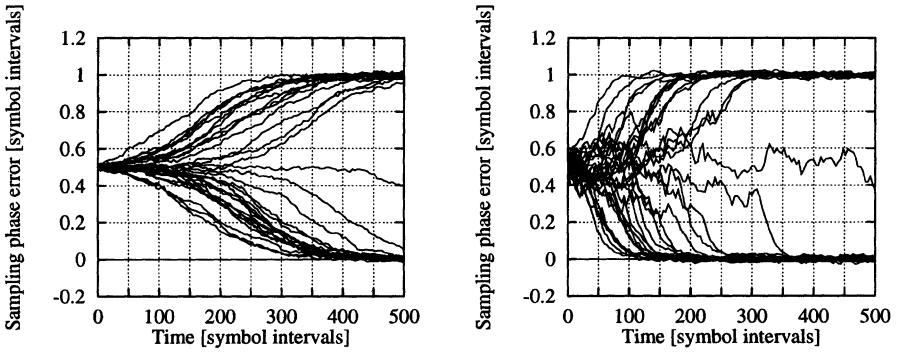
The lock time is a strong function of the phase offset between incoming signal and VCO at the end of the frequency pull-in stage. This offset is normally a random variable that cannot be controlled. The timing-recovery loop must be designed to seize sufficiently rapidly even for worst-case phase offsets. In some cases a reasonable initial guess about the phase offset can be made, notably for periodic preambles. This permits the VCO to be restarted with a small initial sampling-phase error. The zero-phase restart technique of Section 10.3.10 is an example of this approach.

Loop bandwidth Enlargement of loop bandwidths helps to speed up acquisition but increases steady-state jitter. This tradeoff was addressed in Examples 9.7 and 9.8, and is discussed in more detail in Chapter 11. Several acquisition aids are meant to acquire with a large loop bandwidth while performing tracking with a smaller one (see Section 9.5).

Type of timing-recovery scheme Decision-directed schemes often work well in the tracking mode of operation but tend to do a rather poor job at acquisition because of decision errors. Non-data-aided schemes, on the other hand, do not rely on decisions, and acquisition tends to be more predictable.

Example 9.13 Let us consider a Class IV $(1 - \mathcal{D}^2)$ partial-response system with uncorrelated binary data, white noise and no excess bandwidth at an SNR of 20 dB. Fig. 9.15 depicts corresponding phase acquisition curves for the non-data-aided timing-recovery scheme of Section 10.3.9 and the decision-directed version of the zero-forcing scheme of Example 10.13. In the latter case decisions are produced by a bit-by-bit feedback detector. Both schemes are dimensioned as a first-order delay-tracking loop for a steady-state phase jitter of 1% RMS. There are no frequency errors, and acquisition starts from a worst-case initial phase error of $T/2$ seconds. For the zero-forcing scheme, acquisition characteristics in the face of initial frequency errors may be found in Section 11.7 (Fig. 11.25).

Both schemes suffer from hangup. The non-data-aided scheme is not hampered by decision errors, and the hangup region is typically left within a few hundred symbol intervals. From that point on the sampling phase moves steadily towards a proper value. The decision-directed scheme, on the other hand, produces no meaningful control information when there are many bit errors, i.e. for sampling-phase errors in the vicinity of $T/2$ seconds. As a result, hangup continues until the sampling phase happens to drift into the zone where decisions are predominantly

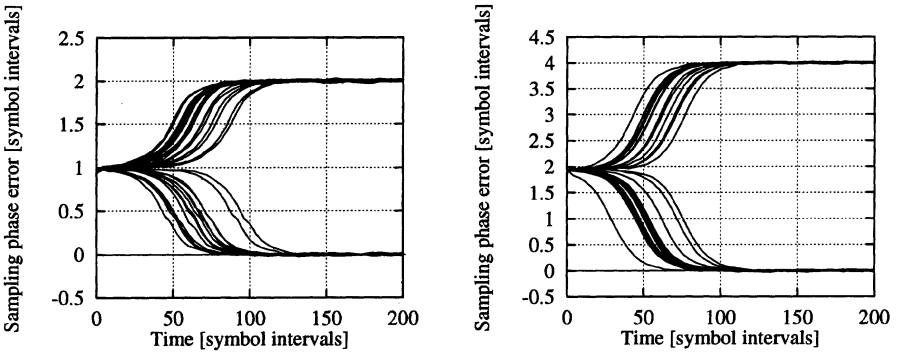


Non-data-aided scheme of Section 10.3.9 Decision-directed scheme of Example 10.13

Figure 9.15: Phase acquisition curves of open- and closed-loop timing-recovery schemes for a Class IV partial-response system with uncorrelated binary data, white noise, no excess bandwidth, and SNR = 20 dB.

correct. Due to its high efficiency, the scheme has a comparatively large loop bandwidth for a given steady-state jitter level. This causes the loop to lock up rapidly once the sampling phase leaves the 'danger zone'. Because of this 'overdrive', lock time is, upon average, still smaller than for the closed-loop scheme.

Decision errors can be avoided by basing acquisition on a known preamble. This can improve acquisition speeds dramatically. Of particular interest are periodic preambles.



Non-data-aided scheme of Sections 10.3.9 and 10.3.11 Data-aided version of scheme of Example 10.13

Figure 9.16: As Fig. 9.15, but with data pattern '$\ldots + + - - + + - - \ldots$' of period $4T$.

Example 9.14 Let us reconsider the Class IV partial-response system of Example 9.13. The binary pattern '$\ldots + + - - + + - - + + \ldots$' of period $4T$ is well suited for Class IV because its power is concentrated in the middle of the pass-band. The received signal is a sine

wave of frequency $1/(4T)$ embedded in noise. The non-data-aided scheme of Section 10.3.9 can be amended to this type of signal along the lines of Section 10.3.11. The VCO frequency is halved and the transfer characteristic of the timing-error detector becomes triangular with period $2T$. For this reason the worst-case initial phase error increases to one symbol interval T. Even so, acquisition is much faster than with random data for the same amount of steady-state jitter (Fig. 9.16). A similar remark applies to the closed-loop scheme of Example 9.13 when decisions are replaced by the known data symbols. In both cases acquisition speeds remain seriously limited by hangup. Improvements in this respect are possible, for example, by introducing hysteresis as in [54], [14] and [36], or along the lines of Example 9.18.

Type of data sequence The data pattern on which acquisition is based has a strong impact on acquisition behavior. Of particular importance are the spectral content and the degree of randomness of the pattern. Random data generally yields slower and less predictable acquisition behavior than periodic data (compare Examples 9.13 and 9.14; see also Section 11.7). This explains why a periodic *clock run-in pattern* (*training sequence*, *clock run-in sequence*) is often used as part of the data preamble in data modems, digital recording systems, and a variety of other digital communication systems. This pattern is normally chosen to have a frequency high in the passband of the channel so as to convey comparatively much timing information. The corresponding received signal is (or approximates) a sine wave embedded in noise. This is a classical type of input signal for phase-locked loops, and an assortment of techniques is available for phase and frequency detection (see, for example, [11], [12], [21], [30], [37], [38], [40], [44] and references therein). A sample of these approaches is described in Appendix 9B. The techniques of Appendix 9B are all based on coherent demodulation of the received signal, and can be tailored for fast acquisition without hangup. For the sake of completeness we mention that the timing-recovery techniques of Chapter 10, while primarily meant for random data, are generally also applicable to periodic patterns.

9.5 AIDED ACQUISITION

Acquisition aids are of interest whenever the 'bare' acquisition properties as inventorized above are not good enough. A small assortment of aids is described below. The resulting acquisition speeds are all subject to the fundamental limits of Section 9.6.

Frequency sweep The VCO can be swept at a suitable rate in order to search for the signal. The sweep rate must not be too large in order to avoid that the desired frequency is 'missed'. We refer to [21] and [44] for a discussion of permissible sweep rates and the degree of acquisition speed-up that is to be expected. Once lock is acquired, the presence of the sweep voltage manifests itself in the form of a steady-state phase offset that degrades bit error rates. To avoid this *loop stress* the sweep voltage may be turned off under the control of a *lock indicator* (see Fig. 10.11 and Problem 10.22 for examples of lock indicators).

Frequency discriminator A *frequency discriminator* (also called *frequency detector*) can be used to adjust the VCO in order to acquire rapidly [23, 43]. Fig. 9.17 shows a typical arrangement.

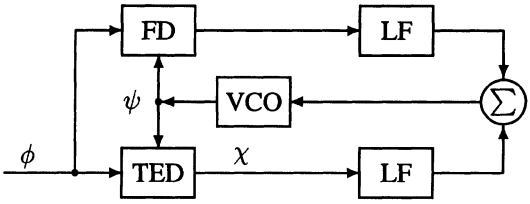


Figure 9.17: Discriminator-aided acquisition.

If the initial frequency difference is large, the frequency discriminator (FD) pulls the VCO toward lock. In this stage the loop functions effectively as a frequency-locked loop (FLL). When the difference becomes sufficiently small, the timing-error detector (TED) takes over and locks the loop. It is, in principle, advantageous to use different loop filters for FD and TED [43]. Depending on these filters, a loop stress may arise that may force the FD to be switched off after acquisition. This is best done under the control of a lock indicator.

Frequency detection may be based on a variety of features of the incoming signal. For example, in systems with a run-length-limited modulation code, violations of the run-length constraints can serve as an indication of large frequency errors [59]. In situations where the spectrum of the received signal is well defined, the center of gravity of that spectrum may serve as a frequency reference. This approach is at the heart of the *quadricorrelator* [49, 23, 3], which is derived and analyzed in Appendix 9B.3. Other methods of frequency detection are described in [43], [23], and references therein.

Rapid and accurate frequency detection is possible when the received signal is a sinusoid embedded in noise, but tends to be much more difficult for a random data signal (see Example 9.19).

Use of frequency references This approach is best outlined with an example.

Example 9.15 In hard disk drives, data is written under the control of a crystal-based frequency synthesizer that determines the nominal data rate. The angular velocity of the disk is controlled by a motor control circuit such that read and write data rates are within a fraction of a percent. Detection instants are determined by an integrated read VCO whose free-running frequency is not accurately known. To overcome this problem, the VCO may be locked to the frequency synthesizer whenever no data is being read. Once reading starts, VCO control is taken over by the timing-recovery loop. Acquisition can be fast because only a small residual frequency error needs to be dealt with.

In this example, a motor control circuit and a frequency synthesizer together serve to minimize initial frequency offsets upon reception. In cable transmission systems, the use of crystal-controlled VCOs at both ends of the system essentially has the same purpose.

Hysteresis Hangup may occur whenever the timing-error detector has no response at one or more non-desired sampling phases. It is sometimes possible to add hysteresis to the TED in order to prevent it from ever dwelling near such a spot. Cases in point include the technique of [54] for random data with full-response signaling, and the approaches of [14] and [36], which are meant for a $4T$ preamble $(\ldots++--++--\ldots)$ in conjunction with Class IV $(1 - \mathcal{D}^2)$ reception. Another example is the phase-and-frequency detector (PFD) of Section 10.3.7, which is sometimes used in conjunction with run-length-limited data. The transfer characteristic of this PFD is shown in Fig. 9.18.

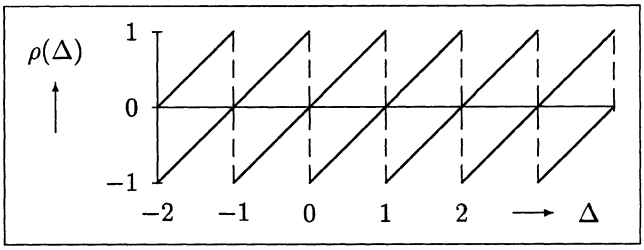


Figure 9.18: Transfer characteristic of the phase-and-frequency detector (PFD) of Section 10.3.7.

The characteristic is made up of parallel and partly overhanging linear segments. At each point in time, one of these segments is active and uniquely determines the PFD output as a function of the input. The PFD output follows changes of the input until the end of a segment is reached, and then jumps to the nearest segment. At that point the PFD output is close to zero, the phase is close to a desired one,[4] and the loop is likely to lock up at this desired phase. Because of the hysteresis, there are no longer any unstable equilibria where the loop might hang up for an ill-predictable period.

Exercise 9.3 Assume that there is a frequency error between the incoming data and the VCO. Argue that the average output of the PFD is proportional to this error. Is this an advantage?

Zero-phase restart Here the clock is momentarily stopped and then restarted in accurate phase-alignment with an event in the replay signal, such as a transition or the occurrence of a peak. The aim is to provide the VCO with an initial sampling-phase at a safe distance from the 'hangup zone'. Zero-phase restart is widely applied in hard disk drives (see also Section 10.3.10).

Variable loop bandwidth Enlargement of loop bandwidths helps to expedite acquisition. This bandwidth can be controlled via both timing-error detector and loop filter. Approaches include:

[4]We recall that proper phase errors for data detection are $\Delta = n$ for any integer n.

Programmed gain sequence: Here loop gain is set at a large initial value, and is lowered in one or more steps at a predefined pace. This pace must be low enough to ensure acquisition even with worst-case initial phase and frequency offsets. For this reason, best results are to be expected when acquisition behavior is highly predictable.

Adaptive loop filter: The loop filter may be adapted in response to the instantaneous estimates of phase and frequency errors (see, for example, [26, 29]).

TED nonlinearity: A memoryless nonlinearity can be added to the TED in order to increase TED gain whenever phase errors are large [27], [44, sec. 5.2.3]. This will accelerate acquisition provided that large phase errors can be detected reliably. While this is true for many non-data-aided timing-recovery schemes, it is normally untrue for data-aided schemes in the decision-directed mode of operation. In such schemes, large phase errors inevitably yield decision errors that cause the useful portion of the TED output to be overwhelmed by disturbance components. Extra amplification of these disturbances is clearly to no avail.

Example 9.18 illustrates the effect of a nonlinearity in a coherent tracking loop for sinusoidal input signals. Further examples may be found in [27] and [44, Section 5.2.3].

9.6 ACQUISITION LIMITS

Even with aided acquisition it inevitably takes some time to determine phase and frequency to a given degree of accuracy. The main culprit is additive noise, whose effect can only be lowered to prescribed proportions by averaging across a certain time interval. We now elaborate this effect to greater detail. Section 9.6.1 is concerned with data-aided maximum-likelihood estimation of a *fixed* delay ϕT based on a transmitted data sequence $\underline{a}_N$ of only N symbol intervals. We will find that the accuracy with which ϕ can be estimated depends on three factors, viz. the message length N, the channel signal-to-noise ratio SNR, and an efficiency factor $\gamma(\underline{a}_N)$ that is closely related to the efficiency γ of (9.3). Section 9.6.2 provides a similar analysis of frequency acquisition. Frequency acquisition is often based on sinusoidal training patterns, and the analysis in Section 9.6.2 is restricted to maximum-likelihood frequency estimation of a sinusoid embedded in noise.

9.6.1 Data-aided maximum-likelihood phase acquisition

In Fig. 9.19, given data preamble $\underline{a}_N \triangleq [a_0, \ldots, a_{N-1}]$ of N symbols is transmitted at a data rate $1/T$ across a channel with finite impulse response $h(t)$, additive Gaussian noise $n(t)$ with zero mean, and a fixed but *a priori* unknown delay ϕT.

The preamble $\underline{a}_N$ is known to the receiver, and so are $h(t)$ and the power spectral density of the noise $n(t)$. The normalized delay ϕ is to be estimated. To this end, the timing-recovery subsystem of the receiver is allowed to observe $r(t)$ as long as it desires before producing a 'one-shot' estimate $\tilde{\phi}$ of ϕ. It obviously takes at least NT seconds

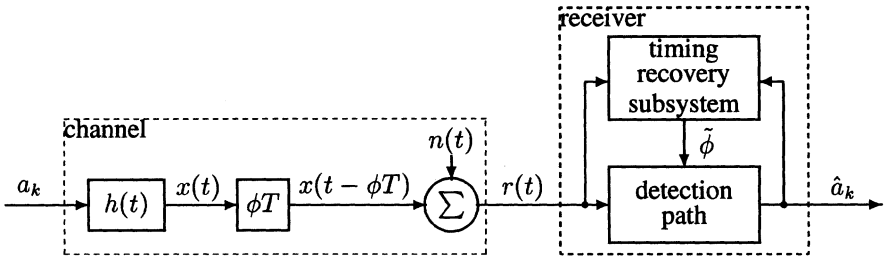


Figure 9.19: System model for maximum-likelihood data-aided timing recovery.

from the start of the preamble before $\tilde{\phi}$ becomes available, i.e. the acquisition time is lower-bounded by NT seconds. Appendix 9C is concerned with the situation that $\tilde{\phi}$ is produced in a maximum-likelihood fashion. For any given preamble $\underline{a}_N$, it is found that this approach yields an error $\Delta \triangleq \tilde{\phi} - \phi$ with mean-square value

$$\sigma_\Delta^2(\underline{a}_N) = \frac{1}{\gamma(\underline{a}_N) N \mathrm{SNR}}. \tag{9.7}$$

Here $\gamma(\underline{a}_N)$ is an efficiency factor akin to the one of (9.3) that is defined as

$$\gamma(\underline{a}_N) \triangleq \frac{1}{\mathrm{SNR}} \int_{-\infty}^{\infty} (2\pi\Omega)^2 \frac{|A(e^{j2\pi\Omega})|^2}{N} \mathrm{SN}(\Omega) d\Omega \tag{9.8}$$

and $A(e^{j2\pi\Omega})$ denotes the discrete-time Fourier transform of a_k, i.e.

$$A(e^{j2\pi\Omega}) \triangleq \sum_{k=0}^{N-1} a_k e^{-j2\pi\Omega k}. \tag{9.9}$$

Fig. 9.20 visualizes the dependence of the RMS error σ_Δ on SNR, N and $\gamma(\underline{a}_N)$.

An increment of SNR by 6 dB halves σ_Δ. The same effect is achieved by increasing N or γ by a factor of 4. Efficiency factors $\gamma(\underline{a}_N)$ depend rather strongly on the bandwidth of the channel as well as on the type of data a_k. Expression (9.8) suggests that for optimum efficiency, $\underline{a}_N$ should have its energy concentrated in portions of the spectrum where $\Omega^2 \mathrm{SN}(\Omega)$ is largest. A sinusoidal preamble is likely to be a good choice, provided that its frequency is selected properly. For periodic preambles it is easy to see that $|A(e^{j2\pi\Omega})|^2$ grows essentially linearly with N for sufficiently large N. This causes $\gamma(\underline{a}_N)$ to become independent of N. The limiting value is the efficiency γ_{ML} for maximum-likelihood data-aided timing recovery that we will encounter in Chapter 10 (equation (10.15)). The relation between $\gamma(\underline{a}_N)$ and γ_{ML} is elaborated to greater detail in Problem 9.8.

Example 9.16 Let us consider the $4T$ pattern $\underline{a}_N = \ldots + + - - + + - - \ldots$ of period $L = 4$. Fig. 9.21.a depicts the corresponding amplitude spectra $|A(e^{j2\pi\Omega})|^2/N$ for various

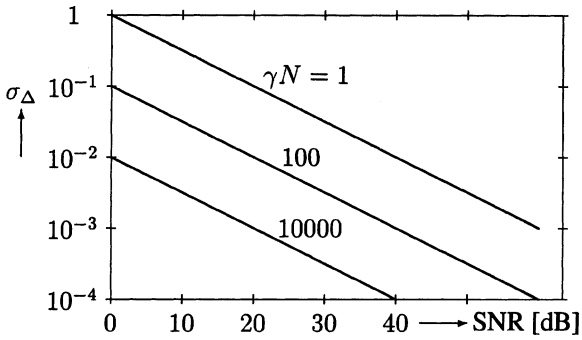
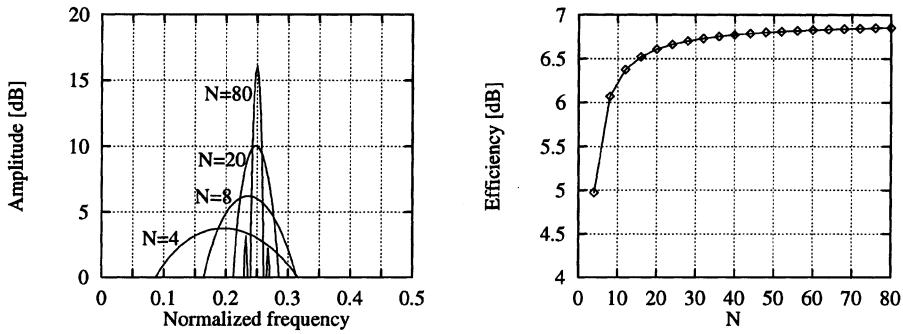


Figure 9.20: RMS jitter variance σ_Δ (in symbol intervals) versus signal-to-noise ratio SNR for various values of $\gamma(\underline{a}_N)N$.



a. Normalized amplitude $|A(\Omega)|^2/N$ versus normalized frequency Ω

b. Efficiency factors $\gamma(\underline{a}_N)$ for a Class IV $(1-\mathcal{D}^2)$ channel with white noise and no excess bandwidth

Figure 9.21: Characteristics of binary $4T$ patterns of length N.

values of N. As N grows, $|A(e^{j2\pi\Omega})|^2/N$ tends towards a line spectrum with lines at the fundamental frequency $\Omega = 0.25$ and its harmonic repetitions. The pattern under consideration is often applied as a preamble with partial-response Class IV $(1-\mathcal{D}^2)$ signaling. Corresponding efficiency factors $\gamma(\underline{a}_N)$ are depicted in Fig. 9.21.b for the situation that noise is white and that excess bandwidth is absent. Efficiencies increase monotonically with N and rapidly approach their limiting value $\pi^2/2 \simeq 4.93$ (6.93 dB). For this value and an SNR of 20 dB we find with (9.7) that at least 20 symbol intervals will be needed to estimate the phase ϕ to within 1% RMS. The coherent schemes of Example 9.18 approach this limit within a factor of about 3.

Exercise 9.4 Assume that the received signal consists of a continuous-time sinusoid of normalized frequency Ω_0 and power P_s embedded in white noise of power spectral density N_0, and that N symbol intervals T of this received signal are used for phase acquisition. Prove that $\sigma_\Delta^2 \simeq 1/[N(2\pi\Omega_0)^2\text{SNR}_i]$ for sufficiently large N where $\text{SNR}_i \triangleq P_sT/N_0$.

9.6.2 Frequency acquisition

In this section we are concerned with the problem of estimating the frequency ω_0 of a sinusoidal signal $s(t) = \sqrt{2}A\sin(\omega_0 t + \theta_0)$ with RMS magnitude A embedded in additive Gaussian white noise $n(t)$ of power spectral density N_0. The phase θ_0 of $s(t)$ is assumed to be unknown and is taken to be a random variable that is uniformly distributed in $[0, 2\pi]$. The frequency estimator is desired to produce a maximum-likelihood estimate $\tilde{\omega}_0$ of ω_0 based on observation of the received signal $r(t) = s(t) + n(t)$ during a finite interval $t \in [0, T_o]$. Intuitively, a reasonable approach for estimating ω_0 would be to apply $r(t)$ to a spectrum analyzer and to pick the frequency for which the spectrum has largest magnitude. In Appendix 9D it is shown that this approach is, in fact, optimal when based on the energy spectrum $S_r(\omega)$ of $r(t)$ across the observation interval. This spectrum may be determined via a coherent demodulator (see Appendix 9D), and is the Fourier transform of the autocorrelation function

$$R(\tau) \triangleq \int_0^{T_o - |\tau|} r(t)r(t+\tau)dt \tag{9.10}$$

of $r(t)$. Since $R(\tau)$ is a symmetric function of τ that vanishes for $|\tau| > T_o$ we may write $S_r(\omega)$ as

$$S_r(\omega) = 2\int_0^{T_o} R_r(\tau)\cos\omega\tau d\tau. \tag{9.11}$$

Normally $A^2T_o \gg 2N_0$. An approximative derivation [64, 32] for this situation reveals that the estimate $\tilde{\omega}_0$ is unbiased ($E[\tilde{\omega}_0] = \omega_0$), and that the normalized error $(\tilde{\omega}_0 - \omega_0)/\omega_0$ has variance

$$\lambda^2 \triangleq E[(\frac{\tilde{\omega}_0}{\omega_0} - 1)^2] = \frac{12N_0}{A^2\omega_0^2 T_o^3}. \tag{9.12}$$

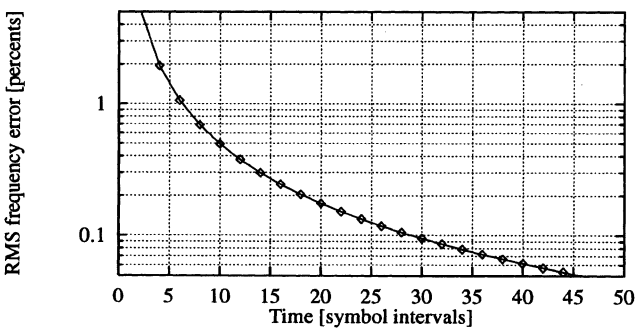


Figure 9.22: RMS frequency error for maximum-likelihood frequency estimation of a sinusoid of period $4T$ embedded in white noise at SNR = 20 dB. The estimator is allowed to observe this signal during N symbol intervals T.

Example 9.17 Let us reconsider the Class IV partial-response system of Examples 9.19 and 9.16 in conjunction with the data pattern '$\ldots + + - - + + - - \ldots$' of period $4T$. The received signal is a sinusoid $s(t) = 2\sqrt{2}\sin\frac{\pi t}{2T}$ with $A = 2$ and $\omega_0 = \frac{\pi}{2T}$. The channel has additive white noise whose power spectral density N_0 is related to the channel signal-to-noise ratio SNR as $N_0 = 2T/\text{SNR}$. Just as in Example 9.16 we consider an observation interval that spans N symbol intervals, i.e. $T_o = NT$. Then the output of the ML estimator is unbiased provided that $N\text{SNR} \gg 1$. Even for small values of N this condition is almost certainly met. The normalized error has variance $\lambda^2 = 24/(\pi^2 N^3 \text{SNR})$. The RMS error λ is depicted as a function of N in Fig. 9.22. Some 10 symbol intervals are needed to lower the RMS frequency error to 0.5%. The frequency-locked loop of Example 9.19 approaches this limit within a factor of about 3.

9.7 FURTHER READING

It appears that the Russian A.A. Pistol'kors was the first to suggest the extraction of a synchronization reference from the information bearing signal [50]. Self-timing was first used on an industrial scale in digital magnetic recording systems [20, 31] and in regenerative repeaters. Classical papers on timing recovery are mostly written against the latter background [60, 8, 1, 53]. Timing recovery has long been perceived as a field in which 'golden fingers' and intimate familiarity with analog circuitry are crucial. This is evidenced by a myriad of timing recovery schemes in which clever implementation tricks tend to obscure the underlying ideas. It is perhaps for this reason that tutorial reviews are rare and restricted in scope (partial exceptions are [19], [35, Chapter 15], and [44]). The present chapter has set the stage for a comparatively extensive review. Chapter 10 presents a catalog of timing recovery schemes, with emphasis on how timing information is extracted from the incoming signal. The common trait of these schemes is that they all have the form of a phase-locked loop. This trait is studied in Chapter 11. It is worth remarking that PLLs are also widely applied outside the realm of timing recovery. Their properties are well understood, as witnessed by a variety of review articles [28, 30, 19], books [21, 62, 58, 37, 12, 11, 44] and collections of benchmark papers [38, 40]. Our treatment in Chapter 11 serves as an introduction to this body of work, and provides some extensions that are specific to PLLs for timing recovery.

APPENDIX 9A

All-digital timing-recovery scheme of Fig. 9.7

In the system of Fig. 9.7, the analog-to-digital converter delivers samples $z_m = z(mT_x)$ at a fixed sampling rate $1/T_x$ that is unrelated to (but higher than) the symbol rate $1/T$. The filter before the ADC prevents aliasing and therefore $z(t)$ can, in principle, be reconstructed from z_m without error. The prefilter has discrete-time impulse response w_m and the output $x_m = (z * w)_m$ is a sampled version $x(mT_x)$ of a fictitious continuous-time signal $x(t) = \sum_{m=-\infty}^{\infty} w_m z(t - mT_x)$. The interpolator serves to resample $x(t)$ at instants t_k. Conceptually, this is possible in the manner shown in Fig. 9A.1.

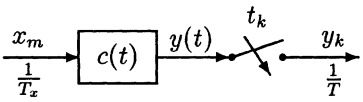


Figure 9A.1: Sampling-rate converter with continuous-time reconstruction filter.

A linear pulse modulator with symbol response $c(t)$ converts x_m into a continuous-time signal $y(t)$ that is desired to approximate $x(t)$ as well as possible. A perfect approximation requires that $c(t) = \sin(\pi t/T_x)/(\pi t/T_x)$, i.e. that $c(t)$ characterizes an ideal reconstruction filter. Such a filter is not implementable and therefore $y(t)$ cannot equal $x(t)$ precisely. A sampler which closes at the instants t_k yields a sequence $y_k = y(t_k) \simeq x(t_k)$.

The configuration of Fig. 9A.1 can be transformed into an equivalent discrete-time configuration that is well-suited for a digital implementation. Following [24] we express t_k in terms of a *basepoint index* $m_k \triangleq \lfloor t_k/T_x \rfloor$ and a *fractional interval* $\mu_k \triangleq t_k/T_x - m_k$ according to $t_k = (m_k + \mu_k)T_x$. Clearly $0 \leq \mu_k < 1$; both m_k and μ_k are fully determined by t_k. It is not difficult to recast y_k in terms of m_k and μ_k. The result is

$$y_k = \sum_{i=-\infty}^{\infty} x_{m_k - i} c((i + \mu_k)T_x). \tag{9A.1}$$

This may be written more compactly as a discrete-time convolution:

$$y_k = (x * c^{\mu_k})_{m_k} \tag{9A.2}$$

where $c_m^{\mu_k} \triangleq c((m + \mu_k)T_x)$ is a discrete-time impulse response that is fully determined by μ_k. In practice one might subdivide the continuum of possible phases μ_k into a sufficiently large number P of sub-intervals and store a fixed impulse response c_m^p ($p \in \{0, \ldots, P-1\}$) for each interval. The interpolator can then be constructed with a single discrete-time filter whose impulse response c_k^p is renewed at every instant t_k.

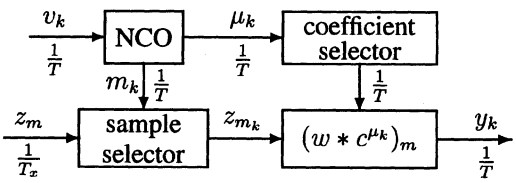


Figure 9A.2: Discrete-time NCO/prefilter/interpolator combination.

In many instances it will be attractive from an implementation standpoint to combine the discrete-time interpolator filter with the prefilter. This leads to the configuration of Fig. 9A.2. At every sampling instant t_k, the NCO produces a fractional

interval μ_k which determines the coefficients $(w * c^{\mu_k})_m$ that are to be loaded. For reasons of causality and implementability we may assume that $(w * c^{\mu_k})_m = 0$ for $m \notin \{0, \ldots, M\}$ where M denotes the memory length of the filter. The baseline index m_k which is also produced by the NCO serves to select the samples $z_{m_k}, \ldots, z_{m_k - M}$ from the incoming sequence z_m for the filter to act upon. The result is the desired interpolant

$$\sum_{i=0}^{M} z_{m_k - i}(w * c^{\mu_k})_i = (z * w * c^{\mu_k})_{m_k} = (x * c^{\mu_k})_{m_k} = y_k. \tag{9A.3}$$

It remains to describe the NCO in more detail and to determine the coefficients $(w*c)^{\mu}_m$. This is done in the following two sections.

9A.1 Numerically controlled oscillator (NCO)

The instantaneous frequency $1/T_k$ of the NCO is just the inverse of the kth sampling interval $T_k \triangleq t_k - t_{k-1}$. This interval is controlled by the NCO input v_k according to

$$\frac{T_k}{T} = 1 + \lambda + K_o v_k \tag{9A.4}$$

where K_o is the NCO sensitivity and λ accounts for a possible error between the free-running NCO frequency and the symbol rate $1/T$. In the usual event that $|\lambda + K_o v_k| \ll 1$ we may write the instantaneous NCO frequency as $1/T_k \simeq (1 - \lambda - K_o v_k)/T$. Hence $1/T_k$ depends linearly on the control voltage v_k and λ may be regarded as the frequency error normalized with respect to the symbol rate $1/T$ (i.e. $\lambda = 0.01$ corresponds to an error of 1%). Upon realizing that $t_k/T_x = m_k + \mu_k$ we find with (9A.4) that

$$(m_k - m_{k-1}) + (\mu_k - \mu_{k-1}) = \frac{T_k}{T_x} = \frac{T}{T_x}(1 + \lambda + K_o v_k). \tag{9A.5}$$

Based on this recursion we can determine separate recursions for m_k and μ_k. We recall that m_k is an integer and that $0 \leq \mu_k < 1$. Correspondingly

$$m_k = m_{k-1} + \lfloor \frac{T_k}{T_x} + \mu_{k-1} \rfloor \tag{9A.6}$$

and

$$\mu_k = (\mu_{k-1} + \frac{T_k}{T_x}) \bmod 1. \tag{9A.7}$$

A problem with these recursions is that λ is, in general, not exactly known and this uncertainty extends via (9A.4) to T_k. It will, therefore, be necessary to update m_k and μ_k based on an estimate of λ, such as $\lambda = 0$.

Exercise 9.5 Assume that $\lambda = 0$. Recast (9A.6) and (9A.7) in terms of the NCO control voltage v_k.

The NCO updates of (9A.6) and (9A.7) are conveniently implemented with the aid of a modulo counter with P positions. At every instant t_k, the counter is advanced in proportion to $(1 + K_o v_k)$. Each counter overflow increases m_k by one, and the updated counter position is a quantized measure of μ_k which directly determines the filter coefficients $(w * c^{\mu_k})_m$ that are to be selected. With minor extensions to this counter it is possible to derive an additional control signal that can be used to re-time the detected data $\hat{a}_k$ in order to allow jitter-free re-transmission [24]. In cases where data is meant for 'local consumption' $\hat{a}_k$ might, for example, be collected in an elastic buffer for re-synchronization to the master clock that controls the remainder of the system.

9A.2 Design of interpolator filters

Linear interpolation We first consider the situation without prefilter. In many instances the sampling rate $1/T_x$ is many times higher than the highest frequency component of $z(t)$. Then $z(t)$ will change only gradually between the sampling instants $m_k T_x$ and $(m_k + 1)T_x$ (Fig. 9A.3).

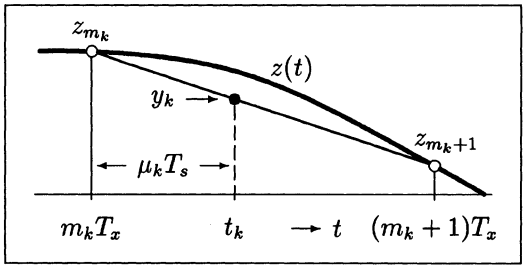


Figure 9A.3: Linear interpolation will yield a reasonable approximation of $z(t_k)$ if $z(t)$ changes only slowly between the sampling instants $m_k T_x$ and $(m_k + 1)T_x$.

We recall that $t_k = (m_k + \mu_k)T_x$. As μ_k varies between 0 and 1, t_k varies between $m_k T_x$ and $(m_k + 1)T_x$. At the same time $z(t)$ varies gradually between $z(m_k T_x) = z_{m_k}$ and $z((m_k + 1)T_x) = z_{m_k+1}$. A first-order approximation y_k of $z(t_k)$ may be obtained by linear interpolation between the available samples $z(m_k T_x) = z_{m_k}$ and $z((m_k + 1)T_x) = z_{m_k+1}$, i.e. $y_k = (1 - \mu_k)z_{m_k} + \mu_k z_{m_k+1}$. This corresponds to a two-tap interpolation filter with impulse response $c_m^\mu = (1 - \mu)\delta_m + \mu\delta_{m+1}$ where δ_m is the Kronecker delta function. This is clearly a non-causal filter and a delay of one sample interval T_x must be added to arrive at a causal implementation. A prefilter with impulse response w_m may be combined with the interpolation filter and the resulting impulse response is $(w * c^\mu)_m$.

Cubic interpolation If the oversampling factor is somewhat too low for linear interpolation then the 4-tap interpolation filter of [18] may be used. This filter has impulse

response

$$c_m^\mu \triangleq \begin{cases} 0.5(\mu^2-\mu), & m=-2, \\ 0.5(\mu-\mu^2)+\mu, & m=-1, \\ 0.5(\mu-\mu^2)+1-\mu, & m=0, \\ 0.5(\mu^2-\mu), & m=1, \\ 0, & \text{else.} \end{cases} \tag{9A.8}$$

Higher-order interpolation As the sampling rate $1/T_x$ decreases further, at some point higher-order interpolation becomes needed. Let us assume that c_m^μ has length $M = B + E + 1$ where B and $E + 1$ are the number of 'past' and 'future' samples that are used in the interpolation, i.e. $c_m^\mu = 0$ for $m \notin \{-B, E\}$. (The corresponding filter is again not causal. A pure delay of B sampling intervals T_x will overcome this problem.) The interpolator transfer function

$$C^\mu(e^{j2\pi\Omega}) \triangleq \sum_{m=-B}^{E} c_m^\mu e^{-j2\pi\Omega m} \tag{9A.9}$$

is desired to approximate as well as possible the function $G^\mu(e^{j2\pi\Omega}) = e^{-j2\pi\mu\Omega}$ that accounts for a fractional delay of μ samples. (The expression for $G(e^{j2\pi\Omega})$ is only valid in the fundamental interval $[-0.5, 0.5]$, beyond which the usual symmetry rules apply.) Following [16] we define as a measure of resemblance the weighted mean-square error

$$\varepsilon \triangleq \int_{-0.5}^{0.5} |C^\mu(e^{j2\pi\Omega}) - G^\mu(e^{j2\pi\Omega})|^2 W(e^{j2\pi\Omega}) d\Omega \tag{9A.10}$$

where $W(e^{j2\pi\Omega})$ is some non-negative real-valued weighing function with $W(e^{-j2\pi\Omega}) = W(e^{j2\pi\Omega})$. This is typically a low-pass function so that interpolation will be most accurate at low frequencies and less accurate at higher frequencies. If the power spectral density of z_m (or of x_m for the case with prefilter) is known, then that density would normally be a reasonable weighing function. In order to arrive at the proper DC-gain we add the constraint $C^\mu(0) = \sum_{m=-B}^{E} c_m^\mu = 1$.

To find the optimum coefficients c_m^μ we first recast ε in matrix notation according to $\varepsilon = R(0) - 2\underline{c}^T\underline{v} + \underline{c}^T\mathbf{M}\underline{c}$. Here

$$R(x) \triangleq \int_{-0.5}^{0.5} W(e^{j2\pi\Omega}) \cos(j2\pi\Omega x) d\Omega \tag{9A.11}$$

is a real-valued and symmetric function of x, $\underline{c}$ is a coefficient vector $[c_{-B}^\mu, \ldots, c_E^\mu]^T$ where the superscript 'T' denotes transposition, $\mathbf{M}$ is a symmetric Toeplitz matrix with components $\mathbf{M}_{i,j} \triangleq R(i-j)$ for $i, j \in \{-B, \ldots, E\}$ and $\underline{v}$ is a vector $\underline{v} \triangleq [R(-B+\mu), \ldots, R(E+\mu)]^T$. The DC-constraint may be formulated in vector notation as $\underline{c}^T\underline{1} = 1$ where $\underline{1} \triangleq [1, \ldots, 1]^T$ is the all-ones vector of length $B + E + 1$. In order to accommodate the constraint we define a Lagrangian $L \triangleq \varepsilon - 2\lambda\underline{c}^T\underline{1}$ where λ

is a Lagrange multiplier. We must require that the gradients of L with respect to $\underline{c}$ and λ are zero. This yields the conditions $\mathbf{M}\underline{c} = \underline{v} + \lambda\underline{1}$ and $\underline{c}^T\underline{1} = 1$. These conditions have a unique solution provided that $\mathbf{M}$ is invertible. The optimum Lagrange multiplier is $\hat{\lambda} = [1 - \underline{v}^T\mathbf{M}^{-1}\underline{1}]/[\underline{1}^T\mathbf{M}^{-1}\underline{1}]$. In terms of this multiplier we may express the minimum mean-square error $\varepsilon_{\min}$ and the optimum coefficient vector $\hat{\underline{c}}$ as

$$\varepsilon_{\min} = R(0) - \underline{v}^T\mathbf{M}^{-1}\underline{v} + \hat{\lambda}^2\underline{1}^T\mathbf{M}^{-1}\underline{1} \tag{9A.12}$$

and

$$\hat{\underline{c}} = \mathbf{M}^{-1}\underline{v} + \hat{\lambda}\mathbf{M}^{-1}\underline{1}. \tag{9A.13}$$

Exercise 9.6 Assume that $B = E + 1$, i.e. that interpolation is based on an equal number of 'past' and 'future' samples. Consider the error $\varepsilon_{\min}$ as a function of μ in the interval $\mu \in [0, 1]$. Argue that $\varepsilon_{\min}(\mu)$ has a global maximum for $\mu = 1/2$. How large are $\varepsilon_{\min}(0)$ and $\varepsilon_{\min}(1)$? Assume that B (and thereby E) are chosen for a given value of μ such that $\varepsilon_{\min}(\mu)$ remains below some prescribed value $\bar{\varepsilon}$. How does B depend on μ? What value of B suffices for $\mu = 0$ and $\mu = 1$?

Fractionally-spaced equalizer Consider the situation that the PLL is in lock so that $t_k = (k + \psi)T$ for some fixed sampling phase ψ. Let us assume, furthermore, that T_x is an integer fraction of T, i.e. $T_x = T/L$ for some positive integer L (we will relax this assumption later). Then $t_k = (Lk + L\psi)T_x = (m_k + \mu)T_x$ with $m_k = Lk + \lfloor L\psi \rfloor$ and $\mu = L\psi - \lfloor L\psi \rfloor$. Hence the fractional interval μ has become independent of k and the interpolator has output $y_k = (x * c^{\mu_k})_{m_k} = (x * c^{\mu})_{Lk+\lfloor L\psi \rfloor}$. This shows that y_k is the output of a decimating filter with fixed impulse response c_m^{μ} and decimation factor L (Fig. 9A.4).

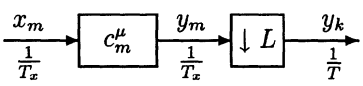


Figure 9A.4: Fractionally-spaced equalizer with a tap spacing of $T_x = T/L$ seconds.

The structure of Fig. 9A.4 is that of a fractionally-spaced equalizer with a tap spacing of T/L seconds. Appendix 6B provides a discussion of this equalizer type and shows how the equalizer coefficients c_m^{μ} may be optimized in the minimum mean-square error (MMSE) sense. By design, the MMSE equalizer combines the task of the prefilter and the interpolator of Fig. 9.7 in an optimum fashion.

In order to restrict complexity of the equalizer it is, in principle, possible to impose the constraint that only one out of every Q coefficients be nonzero, where Q is some integer smaller than L. This brings us into the realm of fractionally-spaced equalizers with a rational tap spacing of QT/L seconds. Design techniques for equalizers of this type are described, for example, in [51]. Net complexity is determined largely by the total number of nonzero coefficients. The value of Q that minimizes this number for

a given performance level depends, among other factors, on the characteristics of the channel and on the fractional delay μ.

In practice T may not be known accurately and may not even nominally be an integer multiple of T_x. Even so, one can take the 'nearest' integer oversampling factor L and design the filter coefficients c_m^μ on this basis. This approach will work better as L increases.

For the sake of brevity we have restricted attention in this section to a single interpolator structure and three interpolator design approaches. Further structures and approaches are described, for example, in [52] and [18].

APPENDIX 9B

Phase- and frequency-locked loops for a sinusoid embedded in noise

In this appendix we consider clock phase- and frequency recovery based on a sinusoidal signal $s(t)$ embedded in noise $n(t)$. A signal of this type is received, in many instances, when a periodic data preamble is applied to a bandlimited channel. We shall assume that $s(t)$ has amplitude A, initial phase θ_0 and angular frequency ω_0 according to $s(t) = A\cos\phi_0(t)$ with $\phi_0(t) \triangleq \theta_0 + \omega_0 t$. This formulation shows that $s(t)$ is a *constant-envelope signal* of the type encountered in analog frequency modulation (FM) and phase modulation (PM). Frequency- and phase-detection may, therefore, be based on standard demodulation techniques for FM and PM. In this appendix we restrict attention to coherent demodulators. At the heart of these demodulators is an in-phase/quadrature down-conversion that we discuss in Section 9B.1. Phase and frequency acquisition are discussed in Sections 9B.2 and 9B.3, respectively. Standard PLL textbooks such as [11], [12], [21], [37], [38] and [40] may be consulted for a review of other estimation and acquisition techniques.

9B.1 Down-conversion

For notational compactness we adopt complex notation in what follows. In this notation we may write the received signal $r(t)$ as

$$r(t) = s(t) + n(t) = \frac{1}{2}(e^{j\phi_0(t)} + e^{-j\phi_0(t)}) + n(t). \tag{9B.1}$$

In Fig. 9B.1, $r(t)$ is multiplied with the in-phase and quadrature outputs of a VCO. This oscillator delivers sinusoidal outputs with amplitude 2, angular frequency ω_1 and time-dependent phase $\phi_1(t) = \theta_1 + \omega_1 t$. The two multiplicands $x_i(t) = 2r(t)\cos\phi_1(t)$ and $x_q(t) = 2r(t)\sin\phi_1(t)$ together determine a complex-valued signal

$$\begin{aligned} x(t) &\triangleq x_i(t) + jx_q(t) = 2r(t)e^{j\phi_1(t)} \\ &= Ae^{j(\phi_0(t)+\phi_1(t))} + Ae^{j(\phi_0(t)-\phi_1(t))} + 2n(t)e^{j\phi_1(t)}. \end{aligned} \tag{9B.2}$$

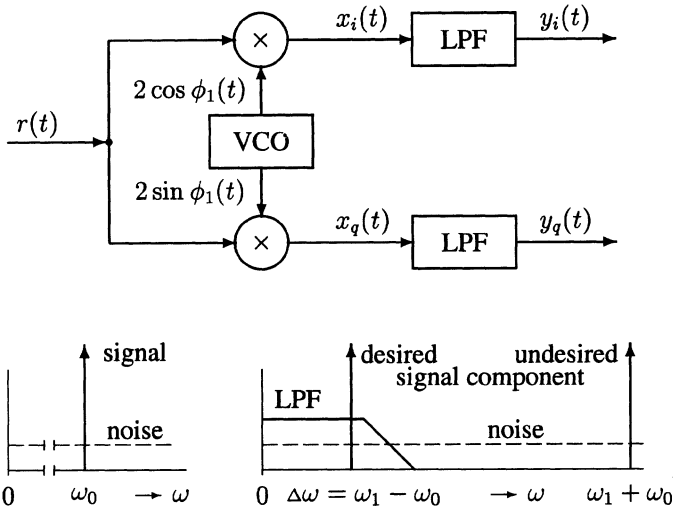


Figure 9B.1: Down-converter. Typical signal and noise spectra are shown underneath.

The term $Ae^{j(\phi_0(t)+\phi_1(t))}$ describes a phasor of frequency $\omega_0 + \omega_1$. The low-pass filters to which $x_i(t)$ and $x_q(t)$ are applied are meant to reject this frequency while passing the difference frequency $\omega_1 - \omega_0$. The result is a complex-valued filter output $y(t) = y_i(t) + jy_q(t) = A.\exp\{j(\phi_1(t) - \phi_0(t))\} + u(t)$ where $u(t)$ is the noise component of $y(t)$ and is bandlimited to the cut-off frequency of the LPFs. Except for this noise, $y(t)$ is a complex-valued phasor with constant magnitude A and phase $\Delta\phi(t) \triangleq \phi_1(t) - \phi_0(t) = \Delta\theta - \Delta\omega t$ where $\Delta\theta \triangleq \theta_1 - \theta_0$ and $\Delta\omega \triangleq \omega_1 - \omega_0$. Normally $|\Delta\omega| \ll \omega_0$. In that case the circuit of Fig. 9B.1 acts as a *down-converter* that translates an input frequency ω_0 into a much lower frequency $\Delta\omega$. We shall say that the VCO is frequency-locked to the incoming signal if $\Delta\omega = 0$ (i.e. if $\omega_1 = \omega_0$), and that it is phase-locked if $\phi_1(t) = \phi_0(t)$. The LPFs are said to be of the *planar* type because they effectively operate on a complex-valued signal $x(t)$.

Exercise 9.7 Assume that the VCO frequency ω_1 can traverse the range $(0, 2\omega_0)$. Determine the required cut-off frequency of the low-pass filters. Would it be desirable for the VCO to have a smaller frequency range than the assumed one?

9B.2 Coherent phase-locked loop

For notational convenience we will henceforth suppress the argument t whenever possible. Let us assume for a moment that noise u is negligible. Then y is a complex-valued phasor $Ae^{j\Delta\phi}$ (Fig. 9B.2). From Fig. 9B.2 we see that $\Delta\phi$ may be reconstructed from y_i and y_q according to

$$\Delta\phi = \arctan\frac{y_q}{y_i} \tag{9B.3}$$

provided that $|\Delta\phi| \leq \frac{\pi}{2}$. This range can be extended to $[-\pi, \pi]$ by taking the principal value of the argument of $y(t)$.

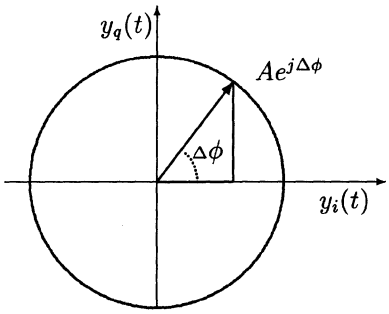


Figure 9B.2: Phasor diagram of $y(t)$ in the absence of noise $u(t)$.

Exercise 9.8 Let $\text{sgn}(x) \in \{-1, 1\}$ be the signum function of x, and let

$$\arg(y) \triangleq \frac{\pi}{2}\text{sgn}(y_q) - \arctan\frac{y_i}{y_q}. \tag{9B.4}$$

Verify that $\Delta\phi = \arg(y)$ as long $|\Delta\phi| \leq \pi$.

In practice noise u will always be present, and the above rules will yield estimates of $\Delta\phi$ rather than the exact value. It can be shown that these estimates have maximum likelihood provided that the noise component n of r is Gaussian and white and that a proper choice is made for the low-pass filters [44, Appendix 8.2.B]. Even when the argument operator of (9B.4) is used, values of $\Delta\phi$ outside $[-\pi, \pi]$ can only be estimated modulo 2π. This causes the transfer characteristic of the timing-error detector to have a sawtooth shape (Fig. 9B.3).

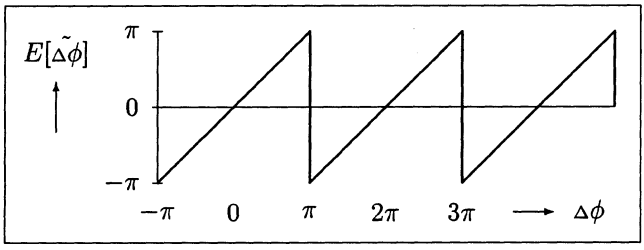


Figure 9B.3: Phase detector transfer characteristic for the situation that the phase estimate $\tilde{\Delta\phi}$ is produced via the argument operator of (9B.4).

A tracking loop that is based on the arg-operator is depicted in Fig. 9B.4. A loop filter LF connects the output of the timing-error detector with the input of the VCO. The loop is closed and will force the average TED output toward zero when stable. As

a result, the phase error $\Delta\phi$ will settle down at one of the upgoing zero-crossings of Fig. 9B.3.

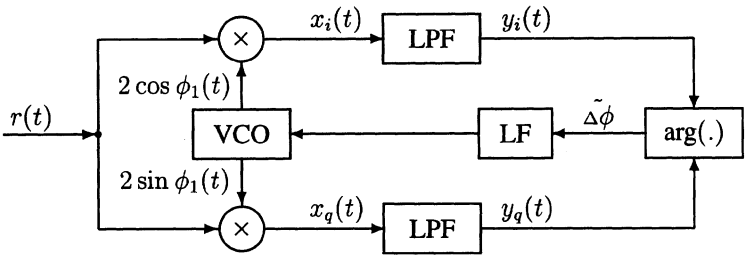


Figure 9B.4: Coherent PLL for sinusoidal input signal embedded in noise.

It should be noted that the planar low-pass filters are embedded in the loop and as such affect loop properties. A qualitative understanding of their effect may be based on the following considerations.

1. Assume that the LPFs have a bandwidth that is very much larger than that of the closed loop. In that case they will provide comparatively little noise suppression, and the output of the arg-operator may jump up and down wildly when $\Delta\phi$ is in the vicinity of any of the discontinuities in Fig. 9B.3. As a result, the output of the LF may be small even though the VCO phase ϕ_1 is nowhere near its desired value, and the loop may hang up at this erroneous phase for some time.

2. The hangup problem can be alleviated by lowering the bandwidth of the LPFs so as to suppress noise before timing-error detection. For stability reasons it is necessary that the effective delay introduced by the LPFs remains much smaller than the inverse of the closed-loop bandwidth (see Chapter 11). This determines the smallest allowable LPF bandwidth.[5]

These considerations suggest that the LPFs be taken to have minimum phase so as to achieve the smallest possible effective delay for a given noise bandwidth. To minimize hangup, the cut-off frequency should be taken as low as permissible for stability reasons. The resulting LPF bandwidth will be much larger than that of the closed loop. For this reason the LPFs will not significantly affect the noise performance of the loop.

Because of the periodicity of the characteristic of Fig. 9B.3, hangup cannot be altogether avoided (see Section 9.4). Addition of a nonlinearity behind the TED, as in Section 9.5, can alleviate the problem. Complete elimination of hangup is, in theory, possible with the aid of a technique that is called *phase unwrapping* [61, 48]. The aim of this technique is to reconstruct the absolute value of the phase rather than its principal value. In essence, this means that the discontinuities in Fig. 9B.3 must be detected

[5]It is worth noting that LPF bandwidth would not be restricted if the scheme were refashioned into an open-loop one, like that of Fig. 9.6. This would permit hangup to be virtually eliminated. Section 10.4.6 discusses a planar scheme of this type.

and repaired. This detection is, in principle, possible by virtue of the fact that the phase $\phi(t)$ is a continuous signal of restricted bandwidth. The net result of phase unwrapping is a monotonic and linear TED transfer characteristic. A characteristic of this type can, of course, at best be approximated in practice. The main merit of phase unwrapping is, therefore, that of a baseline against which practical approaches can be judged.

Example 9.18 Let us reconsider the Class IV $(1-\mathcal{D}^2)$ partial-response system of Example 9.14 in conjunction with a binary $4T$-pattern $\ldots + + - - + + - - \ldots$. The channel has no excess bandwidth, and the received signal is a sinusoid of frequency $1/(4T)$ embedded in noise at SNR = 20 dB. The tracking loop is dimensioned for a steady-state phase error of 1% RMS. The LPFs are of the second-order Butterworth type. Their cut-off frequency is set in accordance with the above considerations and amounts to 8% of the symbol frequency. Acquisition is attempted repeatedly with no initial frequency error for a first-order tracking loop, and with 15% error for a high-gain second-order loop with damping factor $\zeta = 0.707$; in the latter case the scaling factor K_l is replaced by a suitable loop filter (see Chapter 11 for a description of second-order loops). A worst-case initial phase error is used in all cases. Considered are the baseline loop of Fig. 9B.4, a version in which a nonlinearity $f(x) = x(1 + x^2/6)$ is inserted between the arg-operator and the loop filter, and a loop with phase unwrapping. Acquisition characteristics are depicted in Fig. 9B.5.

Hangup is evident in the baseline loop, is slightly less serious with added nonlinearity, and is altogether absent with phase unwrapping. Each of the characteristics in Fig. 9B.5 pertains to a restricted number of experiments. Every once in a while, hangup may last considerably longer than in these experiments. For this reason hangup is a more serious problem than suggested by the figure. Even so, it is much less serious than in Fig. 9.16. The loop with phase unwrapping lowers initial phase errors to within 1% of a symbol interval in around 50 symbol intervals T. This is within a factor of 3 from the theoretical limit as evaluated in Example 9.16. With 15% frequency error, the baseline loop slips cycles before locking up. No slips occur with nonlinearity and phase unwrapping.

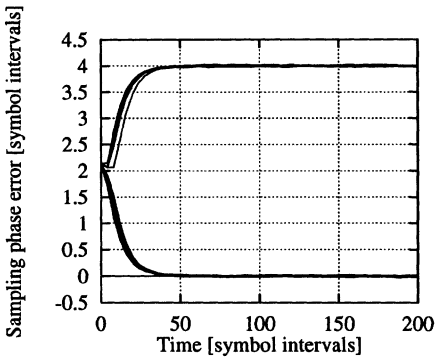
9B.3 Frequency-locked loop

Let us consider in more detail the situation that $\Delta\phi(t)$ in Fig. 9B.2 contains a frequency error $\Delta\omega$. The presence of such an error manifests itself in that the derivative of $\Delta\phi(t)$ with respect to time becomes nonzero: $d\Delta\phi(t)/dt = \Delta\omega$. With the aid of (9B.3) we find that an estimate $\Delta\tilde{\omega}$ of $\Delta\omega$ may be produced according to

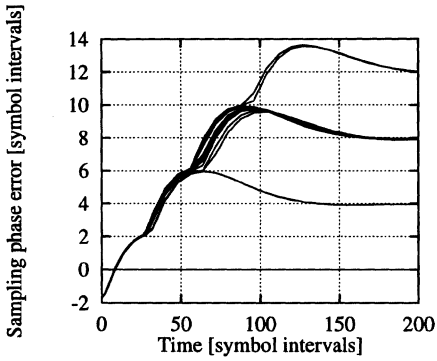
$$\Delta\tilde{\omega} = \frac{d}{dt}\arctan\frac{y_q}{y_i} = \frac{\dot{y}_q y_i - y_q \dot{y}_i}{y_i^2 + y_q^2} \tag{9B.5}$$

where the dot denotes differentiation and the dependence on t has been suppressed for notational convenience.

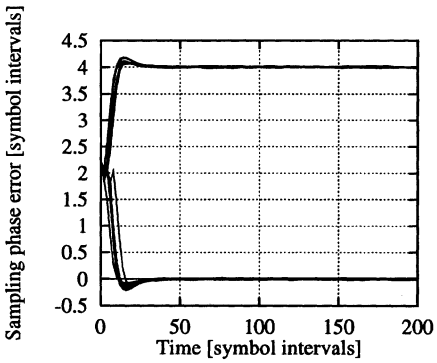
The circuit implied by (9B.5) is depicted in Fig. 9B.6. It is called *balanced quadricorrelator* and was first described by Park [49]. The division by $y_i^2 + y_q^2$ affects only the gain of the quadricorrelator and is often omitted. The circuit can be simplified further



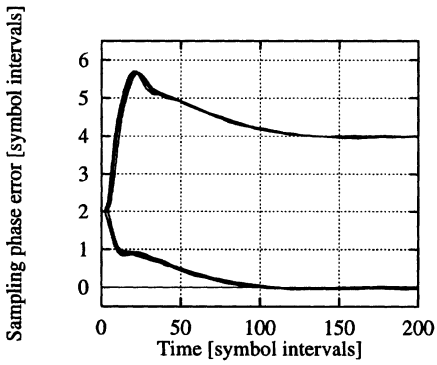
Tracking loop of Fig. 9B.4, no initial frequency offset.



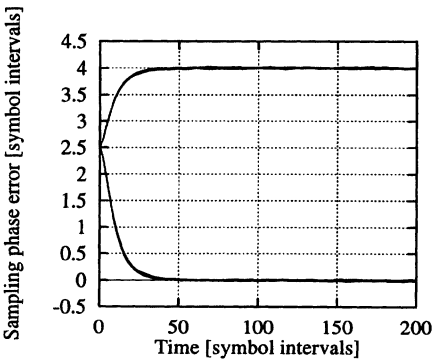
Tracking loop of Fig. 9B.4, 15% initial frequency offset.



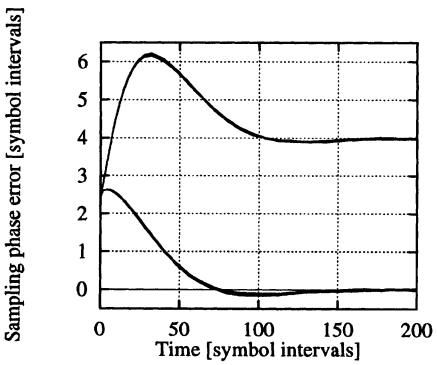
Loop of Fig. 9B.4 with added nonlinearity, no initial frequency offset.



Loop of Fig. 9B.4 with added nonlinearity, 15% initial frequency offset.



Loop with phase unwrapping, no initial frequency offset.



Loop with phase unwrapping, 15% initial frequency offset.

Figure 9B.5: Acquisition characteristics of several coherent tracking loops for a sinusoidal signal of frequency $1/(4T)$ embedded in noise at SNR = 20 dB.

by omitting the product $\dot{y}_q y_i$ at the expense of introducing a ripple component at twice the frequency error $\Delta\omega$ [23]. This leads to the non-balanced quadricorrelator, which is often just referred to as *quadricorrelator*.

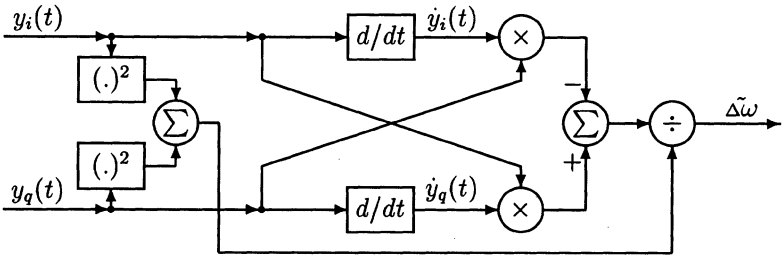


Figure 9B.6: Balanced quadricorrelator.

A first-order frequency-locked loop (FLL) that is based on the balanced quadricorrelator is depicted in Fig. 9B.7. The loop is closed via a constant gain factor K_f and an integrator. The VCO control voltage at the output of the integrator will eventually settle down at the desired value when the FLL is stable. At that point the integrator input and thereby the output of the quadricorrelator will be zero upon average. The implication is that the VCO is frequency-locked to the incoming signal. Frequency acquisition behavior of this FLL is illustrated in Example 9.19.

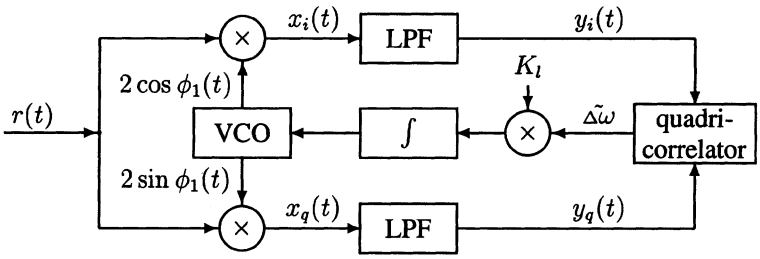


Figure 9B.7: Frequency-locked loop (FLL) based on the balanced quadricorrelator of Fig. 9B.6.

In order for the loop of Fig. 9B.7 to operate, the difference frequency between input and VCO must be within the passband of the low-pass filters. A signal that is well outside the passband will be attenuated by the filter selectivity. Bandwidth of the low-pass filter therefore provides a rough estimate of the capture range of the FLL. It can be shown that the FLL of Fig. 9B.7 remains applicable when input signals are not sinusoidal. In general, the loop acts to match the VCO frequency to the center of gravity of the input spectrum [23]. Performance may be poor when input signals are random.

Example 9.19 Consider the FLL of Fig. 9B.7 when operating on a operating on a Class IV $(1 - \mathcal{D}^2)$ partial-response signal with white additive noise. For an ideal Class IV system with uncorrelated data and no excess bandwidth, the center of gravity of $r(t)$ is located at a quarter

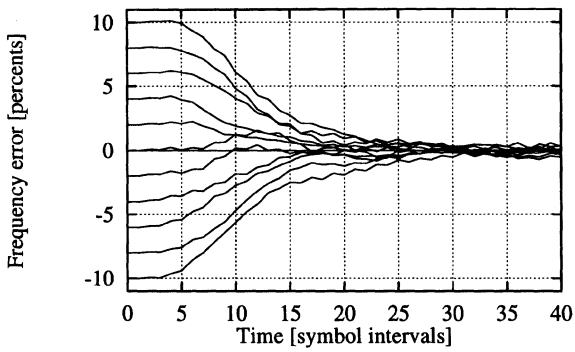


Figure 9B.8: Acquisition curves of frequency-locked loop for binary $1/(4T)$ pattern at SNR = 20 dB.

of the bit frequency $1/T$. Thus the VCO frequency will settle near $1/(4T)$, and up-conversion by a factor of 4 is needed to obtain an estimate of the symbol rate. This same factor is needed for a training pattern '$\ldots + + - - + + - - \ldots$' of period $4T$. The harmonics of this pattern are rejected by the channel, and a pure tone at $1/(4T)$ embedded in noise remains after the channel for the FLL to lock on to. Frequency acquisition curves for this situation at 20 dB SNR are depicted in Fig. 9B.8. The curves pertain to a balanced quadricorrelator with second-order Butterworth low-pass filters whose cut-off frequency amounts to 10 percent of the signaling rate $1/T$. Loop gain is set for a steady-state frequency error of around 0.5% RMS. Fig. 9B.8 illustrates that initial frequency errors between -10 and 10 percents are lowered to the steady-state level in around 30 symbol intervals. For random data this takes many orders of magnitude longer. This is clearly impractical, and for this reason the corresponding convergence curves are not shown.

Exercise 9.9 Assume that the channel is not equalized accurately so that some residual amplitude distortion is left. How will this affect the steady-state frequency at which the FLL settles down for both types of data?

APPENDIX 9C

Data-aided maximum-likelihood one-shot phase estimator

In this appendix we derive the data-aided maximum-likelihood one-shot phase estimator and analyze its performance. For the sake of brevity we refer ahead to Sections 10.2.1 and 10.2.2 for a general description of maximum-likelihood timing recovery. By retracing the steps of these sections it is easy to see for the situation at hand that maximum-likelihood estimation of ϕ is equivalent to finding the central zero-crossing

of the function

$$\dot{\Lambda}(\psi) \triangleq T \sum_{k=0}^{N-1} a_k (r * w)(t_k). \tag{9C.1}$$

Here $t_k \triangleq (k + \psi)T$ and $w(t)$ is an impulse response that is the time-derivative of that of a matched filter (see also Fig. 10.3). The expected value $E[\dot{\Lambda}(\psi)]$ represents the desired component of $\dot{\Lambda}$ and may be regarded as the transfer function $\rho(\Delta)$ of a timing-error detector. This function crosses zero for $\psi = \phi$. Superimposed on ρ is an undesired random component η that causes a displacement of the zero-crossing of $\dot{\Lambda}$ away from the ideal location $\psi = \phi$ (Fig. 9C.1).

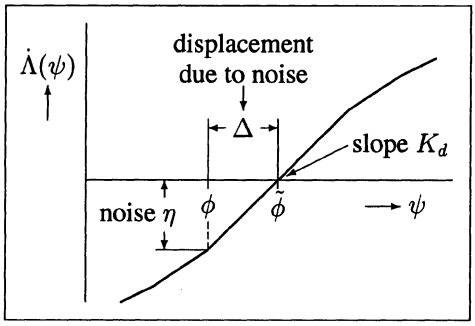


Figure 9C.1: Sample input-output characteristic of the timing-error detector according to (9C.1).

For small $|\psi - \phi|$ we may write

$$\dot{\Lambda}(\psi) \simeq K_d(\psi - \phi) + \eta, \tag{9C.2}$$

where K_d is the slope of ρ for $\psi \simeq \phi$. The zero crossing of $\dot{\Lambda}$ near $\psi = \phi$ demarcates the estimate $\tilde{\phi}$ of ϕ. The corresponding error $\Delta \triangleq \tilde{\phi} - \phi$ may be approximated by $\Delta \simeq -\eta/K_d$. This error has power

$$\sigma_\Delta^2 \simeq \frac{\sigma_\eta^2}{K_d^2}, \tag{9C.3}$$

where σ_η^2 is the variance of η. We first determine $\rho(\Delta)$ and K_d. To this end we observe from (9C.1) that

$$\rho(\psi - \phi) = T \sum_{k=0}^{N-1} E[a_k (x * w)((k + \psi)T)] \tag{9C.4}$$

where $x(t) = \sum_{l=0}^{N-1} a_l h(t-(l+\phi)T)$ is the data component of $r(t)$. Correspondingly

$$\rho(\Delta) = \sum_{k=0}^{N-1}\sum_{l=0}^{N-1} a_k a_l q_{k-l}, \tag{9C.5}$$

where $q_k \triangleq T(h * w)((k+\Delta)T)$ is a discrete-time sequence with Fourier transform

$$Q(e^{j2\pi\Omega}) = \sum_{n=-\infty}^{\infty} H(\Omega+n)W(\Omega+n)e^{j2\pi\Delta(\Omega+n)}. \tag{9C.6}$$

Translation of (9C.5) to the frequency domain yields

$$\begin{aligned}\rho(\Delta) &= \int_{-0.5}^{0.5} |A(e^{j2\pi\Omega})|^2 Q(e^{j2\pi\Omega})d\Omega \\ &= \int_{-\infty}^{\infty} |A(e^{j2\pi\Omega})|^2 H(\Omega)W(\Omega)e^{j2\pi\Delta\Omega}d\Omega,\end{aligned} \tag{9C.7}$$

where the last step stems from the periodicity of $A(e^{j2\pi\Omega})$. Since $w(t)$ is the time-derivative of the impulse response of a matched filter, the Fourier integral of $w(t)$ is

$$W(\Omega) = \frac{j2\pi\Omega}{T}\frac{H^*(\Omega)}{\mathcal{N}(\Omega)}. \tag{9C.8}$$

Correspondingly $H(\Omega)W(\Omega) = j2\pi\Omega \mathrm{SN}(\Omega)$ and

$$\rho(\Delta) = \int_{-\infty}^{\infty} j2\pi\Omega |A(e^{j2\pi\Omega})|^2 \mathrm{SN}(\Omega)e^{j2\pi\Delta\Omega}d\Omega. \tag{9C.9}$$

It follows that

$$K_d = \frac{\partial\rho(\Delta)}{\partial\Delta}|_{\Delta=0} = \int_{-\infty}^{\infty} (2\pi\Omega)^2 |A(e^{j2\pi\Omega})|^2 \mathrm{SN}(\Omega)d\Omega. \tag{9C.10}$$

As to σ_η^2, we note that $\eta = \sum_{k=0}^{N-1} a_k z_k$, where $z_k \triangleq T(n * w)((k+\psi)T)$ is a stationary noise sequence with autocorrelation function $d_k \triangleq E[z_{k+l}z_k]$. Thus

$$\sigma_\eta^2 = \sum_{k=0}^{N-1}\sum_{l=0}^{N-1} a_k a_l E[z_k z_l] = \sum_{k=0}^{N-1}\sum_{l=0}^{N-1} a_k a_l d_{k-l}. \tag{9C.11}$$

In frequency-domain notation this expression becomes

$$\sigma_\eta^2 = \int_{-0.5}^{0.5} |A(e^{j2\pi\Omega})|^2 D(e^{j2\pi\Omega})d\Omega \tag{9C.12}$$

where $D(e^{j2\pi\Omega})$ denotes the discrete-time Fourier transform of d_k. To find $D(e^{j2\pi\Omega})$, we note that d_k is a sampled version of the autocorrelation function $v(t) \triangleq T^2E[(n * w)(t+\tau)(n*w)(\tau)]$ of $T(n*w)(t)$ according to $d_k = v(kT)$. Spectrally, this sampling amounts to a folding operation according to

$$D(e^{j2\pi\Omega}) = \frac{1}{T}\sum_{n=-\infty}^{\infty} P_v(\Omega + n), \tag{9C.13}$$

where $P_v(\Omega)$ denotes the Fourier integral of $v(t)$, i.e. the PSD of $T(n * w)(t)$. This PSD amounts to $P_v(\Omega) = T^2\mathcal{N}(\Omega)|W(\Omega)|^2 = T(2\pi\Omega)^2\mathrm{SN}(\Omega)$. Upon combining this expression with (9C.13) and (9C.12) we find after some elementary rearrangements that

$$\sigma_\eta^2 = \int_{-\infty}^{\infty} (2\pi\Omega)^2|A(e^{j2\pi\Omega})|^2\mathrm{SN}(\Omega)d\Omega. \tag{9C.14}$$

Together with (9C.3) (9C.10) it follows that

$$\sigma_\Delta^2 \simeq \frac{1}{\int_{-\infty}^{\infty}(2\pi\Omega)^2|A(e^{j2\pi\Omega})|^2\mathrm{SN}(\Omega)d\Omega}. \tag{9C.15}$$

This validates (9.7) and (9.8). □

APPENDIX 9D

One-shot maximum-likelihood frequency estimator

In this appendix we derive the one-shot maximum-likelihood estimator of the frequency of a sinusoid $s(t) = \sqrt{2}A\sin(\omega_0 t+\theta_0)$ embedded in additive Gaussian white noise $n(t)$ of power spectral density N_0. The frequency ω_0 and phase θ_0 are *a priori* unknown and are assumed to be constant. The estimator is desired to base its estimate of ω_0 on observation of $r(t) = s(t) + n(t)$ during an observation interval $t \in [0, T_o]$ of duration T_o.

Would the phase θ_0 be known, then for every allowable frequency ω_c we could compute a candidate signal $s_c(\omega_c, t) \triangleq \sqrt{2}A\sin(\omega_c t + \theta_o)$. We could then select that candidate frequency for which $s_c(\omega_c, t)$ best resembles $r(t)$ in the mean-square sense. This choice would have maximum likelihood because noise is Gaussian and white (cf. Appendix 3C). Unfortunately, θ_0 is not known and therefore maximum-likelihood estimation of ω_0 becomes somewhat more involved. Specifically, for every allowable phase θ_c we must determine a *conditional likelihood function* $p(r|\omega_c, \theta_c)$ that specifies how likely the observed signal $r(t)$ is for a given pair (ω_c, θ_c), i.e. for a given candidate signal $s_c(\omega_c, \theta_c, t) \triangleq \sqrt{2}A\sin(\omega_c t + \theta_c)$. We can then obtain an unconditional likelihood function $p(r|\omega_c)$ by averaging across all allowable phases according to

$$p(r|\omega_c) = \int_{-\infty}^{\infty} p(r|\omega_c, \theta)p(\theta)d\theta \tag{9D.1}$$

where $p(\theta)$ denotes the distribution function of θ_0. Since we have no prior knowledge of θ_0 we must consider θ_0 to be uniformly distributed in $[-\pi, \pi]$. Correspondingly

$$p(r|\omega_c) = \frac{1}{2\pi}\int_{-\pi}^{\pi} p(r|\omega_c, \theta)d\theta. \tag{9D.2}$$

The maximum-likelihood estimate $\tilde{\omega_0}$ of ω_0 is the value of ω_c that maximizes $p(r|\omega_c)$. It remains to determine $p(r|\omega_c)$. To this end we note that s_c is a deterministic function of ω_c, θ and t. Furthermore, the only non-deterministic component of $r(t)$ is the noise $n(t)$. This noise is Gaussian and white with power spectral density N_0 so that

$$p(r|\omega_c, \theta) = \frac{1}{\sqrt{4\pi N_0}}\exp\{-\frac{1}{2N_0}\int_0^{T_o}(r(t) - s_c(\omega_c, \theta, t))^2 dt\}. \tag{9D.3}$$

In order to simplify this expression we note that

$$\begin{aligned}\int_0^{T_o}(r(t) - s_c(\omega_c, \theta, t))^2 dt &= \int_0^{T_o} r^2(t)d(t) - 2\int_0^{T_o} r(t)s_c(\omega_c, \theta, t)dt \\ &\quad + \int_0^{T_o} s_c^2(\omega_c, \theta, t)dt. \end{aligned} \tag{9D.4}$$

The first term on the right hand side depends neither on ω_c nor on θ. Similarly, the last term is the integral over a DC-value plus a fast oscillation ($\omega_c T_o \ll 1$) and is therefore approximately independent of ω_c and θ. Upon combining these two terms into a constant C we may write (9D.3) as

$$p(r|\omega_c, \theta) = C'\exp\{\frac{1}{N_0}\int_0^{T_o} r(t)s_c(\omega_c, \theta, t)dt\} \tag{9D.5}$$

where $C' \triangleq \frac{1}{\sqrt{4\pi N_0}}\exp\{-\frac{C}{2N_0}\}$. It follows that

$$p(r|\omega_c) = \frac{C'}{2\pi}\int_{-\pi}^{\pi}\exp\{\frac{1}{N_0}\int_0^{T_o} r(t)s_c(\omega_c, \theta, t)dt\}d\theta. \tag{9D.6}$$

We now expand s_c according to $s_c(\omega_c, \theta, t) = 2\sqrt{2}A[\cos\theta\cos\omega_c t + \sin\theta\sin\omega_c t]$ into a sine and cosine term. Based on this decomposition we may write $p(r|\omega_c)$ after some elementary manipulations as

$$p(r|\omega_c) = C' I_0(\frac{\sqrt{2}A}{N_0}\sqrt{X^2(\omega_c) + Y^2(\omega_c)}) \tag{9D.7}$$

where

$$X(\omega_c) \triangleq \int_0^{T_o} r(t)\cos\omega_c t dt, \tag{9D.8}$$

$$Y(\omega_c) \triangleq \int_0^{T_o} r(t) \sin \omega_c t dt \tag{9D.9}$$

and $I_0(z) \triangleq \frac{1}{\pi} \int_0^{\pi} \exp(x \cos \theta) d\theta$ is the modified Bessel function of first kind and zeroth order [2, Chapter 9]. This function increases monotonically with $X^2 + Y^2$. Maximization of $p(r|\omega_c)$ is, therefore, tantamount to maximizing $S_r(\omega_c) \triangleq X^2(\omega_c) + Y^2(\omega_c)$ over all ω_c. Fig. 9D.1 shows how $S_r(\omega_c)$ may be determined from $r(t)$ via coherent demodulation. We leave it as an exercise to the reader to show that $S_r(\omega)$ may alternatively be determined according to (9.11) from the autocorrelation $R(\tau)$ as defined in (9.10).

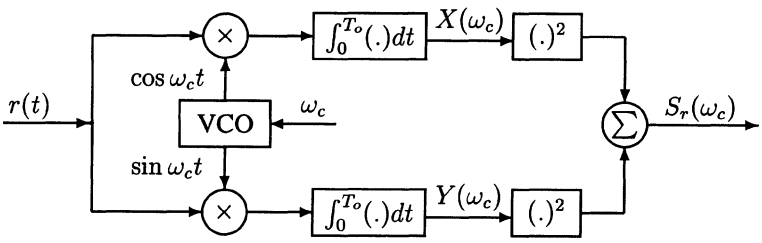


Figure 9D.1: Construction of likelihood indicator function $S_r(\omega_c)$ via coherent demodulation.

PROBLEMS

9.1 In Fig. 9.7, the sequences y_k and $\hat{a}_k$ have a sampling rate $1/T$ that is incommensurate with the clock rate N/T_x of the DSP. How does this incommensurability manifest itself in the time instants at which y_k and $\hat{a}_k$ become available? Is this a serious problem when $N/T_x \gg 1/T$?

9.2 Assume that noise $n(t)$ has power spectral density $\alpha \mathcal{N}(\Omega)$. Demonstrate that the efficiency γ of (9.3) does not depend on α if and only if the noise-induced TED output component u_k depends linearly on the noise $n(t)$ of the channel.

9.3 Assume that the first-order PLL of Fig. 9.14 operates at some baseline values of SNR and Ω_c.

1. Assume that Ω_c is doubled. How does this affect the time constant $1/K_t$ of the loop? How does it affect RMS jitter σ_Δ?
2. Assume that SNR is decreased by 3 dB. How does this affect $1/K_t$ and σ_Δ?

9.4 Prove (9A.1) and (9A.2).

9.5 Prove (9A.6) and (9A.7).

9.6 Consider the PLL of Fig. 9B.4. Assume that the VCO delivers orthogonal square waves of frequency ω_1 instead of orthogonal sine waves. Does this hamper the operation of the PLL

in any way? Does it facilitate PLL implementation? What happens if $r(t)$ is not a pure sinusoid embedded in noise, but also contains harmonics of the fundamental frequency ω_1?

9.7 Use (9B.4) to verify that (9B.5) remains valid irrespective of the absolute value of $\phi(t)$, i.e. that the quadricorrelator, unlike the phase detector according to Fig. 9B.3, exhibits no phase discontinuities.

9.8 In this problem we are concerned with the relation between the efficiency factors $\gamma(\underline{a}_N)$ and γ_{ML} of (9.8) and (10.15). We recall that $\gamma(\underline{a}_N)$ pertains to a *deterministic* data sequence $\underline{a}_N = [a_0, \ldots, a_{N-1}]$ of *finite* length N, while γ_{ML} pertains to *random* data sequences of *infinite* length.

1. Let us assume that the preamble $\underline{a}_N$ is drawn randomly from an ensemble, and that the associated stochastic process is ergodic. Then $\sigma^2_\Delta(\underline{a}_N)$ of (9.7) will become a stochastic function of $\underline{a}_N$ with expected value $\sigma^2_\Delta(N) \triangleq E_a[\sigma^2_\Delta(\underline{a}_N)]$ where $E_a[.]$ denotes expectation across the data ensemble. Prove that $\sigma^2_\Delta(N) = 1/(N\gamma(N)\mathrm{SNR})$ where
$$\gamma(N) \triangleq \frac{1}{\mathrm{SNR}} \int_{-\infty}^{\infty} (2\pi\Omega)^2 \mathcal{A}_N(e^{j2\pi\Omega})\mathrm{SN}(\Omega)d\Omega \tag{9P.1}$$
and $\mathcal{A}_N(e^{j2\pi\Omega}) \triangleq E_a[|A(e^{j2\pi\Omega})|^2]/N$.
2. It can be observed that (9P.1) is similar to γ_{ML} according to (10.15). The only difference is that the power spectral density $\mathcal{A}_N(e^{j2\pi\Omega})$ in (9P.1) pertains to *finite* data sequences of length N while $\mathcal{A}(e^{j2\pi\Omega})$ in (10.15) corresponds to sequences of unrestricted length. Let us consider a stationary random process of which the sample functions are data sequences a_k of infinite length, and let us denote the corresponding power spectral density by $\mathcal{A}(e^{j2\pi\Omega})$. Let us further consider a sub-process that is obtained by isolating the samples $a_0, \ldots, a_{N-1}$ of a_k into a preamble $\underline{a}_N = [a_0, \ldots, a_{N-1}]$, and let us denote the corresponding power spectral density by $\mathcal{A}_N(e^{j2\pi\Omega})$. Let us finally denote the autocorrelation function of a_k (i.e. the inverse Fourier transform of $\mathcal{A}(e^{j2\pi\Omega})$) by α_k. Prove that $\mathcal{A}(e^{j2\pi\Omega}) = \mathcal{A}_N(e^{j2\pi\Omega})$ if and only if $\alpha_k = 0$ for $|k| > N$. What happens if this condition is not met? Illustrate this with a simple example.

9.9 This problem is concerned with the efficiency factor $\gamma(\underline{a}_N)$ according to (9.8) for sinusoidal preambles with normalized frequency Ω_0 and length N according to
$$a_k = \begin{cases} \sin 2\pi\Omega_0 k & \text{for } k \in \{0, \ldots, N-1\}, \\ 0 & \text{else.} \end{cases} \tag{9P.2}$$

1. Assume that the sinusoid has an integer-valued period of L symbol intervals ($\Omega_0 = 1/L$), and that $N = mL$ for some integer m, i.e. that the preamble consists of m full periods of the sinusoid. Show that the preamble has data spectrum
$$|A(e^{j2\pi\Omega})|^2 = [\frac{\sin 2\pi\Omega_0 \sin \pi\Omega N}{\cos 2\pi\Omega_0 - \cos 2\pi\Omega}]^2. \tag{9P.3}$$
2. Sketch $|A(e^{j2\pi\Omega})|^2/N$ for $L = 2$ and several values of N. Repeat this exercise for $L = 4$. Explain the tendencies.
3. Determine the power $P_a \triangleq \frac{1}{N}\sum_{k=0}^{N-1} a_k^2$.

4. Consider the limit as $m \to \infty$. Express $|A(e^{j2\pi\Omega})|^2$ in terms of Ω_0 and P_s. Assume that the channel passess only the fundamental frequency Ω_0, i.e. that $\mathrm{SN}(\pm\Omega_0 + n) = 0$ for all $n \neq 0$. Express the efficiency factor $\gamma(\underline{a}_N)$ according to (9.8) in terms of P_s, Ω_0 and $\mathrm{SN}(\Omega_0)$.

REFERENCES

[1] M.R. Aaron, 'PCM Transmission in the Exchange Plant', *Bell Syst. Tech. J*, Vol. 41, pp. 99-141, Jan. 1962.

[2] M. Abramowitz and I.A. Stegun, *Handbook of Mathematical Functions*. New York: Dover Publications, 1965.

[3] A.N. D'Andrea and U. Mengali, 'Design of Quadricorrelators for Automatic Frequency Control Systems', *IEEE Trans. Commun.*, Vol. COM-41, No. 6, pp. 988-997, June 1993.

[4] G. Ascheid, M. Oerder, J. Stahl, and H. Meyr, 'An All Digital Receiver Architecture for Bandwidth Efficient Transmission at High Data Rates', *IEEE Trans. Commun.*, Vol. COM-37, No. 8, pp. 804-813, Aug. 1989.

[5] M. Banu and A.E. Dunlop, 'Clock Recovery Circuits with Instantaneous Locking', *Electron. Lett.*, Vol. 28, No. 23, pp. 2127-2130, Nov. 5, 1992.

[6] R.H. Barker, 'Group Synchronization of Binary Digital Systems', in *Communication Theory*, W. Jackson (ed.) London: Butterworth Scientific Publications, 1953, pp. 273-287.

[7] S.K. Barton, European Patent No. 0 222 593 B1, June 6, 1991.

[8] W.R. Bennett, 'Statistics of Regenerative Data Transmission', *Bell Syst. Tech. J.*, Vol. 37, pp. 1501-1542, Nov. 1958.

[9] W.R. Bennett and J.R. Davey, *Data Transmission*. New York: McGraw-Hill, 1965.

[10] J.W.M. Bergmans, 'Performance Consequences of Timing Errors in Digital Magnetic Recording', *Philips J. Res.*, Vol. 42, No. 3, pp. 281-307, 1987.

[11] R.E. Best, *Phase-Locked Loops*. New York: McGraw-Hill, 1984.

[12] A. Blanchard, *Phase-Locked Loops*. New York: Wiley, 1976.

[13] R. Blahut, *Digital Transmission of Information*. Reading, MA: Addison-Wesley, 1990.

[14] R.D. Cideciyan, F. Dolivo, R. Hermann, W. Hirt, and W. Schott, 'A PRML System for Digital Magnetic Recording', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 38-56, Jan. 1992.

[15] J.P. Costas, 'Synchronous Communications', *Proc. IRE*, Vol. 44, pp. 1713-1718, Dec. 1956.

[16] R.E. Crochiere and L.R. Rabiner, *Multirate Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[17] R.C. Den Dulk, *An Approach to Systematic Phase-Lock Loop Design*, Ph.D. Thesis, Delft University of Technology, The Netherlands, Nov. 1989.

[18] L. Erup, F.M. Gardner and R.A. Harris, 'Interpolation in Digital Modems - Part II: Implementation and Performance', *IEEE Trans. Commun.*, Vol. COM-41, No. 6, pp. 998-1008, June 1993.

[19] L.E. Franks, 'Carrier and Bit Synchronization in Data Communication - A Tutorial Review', *IEEE Trans. Commun.*, Vol. COM-28, pp. 1107-1121, Aug. 1980.

[20] A. Gabor, 'High-Density Recording on Magnetic Tape', *Electronics*, Vol. 32, pp. 72-75, Oct. 16, 1959.

[21] F.M. Gardner, *Phaselock Techniques*. New York: Wiley, 1966 (2nd ed. 1979).

[22] F.M. Gardner, 'Self-Noise in Synchronizers', *IEEE Trans. Commun.*, Vol. COM-28, No. 8, pp. 1159-1163, Aug. 1980.

[23] F.M. Gardner, 'Properties of Frequency Difference Detectors', *IEEE Trans. Commun*, Vol. COM-33, No. 3, pp. 131-138, Feb. 1985.

[24] F.M. Gardner, 'Interpolation in Digital Modems - Part I: Fundamentals', *IEEE Trans. Commun.*, Vol. COM-41, No. 3, pp. 501-507, March 1993.

[25] R.D. Gitlin and J.F. Hayes, 'Timing Recovery and Scramblers in Data Transmission', *Bell Syst. Tech. J.*, Vol. 54, pp. 569-593, March 1975.

[26] R.D. Gitlin, 'Adaptive Phase-Jitter Tracker', U.S. patent No. 4,320,526, March 16, 1982.

[27] B.S. Glance, 'New Phase-Lock Loop Circuit Providing Very Fast Acquisition', *IEEE Trans. on Microwave Theory and Techniques*, Vol. 33, No. 9, Sept. 1985.

[28] S.W. Golomb, J.R. Davey, I.S. Reed, H.L. Van Trees and J.J. Stiffler, 'Synchronization', *IEEE Trans. Commun. Systems*, Vol. CS-11, pp. 481-491, Dec. 1963.

[29] R.P. Gooch and M.J. Ready, 'An Adaptive Phase-Lock Loop for Phase Jitter Tracking', Proc. 21st Asilomar Conf. on Signals, Systems and Computers, Nov. 2-4, 1987.

[30] S.C. Gupta, 'Phase-Locked Loops', *Proc. IEEE*, Vol. 63, pp. 291-306, Feb. 1975.

[31] J.P. Harris, W.B. Phillips, J.F. Wells, and W.D. Winger, 'Innovations in the Design of Magnetic Tape Subsystems', *IBM J. Res. Develop.*, Vol. 25, No. 5, pp. 691-699, Sept. 1981.

[32] C.W. Helstrom, *Statistical Theory of Signal Detection*. Oxford: Pergamon press, 1960.

[33] A. Hoogendoorn, 'Digital Compact Cassette', *Proc. IEEE*, Vol. 82, No. 10, pp. 1479-1489, Oct. 1994.

[34] P. Kabal and S. Pasupathy, 'Partial-Response Signaling', *IEEE Trans. Commun.*, Vol. COM-23, pp. 921-934, Sept. 1975.

[35] E.A. Lee and D.G. Messerschmitt, *Digital Communication*. Boston: Kluwer Academic Publishers, 1988.

[36] J.-Y. Lin and C.-H. Wei, 'Fast Timing Recovery Scheme for Class IV Partial Response Channels', *Electron. Lett.*, Vol. 31, No. 3, pp. 159-161, Feb. 2, 1995.

[37] W.C. Lindsey, *Synchronization Systems in Communications*. Englewood Cliffs, N.J.: Prentice-Hall, 1972.

[38] W.C. Lindsey and M.K. Simon (Eds.), *Phase-Locked Loops and Their Applications*. New York: IEEE Press, 1978.

[39] W.C. Lindsey and C.M. Chie, 'A Survey of Digital Phase-Locked Loops', *Proc. IEEE*, Vol. 69, pp. 410-431, Apr. 1981.

[40] W.C. Lindsey and C.M. Chie (Eds.), *Phase-Locked Loops*. New York: IEEE Press, 1985.

[41] B.H. Marcus, P.H. Siegel and J.K. Wolf, 'Finite-State Modulation Codes for Data Storage', *IEEE J. Selected Areas in Commun.*, Vol. SAC-10, No. 1, pp. 5-37, Jan. 1992.

[42] J.L. Massey, 'Optimum Frame Synchronization', *IEEE Trans. Commun.*, Vol. COM-20, No. 2, pp. 115-119, Apr. 1972.

[43] D.G. Messerschmitt, 'Frequency Detectors for PLL Acquisition in Timing and Carrier Recovery', *IEEE Trans. Commun.*, Vol. COM-27, No. 9, pp. 1288-1295, Sept. 1979.

[44] H. Meyr and G. Ascheid, *Synchronization in Digital Communications - Volume I.* New York: Wiley, 1990.

[45] C.S. Miller, European patent No. 0 092 400, issued 09.11.1988.

[46] K.H. Mueller and M. Müller, 'Timing Recovery in Digital Synchronous Data Transmission', *IEEE Trans. Commun.*, Vol. COM-24, No. 5, pp. 516-531, May 1976.

[47] M. Oerder and H. Meyr, 'Digital Filter and Square Timing Recovery', *IEEE Trans. Commun.*, Vol. COM-36, No. 5, pp. 605-612, May 1988.

[48] A.V. Oppenheim and R.W. Schafer, *Digital Signal Processing*. Englewood Cliffs, N.J.: Prentice-Hall, 1975.

[49] J.H. Park, Jr., 'An FM Detector for Low S/N', *IEEE Trans. Commun.*, Vol. COM-18, pp. 110-118, Apr. 1970.

[50] A.A. Pistolkors, Russian Author's Certificate 34039, April 24, 1932.

[51] S.U.H. Qureshi, 'Adaptive Equalization', *Proc. IEEE*, Vol. 73, No. 9, pp. 1349-1387, Sept. 1985.

[52] T.A. Ramstad, 'Digital Methods for Conversion Between Arbitrary Sampling Frequencies', *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-32, No. 3, pp. 577-591, June 1984.

[53] B.R. Saltzberg, 'Timing Recovery for Digital Synchronous Data Transmission', *Bell Syst. Tech. J.*, Vol. 46, pp. 593-622, March 1967.

[54] H. Sari, H. Houeix, and G. Karam, 'Phase and Frequency Detectors for Clock Synchronization in High-Speed Optical Transmission Systems', *European Trans. on Telecommun. and Related Technologies*, Vol. 5, No. 5, pp. 635-641, Sept./Oct. 1994.

[55] J.E. Savage, 'Some Simple Self-Synchronizing Digital Data Scramblers', *Bell Syst. Tech. J.*, Vol. 46, pp. 449-487, Feb. 1967.

[56] P.H. Siegel, 'Recording Codes for Digital Magnetic Storage', *IEEE Trans. Magn.*, Vol. MAG-21, No. 5, pp. 1344-1349, Sep. 1985.

[57] P.H. Siegel and J.K. Wolf, 'Modulation and Coding for Information Storage', *IEEE Communications Magazine*, Vol. 29, No. 12, pp. 68-86, Dec. 1991.

[58] J.J. Stiffler, *Theory of Synchronous Communications*. Englewood Cliffs, N.J.: Prentice-Hall, 1971.

[59] E.F. Stikvoort and J.A.C. v. Rens, 'An All-Digital Bit Detector for Compact Disc Players', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 191-200, Jan. 1992.

[60] E.D. Sunde, 'Self-Timing Regenerative Repeaters', *Bell Syst. Tech. J*, Vol. 36, pp. 891-937, Jul. 1957.

[61] J.M. Tribolet, 'A New Phase Unwrapping Algorithm', *IEEE Trans. Acoust., Speech, Signal Processing*, Vol. ASSP-25, No. 2, pp. 170-177, Apr 1977.

[62] A.J. Viterbi, *Principles of Coherent Communications*. New York: McGraw-Hill, 1966.

[63] B. Widrow, J.M. McCool, M.G. Larimore and R. Johnson, Jr., 'Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter', *Proc. IEEE*, Vol. 64, No. 8, pp. 1151-1162, Aug. 1976.

[64] P.M. Woodward, *Probability and Information Theory, with Applications to Radar.* McGraw-Hill: New York, 1954.

[65] L.E. Zegers, 'Common-Bandwidth Transmission of Data Signals and Wide-Band Pseudonoise Synchronization Waveforms', Ph.D. Thesis, Twente University of Technology, The Netherlands, June 29, 1972.

10

A CATALOG OF TIMING-RECOVERY SCHEMES

10.1 INTRODUCTION

This chapter presents a range of timing-recovery schemes in more or less chronological order. The emphasis is on the manner in which timing information is extracted from the incoming signal. We refer back to Chapter 9 for a review of basics of timing recovery, and ahead to Chapter 11 for a treatment of PLLs. Each section of the chapter is devoted to a single category of schemes. An attempt has been made to make these treatments self-contained. The reader is encouraged to skip those sections that are of no concern to him. A selection may be based on the concise overview below, or on the introductions of the sections. To facilitate comparisons between schemes, the same notation is used throughout the chapter, and the same idealized recording channels are used for benchmarking purposes. Timing recovery has been a subject of active research for many years, and a vast array of techniques has evolved. Our coverage in this chapter does include some of the mainstream approaches, but is by no means complete. We recommend [22, 50, 44] and [36, Chapter 15] as starting points for further reading.

The catalog is arranged as follows. Section 10.2 derives maximum-likelihood data-aided (DA) and non-data-aided (NDA) timing-recovery techniques and analyzes their performance. These optimum techniques serve as a basis of reference for the other schemes in the chapter.

Sections 10.3 and 10.4 are devoted to two classical deductive techniques in which a timing tone is extracted from the received signal. The average phase of this tone defines the decision instants. In the approach of Section 10.3 the timing tone is produced via a threshold-crossing detector, while a symmetric nonlinearity is used in the one of Section 10.4. Both schemes are applicable to full-response and partial-response systems, and work best when excess bandwidth is plentiful.

Section 10.5 is concerned with early-late timing recovery. Here timing information is extracted with the aid of samplers that close just before and just after the decision instants. The approach has a non-data-aided (NDA) variant that is applicable to full-response systems with significant excess bandwidth, and a DA variant that is suited for a broader category of channels and reception schemes.

Sections 10.6 and 10.7 provide a development of minimum mean-square error (MMSE) and zero-forcing (ZF) timing recovery and of several related approaches. These DA schemes are closely related to MMSE and ZF adaptation schemes (see Chapter 8), and can be used with a variety of modulation and detection techniques. For systems with little excess bandwidth the sampling rate can be as low as the baud rate, while tracking performance can approach that of maximum-likelihood schemes. In the baseline MMSE and ZF schemes, timing information is extracted from an error e that represents the difference between the actual detector input and the desired one. The error e may be replaced by one of these two inputs at a modest expense to pattern-dependent jitter, and this leads to schemes that encompass the classical sampled-derivative approach.

Throughout the chapter the emphasis is more on principles of the various schemes than on digital and analog implementation issues. These important issues are explored, for example, in [42] and [63].

10.2 MAXIMUM-LIKELIHOOD TIMING RECOVERY

In this section we discuss a timing-recovery technique that is optimum whenever the receiver has no prior knowledge about the delay ϕ of the channel. The technique identifies the normalized delay $\tilde{\phi}$ that has greatest likelihood, given the received signal $r(t)$. This is analogous to the maximum-likelihood sequence detector, which identifies the data sequence $\hat{a}_k$ that has greatest *a posteriori* likelihood. We have seen in Chapters 3 and 7 that maximum-likelihood sequence detection tends to be highly complicated. This is not necessarily true for maximum-likelihood (ML) timing recovery. Even so, several other techniques are either simpler or more compatible to various detection and adaptation techniques, while affording close-to-ML performance. For this reason the practical significance of ML timing recovery is somewhat restricted. The main purpose of this section is, therefore, to provide a basis of reference for future schemes, both performance-wise and topology-wise. ML timing recovery comes in both data-aided (DA) and non-data-aided (NDA) varieties. We will see that the NDA variety degenerates into a decision-directed form of the DA variety whenever SNRs are high enough to support reliable bit-decisions. Only at very poor SNRs does a real difference between DA and NDA emerge.

10.2.1 Data-aided maximum-likelihood timing recovery

In Fig. 10.1, a finite data sequence a_k of data rate $1/T$ is transmitted across a channel with finite impulse response $h(t)$, additive Gaussian noise $n(t)$ with zero mean, and a fixed but *a priori* unknown delay ϕT. For the moment we assume that noise is white.

The timing-recovery subsystem of the receiver is allowed to observe $r(t)$ as long as it desires in order to produce an estimate $\tilde{\phi}$ of ϕ. It is assumed to have full knowledge of

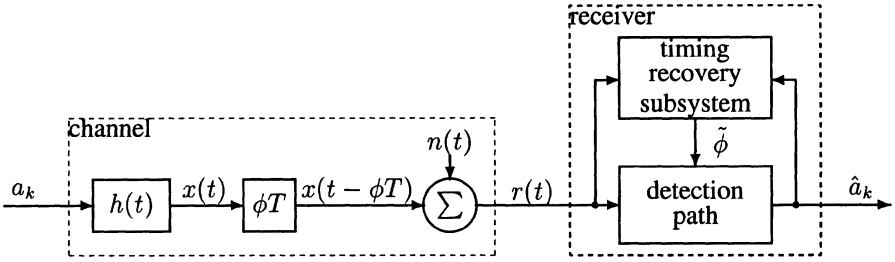


Figure 10.1: System model for maximum-likelihood data-aided timing recovery.

a_k and $h(t)$. For this reason it can compute a signal $x(t-\psi T) \triangleq \sum_k a_k h(t-(k+\psi)T)$ that would be received if the channel had delay ψT and noise were absent. The error signal $e(t) \triangleq r(t) - x(t-\psi T) = x(t-\phi T) - x(t-\psi T) + n(t)$ consists only of noise $n(t)$ when the candidate delay ψT coincides with the actual channel delay ϕT. For all other candidate delays it will also contain a data component, and its power is bound to increase. This suggests that we choose the value of ψ that minimizes the energy of $e(t)$ as our estimate $\tilde{\phi}$ of ϕ. This estimate has maximum likelihood because noise is Gaussian and white [22] (see also Appendix 3C). The objective, then, is to minimize the function

$$\Lambda(\psi) \triangleq \int_{-\infty}^{\infty} e^2(t)dt \tag{10.1}$$

over all ψ. We may equivalently look for the central zero-crossing of the derivative $\dot{\Lambda}(\psi)$ of Λ with respect to ψ. This is possible by means of a *gradient search*. Such a search starts from some initial value of ψ, and changes ψ in small steps in proportion to $\dot{\Lambda}(\psi)$. This will cause ψ to converge towards $\tilde{\phi}$. One easily verifies that

$$\dot{\Lambda}(\psi) = 2\int_{-\infty}^{\infty} e(t)\frac{\partial e(t)}{\partial \psi}dt = 2T\int_{-\infty}^{\infty} e(t)x'(t-\psi T)dt, \tag{10.2}$$

where "$'$" denotes time-derivative. This suggests the receiver topology of Fig. 10.2.

The receiver contains a detection path that produces decisions $\hat{a}_k$, which we assume to be correct. The decision instants $t_k \triangleq (k+\psi)T$ are determined by a voltage-controlled oscillator (VCO) that forms part of the timing-recovery subsystem. The receiver knows the channel impulse response $h(t)$. Using decisions $\hat{a}_k$ that are paced by the VCO, it can compute a data signal

$$\sum_{k=-\infty}^{\infty} a_k h(t-t_k) = x(t-\psi T) \tag{10.3}$$

as well as its derivative $x'(t-\psi T)$. Subtraction of $x(t-\psi T)$ from $r(t)$ yields the error signal $e(t)$, which is multiplied by $x'(t-\psi T)$. The resulting signal $\chi(t)$ is the indicator function $\dot{\Lambda}(\psi)$ of (10.2), except for an integration. The loop is closed via a loop

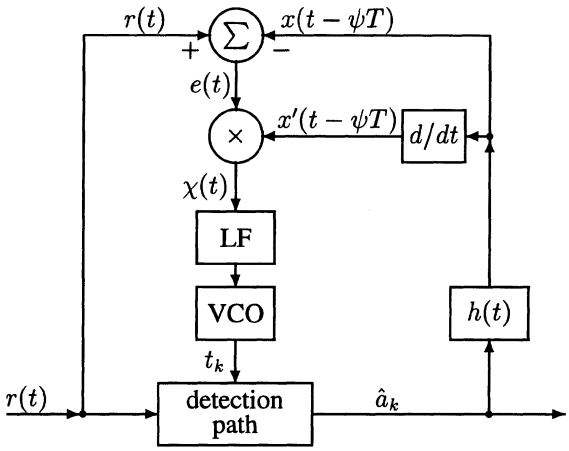


Figure 10.2: Data receiver with decision-directed timing-recovery loop based on the gradient (10.2).

filter (LF) and will act to suppress any rapid variations of $\chi(t)$. This action approximates the integration in (10.2), especially when the loop is dimensioned to have a small bandwidth. In that case the loop might, however, become too slow to track dynamic fluctuations of the channel delay ϕT. For this reason one would rather use a bandwidth that permits adequate tracking while reducing jitter of $\Delta = \psi - \phi$ to acceptable proportions. For more details about this tradeoff we refer to Chapter 11.

At the ideal sampling phase $\psi = \phi$, the error $e(t)$ consists purely of noise. As a result, steady-state fluctuations of $\chi(t)$ and Δ are induced by noise, and pattern-dependent jitter is negligible.

In spite of its simple structure, the loop of Fig. 10.2 has restricted practical value. One reason is that analog latencies in the r, x and x' branches must be accurately matched. Another reason is that an analog filter is needed whose impulse response is an accurate copy of the channel impulse response $h(t)$. In practice it may not be easy to meet these requirements. In Section 10.7 we will encounter a discrete-time counterpart of the loop of Fig. 10.2 that goes without these disadvantages and affords close-to-ML performance when excess bandwidth is small. We proceed to develop an alternative DA ML TED that uses a matched filter.

10.2.2 ML DA TED based on matched filter

Let us reconsider expression (10.2) for $\dot{\Lambda}(\psi)$. Upon realizing that $e(t) = r(t) - x(t-\psi)$, we arrive at the equivalent expression

$$\dot{\Lambda}(\psi) = 2T \int_{-\infty}^{\infty} r(t)x'(t-\psi T)dt - 2T \int_{-\infty}^{\infty} x(t-\psi T)x'(t-\psi T)dt. \qquad (10.4)$$

The second integral amounts to zero. This is readily shown by using Parseval's equality and exploiting the fact that x and x' are both real-valued. We should, therefore, choose ψ in order to eliminate any cross-correlation between $r(t)$ and $x'(t-\psi T)$. Now $x'(t-\psi T)$ is the derivative of the signal $x(t-\psi T) = \sum_{k=-\infty}^{\infty} a_k h(t-t_k)$ (see (10.3)). Consequently

$$\begin{aligned}\dot{\Lambda}(\psi) &= 2T\int_{-\infty}^{\infty} r(t)\sum_{k=-\infty}^{\infty} a_k h'(t-t_k)dt = 2T\sum_{k=-\infty}^{\infty} a_k \int_{-\infty}^{\infty} r(t)h'_-(t_k-t)dt \\ &= 2T\sum_{k=-\infty}^{\infty} a_k (r * h'_-)(t_k) = 2T\sum_{k=-\infty}^{\infty} a_k (r * h_-)'(t_k), \qquad (10.5)\end{aligned}$$

where '$*$' and '$_$' denote linear convolution and time-reversal, respectively. We arrive at the circuit of Fig. 10.3.

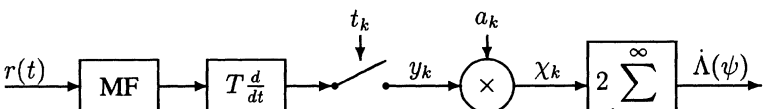


Figure 10.3: Maximum-likelihood timing-error detector based on the differentiated output of a matched filter.

The received signal $r(t)$ is applied to a matched filter (MF), which has impulse response $h_-(t) \triangleq h(-t)$. The output of the MF is scaled, differentiated, sampled at the instants $t_k = (k+\psi)T$ and multiplied with the data sequence a_k. The result is a discrete-time sequence χ_k that is averaged to obtain $\dot{\Lambda}(\psi)$. By analogy to Fig. 10.2, the averaging operation is replaced in practice by the action of the loop, and we may regard χ_k as the output of the TED.

Exercise 10.1 Assume that the data sequence a_k has a finite length N. Show that the summation in (10.5) can be restricted to N terms.

If noise is colored rather than white then the transfer function of the MF becomes $H^*(\Omega)/\mathcal{N}(\Omega)$ (see Section 3.8; we recall that Ω is a normalized measure of frequency, with $\Omega = 1$ corresponding to the symbol rate $1/T$). Irrespective of the noise spectrum, the MF concentrates all information about the transmitted data at the optimum sampling phase $\psi = \phi$ (see Chapter 3). The present analysis reveals that it also concentrates the timing information that is present in the received signal in the vicinity of this phase. For periodic data a_k, the multiplication by a_k in Fig. 10.3 may be viewed as a form of coherent demodulation. After being partly stripped from noise (by the MF) and sampled, the data is demodulated, and the DC-component of the resulting baseband signal is detected.

10.2.3 Characteristics of the ML DA TED

Transfer characteristic The sampler output of Fig. 10.3 is of the form $y_k = (a * q^{\Delta})_k + v_k$ where q_k^{Δ} is the impulse response of the system up until the sampler and v_k is filtered and sampled channel noise. The TED output $\chi_k = a_k y_k$ may be separated into data and noise components according to

$$\chi_k = a_k(a * q^{\Delta})_k + u_k, \tag{10.6}$$

where $u_k \triangleq a_k v_k$. The expected value $E[\chi_k]$ of χ_k constitutes the desired output $\rho(\Delta)$ of the TED and is an antisymmetric function of Δ about $\Delta = 0$. The undesired components of χ_k are noise u_k and a data-dependent term $z_k \triangleq a_k(a * q^{\Delta})_k - \rho(\Delta)$ that induces pattern-dependent jitter. In Appendix 10A.1 it is shown that

$$\rho(\Delta) = -\int_{-\infty}^{\infty} 2\pi\Omega\mathcal{A}(e^{j2\pi\Omega})\mathrm{SN}(\Omega)\sin(2\pi\Delta\Omega)d\Omega \tag{10.7}$$

where $\mathcal{A}(e^{j2\pi\Omega})$ denotes the data PSD and

$$\mathrm{SN}(\Omega) \triangleq \frac{1}{T}\frac{|H(\Omega)|^2}{\mathcal{N}(\Omega)} \tag{10.8}$$

denotes the spectral signal-to-noise ratio of the channel. Two types of data merit particular attention:

Uncoded data: Let a_k be uncorrelated with zero mean and unit power. Then $\mathcal{A}(e^{j2\pi\Omega}) = 1$ for all Ω, whence

$$\rho(\Delta) = -\int_{-\infty}^{\infty} 2\pi\Omega\,\mathrm{SN}(\Omega)\,\sin(2\pi\Delta\Omega)d\Omega. \tag{10.9}$$

Exercise 10.2 Consider the situation that noise $n(t)$ is white. Show that $\rho(\Delta) \propto (h * h_-)'(\Delta T)$, i.e. that the TED transfer characteristic is essentially a differentiated version of the channel autocorrelation function.

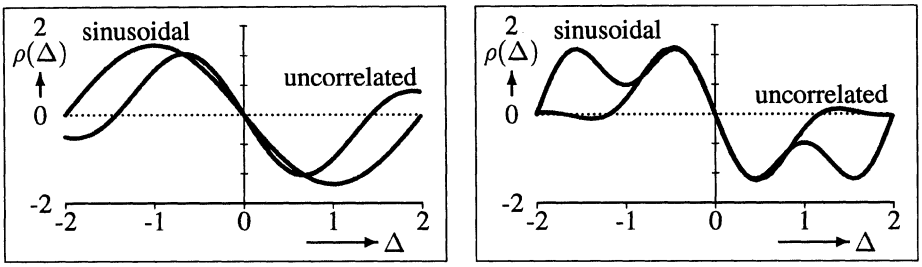
Sinusoidal clock run-in patterns: Let a_k be a sinusoid with unit power and a period of M symbol intervals T, and let $\Omega_0 \triangleq 1/M$. Then a_k will have a PSD in the form of a line spectrum

$$\mathcal{A}(e^{j2\pi\Omega}) = \frac{1}{2}\sum_{n=-\infty}^{\infty}[\delta(\Omega + \Omega_0 + n) + \delta(\Omega - \Omega_0 - n)], \tag{10.10}$$

where $\delta(\Omega)$ is Dirac's delta function. To determine $\rho(\Delta)$ we use the sieve property of Dirac's delta function and the fact that SN is an even function of Ω. It is found that

$$\rho(\Delta) = \sum_{n=-\infty}^{\infty} -2\pi(\Omega_0 + n)\mathrm{SN}(\Omega_0 + n)\sin 2\pi\Delta(\Omega_0 + n). \tag{10.11}$$

This is a periodic transfer characteristic with a period of M symbol intervals T. It becomes sinusoidal when only the fundamental frequency Ω_0 is passed by the channel.



$\beta = 0$ $\beta = 1$

Figure 10.4: Transfer characteristics for uncorrelated data and a sinusoidal training pattern of period $4T$ of the data-aided timing-error of Fig. 10.3. The spectral signal-to-noise ratio $\mathrm{SN}(\Omega)$ of the channel has a raised-cosine characteristic with roll-off factor β.

Example 10.1 Let $\mathrm{SN}(\Omega)$ be a raised-cosine characteristic with roll-off factor β according to

$$\mathrm{SN}(\Omega) = \begin{cases} C & \text{for } |\Omega| \leq \frac{1-\beta}{2}, \\ \frac{C}{2}[1 + \sin\frac{\pi}{\beta}(\frac{1}{2} - |\Omega|)] & \text{for } \frac{1-\beta}{2} < |\Omega| < \frac{1+\beta}{2}, \text{ and} \\ 0 & \text{else,} \end{cases} \tag{10.12}$$

where C is a constant. This function fully determines the folded signal-to-noise ratio

$$\mathrm{FSN}(e^{j2\pi\Omega}) \triangleq \sum_{n=-\infty}^{\infty} \mathrm{SN}(\Omega + n) \tag{10.13}$$

and the average signal-to-noise ratio $\mathrm{SNR} = \int_{-0.5}^{0.5} \mathrm{FSN}(e^{j2\pi\Omega})d\Omega$ as defined in Chapter 3. One readily verifies that $\mathrm{FSN}(e^{j2\pi\Omega}) = C$ for all Ω, whence $\mathrm{SNR} = C$. Since FSN does not depend on frequency, linear equalization of this channel suffices for optimum decisions (see Chapter 5). Fig. 10.4 depicts TED transfer characteristics for uncorrelated data a_k and the sinusoid $..++--++--..$ of period $4T$ and frequency $\Omega_0 = 1/4$. Apart from a central zero-crossing for $\Delta = 0$, there are spurious crossings to which a tracking loop might lock. For the $4T$ pattern, the only effect of such a lock is a sampling-phase ambiguity of one or more units $4T$. This ambiguity can be resolved by means of a word synchronization mechanism and is, therefore, of no concern. For random data, however, $\rho(\Delta)$ has spurious crossings for non-integer values of Δ. Bit decisions would be in error if the loop would lock up here. One way to avoid such a *false lock* is to ensure that tracking is started in the vicinity of $\Delta = 0$. A periodic clock run-in pattern may be used to this end. Even without this procedure, false locks may not be a problem when the timing-recovery subsystem is in the decision-directed mode of operation. The reason is that erroneous decisions near the spurious crossings are likely to cause the loop to drift away randomly, and only the central crossing is likely to be stable.

Exercise 10.3 The transfer characteristics of Fig. 10.4 pertain to the situation that the actual data symbols a_k are used within the timing-recovery subsystem. Argue how these characteristics will change qualitatively if the timing-recovery subsystem is operated in a decision-directed mode.

Pattern-dependent jitter The TED output χ_k comprises a data-dependent component $z_k = a_k(a * q^{\Delta})_k - \rho(\Delta)$ that induces pattern-dependent jitter. In the tracking mode of operation, sampling-phase errors will be small so that $z_k \simeq a_k(a * q^0)_k$ to good approximation. The timing-recovery loop averages z_k across many symbol intervals and pattern-dependent jitter is, therefore, mainly determined by the DC-component of z_k. This component may be written

$$\sum_{k=-\infty}^{\infty} z_k = \sum_{k=-\infty}^{\infty} a_k(a * q^0)_k = (a * a_- * q^0)_0. \tag{10.14}$$

Now q_k^0 is an odd function of k while $(a * a_-)_k$ is even. The implication is that z_k is DC-free irrespective of a_k, i.e. that the power spectral density of z_k vanishes at DC. For this reason pattern-dependent jitter becomes rapidly insignificant as PLL bandwidth decreases [22]. The effect of system parameter variations on pattern-dependent jitter is examined in Problem 10.8.

Noise-induced jitter Since jitter is bound to be primarily induced by noise, performance of the TED may be characterized in terms of the *TED efficiency* γ of Section 9.3.3. In Appendix 10A.1 it is shown that the DA ML TED has efficiency

$$\gamma_{\text{ML}} = \frac{1}{\text{SNR}} \int_{-\infty}^{\infty} (2\pi\Omega)^2 \mathcal{A}(e^{j2\pi\Omega}) \text{SN}(\Omega) d\Omega = \frac{\int_{-\infty}^{\infty} (2\pi\Omega)^2 \mathcal{A}(e^{j2\pi\Omega}) \text{SN}(\Omega) d\Omega}{\int_{-\infty}^{\infty} \text{SN}(\Omega) d\Omega}. \tag{10.15}$$

This efficiency sets an upper limit to the tracking capabilities of practical timing-recovery schemes, and is closely related to the Cramér-Rao bound for parameter estimators [52, 31]. Because of the weighing by Ω^2 in the numerator, especially the high-frequency components of $\mathcal{A}$ and SN contribute to efficiency. This confirms the intuitive expectation that timing information resides in particular at high frequencies. Efficiency γ_{ML} depends rather strongly on the bandwidth of the channel as well as on the data spectrum. Expression (10.15) suggests that for optimum efficiency, data should have its energy concentrated in portions of the spectrum where $\Omega^2 \text{SN}(\Omega)$ is largest. This favours sinusoidal data with a proper frequency over random data.

Example 10.2 Let a_k be uncorrelated with unit power, so that $\mathcal{A} = 1$ for all Ω. Correspondingly

$$\gamma_{\text{ML}} = \frac{\int_{-\infty}^{\infty} (2\pi\Omega)^2 \text{SN}(\Omega) d\Omega}{\int_{-\infty}^{\infty} \text{SN}(\Omega) d\Omega}. \tag{10.16}$$

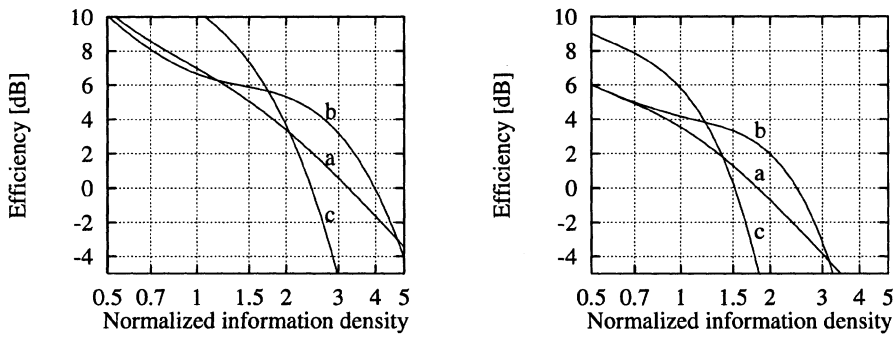
Let us assume that the bandwidth of the channel is variable such that $\text{SN}(\Omega) = F(\Omega/B)$ for some non-negative function F, where $B \geq 0$ is a linear measure of the bandwidth. Then SNR $\propto B$ and $\int_{-\infty}^{\infty} (2\pi\Omega)^2 \text{SN}(\Omega) d\Omega \propto B^3$, i.e. $\gamma \propto B^2$. This shows that the relative difficulty of accurate timing recovery and reliable symbol-detection depends strongly on B. In systems with a high spectral efficiency (small B), accurate timing recovery is comparatively difficult. At the same time it is comparatively critical to reliable data detection (see Section 9.2.5). Both effects are best served by a high TED efficiency.

Example 10.3 Let a_k be a sinusoid with unit power and a period of M symbol intervals T, and let $\Omega_0 \triangleq 1/M$. Upon invoking (10.10) we then find that

$$\gamma_{\mathrm{ML}} = \frac{1}{\mathrm{SNR}} \sum_{n=-\infty}^{\infty} (2\pi(\Omega_0 + n))^2 \mathrm{SN}(\Omega_0 + n). \tag{10.17}$$

Typically, Ω_0 will be in the passband of the channel, while the higher harmonics will be more or less rejected. Correspondingly $\gamma_{\mathrm{ML}} \simeq [(2\pi\Omega_0)^2 \mathrm{SN}(\Omega_0)]/\mathrm{SNR}$. For optimum efficiency we must choose Ω_0 in order to maximize $\Omega_0^2 \mathrm{SN}(\Omega_0)$.

Example 10.4 The Lorentzian model for digital magnetic recording channels, discussed in Section 2.3.5, is characterized by a normalized information density D. The channel has a zero at DC and its bandwidth decreases rapidly with D (see Fig. 2.25). Practical values of D are in the order of 1 to 3. The Gaussian model for optical recording channels, discussed in Section 2.4.3, has a low-pass nature and a parameter S that is similar to D. Practical values of S are in the order of 0.5 to 2. Bandwidth decreases rapidly with S (see Fig. 2.35).



a. Lorentzian magnetic recording channel b. Gaussian optical recording channel

Figure 10.5: Efficiency γ_{ML} for maximum-likelihood data-aided timing recovery on idealized digital magnetic and optical recording channels. a. Uncorrelated binary data; b. binary sine wave of period $4T$; c. binary sine wave of period $2T$.

Efficiencies γ_{ML} for these two models are depicted in Fig. 10.5 for random binary data a_k as well as the binary sinusoids '..++−−++−−..' and '..+−+−+−+−..' of period $4T$ and $2T$, respectively. The Lorentzian curves may be computed analytically (see Appendix 10A.2). Efficiency declines rapidly with D and S. The $4T$ pattern is preferable across a considerable range of densities. Because of its high frequency, the $2T$ pattern is unsuited for high densities. On the whole, efficiencies for the optical channel are lower than for the magnetic channel. The reason is that the optical channel passes mainly low-frequency data components, which contribute hardly to efficiency (see (10.15)).

10.2.4 Non-data-aided maximum-likelihood TED

We now drop the assumption that the timing-recovery subsystem has knowledge of the data a_k. Appendix 10A.3 derives the corresponding ML TED. To keep derivations

manageable, the appendix restricts attention to systems with uncorrelated binary data $a_k \in \{-1, 1\}$ and a flat folded signal-to-noise ratio. The principal results of the analysis do, however, apply more generally. Two outcomes are particularly noteworthy.

1. At SNRs that are high enough to support somewhat reliable bit-decisions, the NDA ML TED produces its own decisions $\hat{a}_k$ so as to become a DA ML TED in decision-directed disguise.

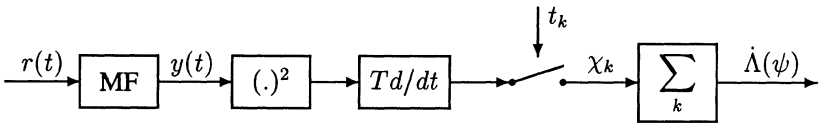


Figure 10.6: Approximate NDA ML timing-error detector for systems with uncorrelated binary data and a very low flat folded signal-to-noise ratio.

2. Only at SNRs that are too low to be practical does a real distinction between DA and NDA emerge. For uncorrelated binary data $a_k \in \{-1, 1\}$ and a channel with a flat folded signal-to-noise ratio one arrives at the approximate ML TED of Fig. 10.6. The discrete-time integration in Fig. 10.6 is effectively performed by the tracking loop in which the TED is embedded (compare Section 10.2.1). For this reason the sampled derivative χ_k should be regarded as the actual TED output. The squaring circuit of Fig. 10.6 is central to an important practical category of synchronizers, viz. the *squaring loops*. These are discussed in Section 10.4.

10.3 TIMING RECOVERY BASED ON THRESHOLD CROSSINGS

In this classical category of schemes, the instants at which the signal crosses a fixed threshold are used to derive a timing tone to which a VCO is locked. The involved operations are simple and well-matched to IC technology. The schemes can be tailored to various modulation codes and to uncoded transmission, and are applicable in full-response as well as partial-response systems. They are non-data-aided so that acquisition is not hampered by decision errors. Tracking properties can be quite respectable, especially when excess bandwidth is not too small. These combined features explain in part why schemes in this category have long been popular, and continue to be so. In this section we present a sample of the many tastes that have evolved (see, for example, [23], [55], [62], [13] [54], [17], [53], [47], [32], [34], [9] and [57]). We first discuss a representative scheme that is meant for uncoded full-response signaling, and proceed to analyze its properties (Sections 10.3.1 to 10.3.6). Alternative timing-error detectors are presented in Section 10.3.7. The scope is then expanded to run-length-limited modulation codes with peak detection, to partial-response systems, and to the impact of deterministic data patterns that may form part of a clock run-in pattern (Sections 10.3.9 to 10.3.11). Interspersed throughout this survey are various techniques that

are meant to facilitate acquisition, such as false-lock detection (Section 10.3.3), phase-and-frequency detection (Section 10.3.8), and zero-phase restart (Section 10.3.10).

10.3.1 An all-binary delay-tracking loop

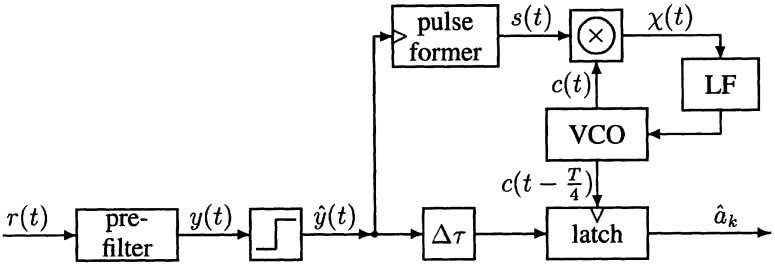


Figure 10.7: Full-response receiver with binary delay-tracking loop based on zero crossings.

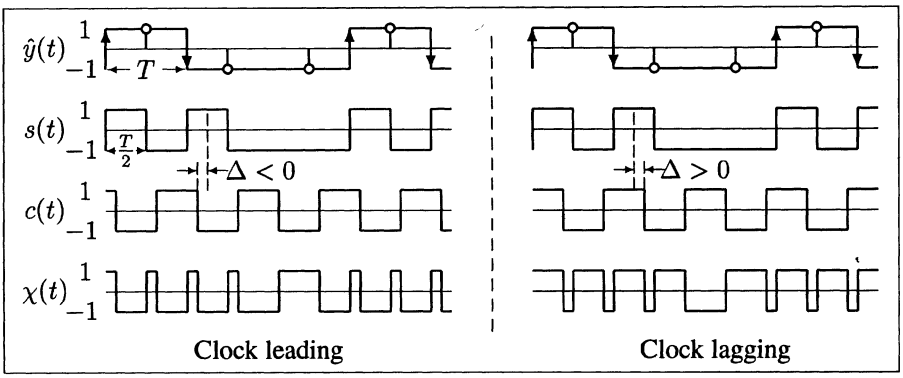


Figure 10.8: Example waveforms in the delay-tracking loop of Fig. 10.7.

Fig. 10.7 depicts a full-response receiver in which clock recovery and decisions are both based on a single logical signal $\hat{y}(t)$. To facilitate the exposition we will first consider a prefiltered signal $y(t)$ in the form of an ideal NRZ signal without noise. For this signal, the binary slicer in Fig. 10.7 has no effect, i.e. $\hat{y}(t) = y(t)$. Fig. 10.8 shows example waveforms. The upgoing and downgoing transitions of $\hat{y}(t)$ occur exactly halfway between the ideal decision instants (which are indicated by open circles), and thereby convey much timing information. To extract this information, the pulse former in Fig. 10.7 produces a pulse of width $T/2$ on every transition of $\hat{y}(t)$. The output $s(t)$ of the pulse former may be regarded as a *timing tone* that is modulated in frequency but not in phase. Specifically, every transition of $\hat{y}$ initiates a full cycle of a square wave at the bit frequency $(1/T)$. Conversely, $s(t)$ contains only DC ($s(t) = -1$) during T seconds when such a transition does not occur. To extract the phase of the timing tone,

$s(t)$ is multiplied with a binary clock signal $c(t)$ at the bit frequency. This is, in effect, a form of coherent demodulation as in Appendix 9B. The resulting signal $\chi(t)$ is a square wave at the bit frequency with a time-varying duty cycle. Every transition of $\hat{y}$ causes the duty cycle of this wave to deviate from 50% during T seconds. The deviation is proportional to the phase of the clock relative to that of the timing tone. As a result of this deviation, the DC content of $\chi(t)$ comes to differ from zero. It is this DC content that serves as the timing error information which drives the tracking loop. The loop will settle down at a clock phase for which $\chi(t)$ contains no DC. Then $\Delta = 0$ (Fig. 10.9). The decision instants are given by the downgoing transitions of a quadrature VCO output $c(t - T/4)$; decisions are taken with the aid of a latch (see Fig. 10.7). The decision instants will not be correct if there is a delay difference $\Delta\tau$ between timing recovery and data paths (see also Section 9.2.2).

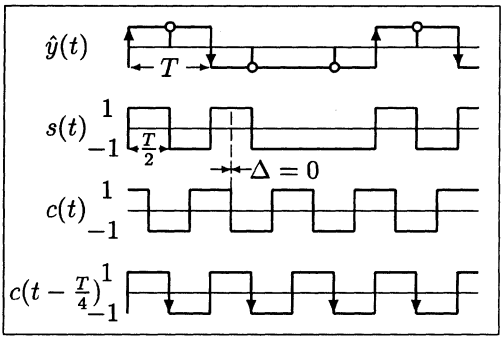


Figure 10.9: Example waveforms in the loop of Fig. 10.7 at the ideal sampling phase.

The loop of Fig. 10.7 is based on continuous-time logical operations, which are easily implemented with digital IC technology. Digital logic normally uses the levels 0 and 1 instead of -1 and $+1$. As a result, the binary multiplier of Fig. 10.7 degenerates into an exclusive-or circuit.

A variant of the loop of Fig. 10.7 is sometimes implemented in the discrete-time domain (see, for example, [61]). The signal $\hat{y}(t)$ is oversampled by a factor L that can be as large as 16 or 32. This gives 16 or 32 discrete sampling phases per symbol interval. During every symbol interval T, one of these phases is selected by the VCO and demarcates the decision instants. The VCO is typically constructed with a modulo counter, whose increments are determined by the control voltage at the output of the loop filter. The action of pulse former, binary multiplier, loop filter and VCO may be intertwined in a clever fashion so as to reduce complexity [37].

10.3.2 TED transfer characteristic

The TED output $\chi(t)$ in Fig. 10.7 is averaged out by the tracking loop. The DC component of $\chi(t)$ constitutes the TED transfer characteristic $\rho(\Delta)$. From Fig. 10.8

one easily recognizes that ρ is a linear measure of the sampling phase error Δ as long as the clock phase is within $T/4$ seconds from the ideal value, i.e. as long as $|\Delta| < 0.25$. Beyond this range we may make use of the fact that a shift of the clock phase over $T/2$ seconds corresponds to an inversion of the clock polarity. Hence $\rho(\Delta+0.5) = -\rho(\Delta)$ for any Δ. As a result, ρ is a triangular function of Δ (Fig. 10.10). This transfer characteristic may also be determined analytically (see Appendix 10B). The derivation is of didactical interest, and is useful to assess the effect of certain imperfections.

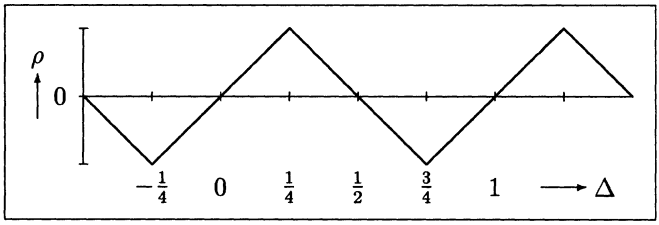


Figure 10.10: TED transfer characteristic for the delay-tracking loop of Fig. 10.7.

10.3.3 False-lock detection

A potential problem of the TED characteristic of Fig. 10.10 is that it exhibits zero-crossings not only at the ideal sampling phase, but also $T/2$ seconds away. The latter crossings cause an unstable equilibrium point halfway between the desired sampling instants at which the loop may dwell for some time. To avoid such a *false lock*, a quadrature branch may be added to the TED (Fig. 10.11). In this branch, $s(t)$ is multiplied by the quadrature clock signal $c(t-T/4)$. The transfer characteristic for the quadrature branch is a quadrature version of the in-phase characteristic, and its output will be maximal whenever the in-phase output vanishes. At this point, the sign of the quadrature output provides a false/correct lock indication.

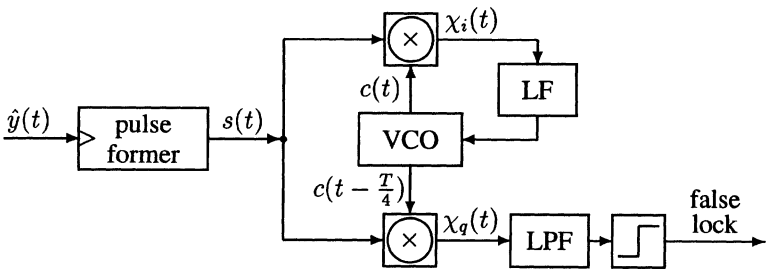


Figure 10.11: A quadrature branch may be added to the TED of Fig. 10.7 to be able to detect false locks.

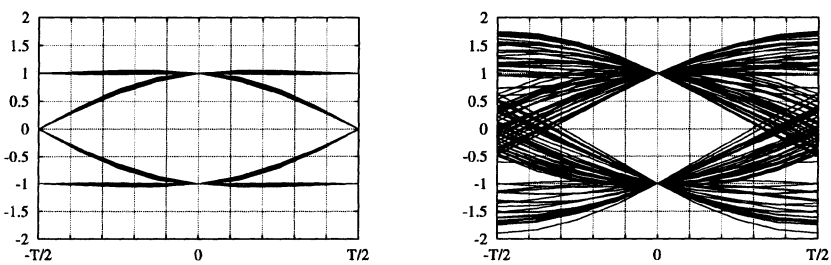
The quadrature configuration of Fig. 10.11 can also be used to expand the linear range of the TED transfer characteristic. By way of illustration we assume that $c(t)$ is a sinusoidal clock signal. Then the in-phase and quadrature outputs $\chi_i(t)$ and $\chi_q(t)$ are a

measure of $\sin 2\pi\Delta$ and $\cos 2\pi\Delta$, respectively (see Problem 10.18). Thus $\chi_i(t)+j\chi_q(t)$ will be a measure of $e^{j2\pi\Delta}$. By analogy to Section 9B.2, an unbiased estimate of Δ may be formed by taking the argument of $\chi_i(t) + j\chi_q(t)$. This leads to the sawtooth TED transfer characteristic of Fig. 9B.3. The jumps in this characteristic are due to the fact that the argument of a complex phasor can only be determined modulo 2π. Just as in Section 9B.2, any filtering before the argument operation is of the planar type and helps to reduce hangup [46]. Filter bandwidth is restricted by the considerations of Appendix 9B.2.

10.3.4 Pattern-dependent jitter

At the ideal sampling phase ($\Delta = 0$), the instantaneous output $\chi(t)$ of the TED of Fig. 10.7 is a square wave with frequency $2/T$ or $1/T$, depending on whether a transition has occurred or not. In both cases the duty cycle is 50%. Thus the average of $\chi(t)$ across a full period of $T/2$ or T seconds is zero, irrespective of the data sequence a_k. Since the loop effectively averages $\chi(t)$ across a considerably larger time interval, pattern-dependent jitter will be negligible. The only condition is that all zero-crossings of $y(t)$ in Fig. 10.7 must occur exactly halfway between the desired sampling instants. This will ensure that $\hat{y}(t)$ is an ideal NRZ signal as in Figs. 10.8 and 10.9. At a first glance it might seem that $y(t)$ must have infinite bandwidth in order to qualify. This is, however, untrue.

Example 10.5 Let us assume that the system response $q(t)$ up until the prefilter is a raised-cosine pulse with roll-off factor β. Fig. 10.12 depicts corresponding eye patterns. Zero crossings occur exactly halfway between the ideal sampling instants for $\beta = 1$. This shows that 100% excess bandwidth suffices for pattern-dependent jitter to vanish. Problems 10.11 to 10.12 reveal that a smaller excess bandwidth will not do. Zero crossings become diffuse as β drops below 1. Fig. 10.13 illustrates how this affects the timing tone $s(t)$. Transitions of $s(t)$ occur at well-defined positions for $\beta = 1$, but exhibit jitter when β drops below 1. As a result, pattern-dependent jitter will arise.



$\beta = 1$ $\beta = 0.25$

Figure 10.12: Eye patterns for a channel without noise that has been equalized into a raised-cosine characteristic with roll-off parameter β.

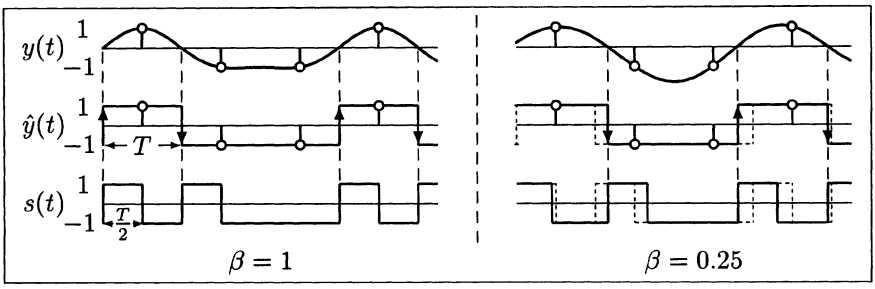


Figure 10.13: Example waveforms in the loop of Fig. 10.7 for two roll-off parameters β.

An analysis of data-induced zero-crossing jitter and the consequential pattern-dependent jitter may be found in [55] and [47]. For symmetric system responses $q(t)$, these papers show that zero-crossing jitter has a power spectral density with a depression near DC. This depression gets deeper as excess bandwidth increases, and pattern-dependent jitter decreases accordingly. Because of this depression, noise tends to be the primary source of jitter of the decision instants, except possibly at high SNRs.

Exercise 10.4 In Fig. 10.13, it can be observed for $\beta = 0.25$ that the first zero-crossing occurs too late, while the next one occurs too early. Does it happen frequently that two successive zero-crossings are shifted in opposite directions? What effect does this have on the PSD of the zero-crossing jitter and consequently on pattern-dependent jitter?

10.3.5 Noise-induced jitter

Noise may cause the zero crossings of $y(t)$ to shift. These shifts, in turn, induce jitter. We now analyze this effect for the system of Fig. 10.7 in the event that pattern-dependent jitter is negligible. At the output of the prefilter we have a signal

$$y(t) = \sum_{k=-\infty}^{\infty} a_k q(t - kT) + n(t), \tag{10.18}$$

where $q(t)$ is the system response and $n(t)$ is noise. Every change of polarity of the data sequence a_k causes a zero crossing of $y(t)$. These transitions may be characterized by a *transition* sequence $b_k \triangleq (a_k - a_{k-1})/2$. Clearly $b_k = 0$ when there is no transition between instants $k-1$ and k (i.e. when $a_{k-1} = a_k$), and $b_k = \pm 1$ according to the direction of a transition whenever $a_{k-1} \neq a_k$.

Let us consider an upgoing transition of $y(t)$ between the instants $(k-1)T$ and kT. Then $a_{k-1} = -1$, $a_k = 1$ and $b_k = 1$. Since pattern-dependent jitter is negligible, $y(t)$ would cross zero at the instant $\hat{t}_k \triangleq (k - 0.5)T$ if noise were absent (Fig. 10.14). Because of noise, the actual zero crossing occurs at an instant $t_k \neq \hat{t}_k$. The displacement $\tau_k \triangleq t_k - \hat{t}_k$, in turn, induces jitter. Provided that noise is not too large, τ_k may be estimated from the noise value $v_k \triangleq n(\hat{t}_k)$ at the instant $\hat{t}_k$ and the slope μ of the data

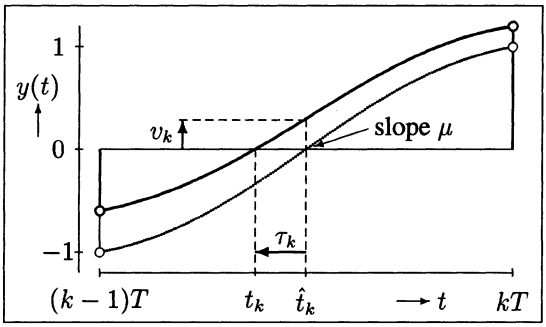


Figure 10.14: Upgoing transition of $y(t)$ (solid). The corresponding curve in the absence of noise is also depicted (dashed).

component of $y(t)$ at this same instant:

$$\tau_k \simeq -\frac{v_k}{\mu}. \tag{10.19}$$

It should be noted that μ may depend to some extent on preceding and subsequent data symbols a_m with $m \notin \{k-1, k\}$. To account for this dependence we may use an *average* slope μ in (10.19). This average slope is analyzed in Appendix 10B. The displacement τ_k induces the noise component of the TED output. It should be compared to the displacement ΔT between the ideal and the actual clock phase, which induces the desired component of the TED output. For down-going transitions of $y(t)$ this desired component does not change, but τ_k will be inverted with respect to (10.19) (this is so because we take μ to be the slope for a $-1 \rightarrow 1$ transition, i.e. μ is always positive.) In the absence of a transition, there is no contribution at all to the TED output. To account for these data dependencies we may multiply τ_k by b_k and ΔT by b_k^2. This leads to the equivalent phase-domain TED model of Fig. 10.15. This model neglects the scaling factor that is involved in converting temporal displacements to TED output voltages. This factor affects the desired and undesired components of the TED output equally, and is therefore of no concern to TED efficiency (see Section 9.3.3).

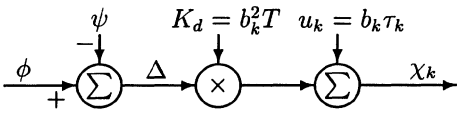


Figure 10.15: Equivalent phase-domain model of the TED of Fig. 10.15.

Consider the case that there is no transition between the instants $k-1$ and k. Then $y(t)$ has no zero crossing, and in the system of Fig. 10.7 there will be no contribution to the TED output. In the model of Fig. 10.15 this is reflected in that b_k and b_k^2 are both zero, so that $\chi_k = 0$. Conversely, when there is a transition, the TED output $\Delta b_k^2 T =$

ΔT will reflect the sampling phase error Δ, but will also contain a noise component $u_k = b_k \tau_k = -b_k v_k/\mu$. By analogy to Section 9.3.3, this leads to a TED efficiency

$$\gamma_{zc} \triangleq \frac{1}{\mathrm{SNR}} \frac{\overline{K_d}^2}{\mathcal{U}(1)}, \tag{10.20}$$

where $\overline{K_d} \triangleq E[b_k^2]T$ is the effective TED gain, and $\mathcal{U}(1)$ is the power spectral density of u_k at DC ($e^{j2\pi\Omega} = 1$).

Example 10.6 Assume that noise v_k is uncorrelated and has variance σ_v^2. Then u_k is uncorrelated and has variance $\sigma_u^2 = \mathcal{U}(1) = E[b_k^2]\sigma_v^2/\mu^2$. Correspondingly $\gamma_{zc} = (\mu T)^2 E[b_k^2]/(\mathrm{SNR}\sigma_v^2)$. This suggests that efficiency can be optimized in two directions:

1. Data a_k can be selected to have as many transitions as possible. This will maximize $E[b_k^2]$. The use of a maximum run-length constraint is beneficial in this respect. Upon average, uncoded binary data has one transition every two symbol intervals so that $E[b_k^2] = 0.5$. By comparison, the highest possible transition density $E[b_k^2] = 1$ occurs when there is a transition every symbol interval, i.e. for the Nyquist-rate data pattern $\ldots + - + - + - \ldots$. This shows that coding can only yield modest improvements of γ_{zc} relative to uncoded transmission.

2. The slope μ can be increased. The limiting case is that of a system response $q(t)$ in the form of a rectangular NRZ pulse. Such a pulse has infinite slope μ, i.e. any *finite* amount of additive noise $n(t)$ cannot displace the zero crossings at all. Thus a single data transition would suffice to identify the desired sampling phase to infinite precision. Unfortunately, infinitely steep transitions require an infinite bandwidth and this is clearly unphysical. Moreover, as bandwidth grows, noise enhancement tends to increase rapidly to the point where reliable data detection becomes impossible (see Chapter 5). For this reason there is generally not much latitude to improve μ.

Example 10.7 The idealized optical recording channel of Example 10.4 may be equalized into a raised-cosine characteristic. Corresponding efficiencies for uncorrelated binary data a_k may be determined via (10P.7), and are compared in Fig. 10.16 with those for maximum-likelihood timing recovery.

Bandwidth of the channel decreases rapidly as the normalized information density S increases. Noise enhancement of the equalizer increases accordingly, particularly if the target response has much excess bandwidth. As a result, efficiency degrades most rapidly at high densities for $\beta = 1$. For densities up to around 1.5, a roll-off factor between 0.25 and 0.5 will keep efficiencies within 5 dB from the optimum. Beyond that point the loss quickly becomes unacceptably large. For the sake of completeness we note that pattern-dependent jitter starts to appear as β drops below 1. For this reason the tracking performance predicted by the $\beta = 0.25$ and $\beta = 0.5$ curves is not strictly attainable, but can at best be approached.

10.3.6 Effect of transition jitter on TED transfer characteristic

In Section 10.3.1 we have seen that every transition of $\hat{y}(t)$ induces a well-defined contribution to the output of the TED. Ideally these individual contributions are all phase-aligned, and together lead to the triangular TED transfer characteristic of

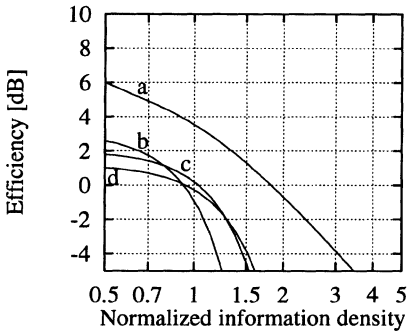


Figure 10.16: Efficiencies for optical recording channel of Example 10.4 with uncorrelated binary data. a: Maximum-likelihood timing recovery (γ_{ML}); b-d: timing recovery based on zero crossings (γ_{zc}) when the channel is equalized to a raised-cosine characteristic with roll-off factor $\beta = 1$ (b.), $\beta = 0.5$ (c.) and $\beta = 0.25$ (d.).

Fig. 10.10. In practice, however, transitions exhibit random displacements under the influence of both data and noise (see Sections 10.3.4 and 10.3.5). As a result, individual contributions are no longer perfectly phase-aligned. This leads to a 'smoothing' of the TED transfer characteristic that is studied in Appendix 10B.3. The analysis reveals that while the shape of the transfer characteristic tends to change rather strongly, its slope in the origin degrades only marginally for practical amounts of transition jitter. For this reason tracking properties will remain essentially unaffected.

10.3.7 Alternative TED structures

The loop of Fig. 10.7 represents just one of the many ways to extract timing information from the zero crossings of a signal $y(t)$. By way of illustration we now discuss two other timing-error detectors. Tracking properties of the corresponding schemes are similar the ones above, and their analysis can proceed along the same lines. Further TED tastes may be found, for example, in [17] and [54].

Sampling TED [55, 59] This TED generates an error sample χ_m each time the signal $y(t)$ crosses zero. The amplitude of χ_m is proportional to the temporal displacement of these zero-crossings with respect to those of the clock signal $c(t)$ (Fig. 10.17).

Upgoing and downgoing transitions of $\hat{y}(t)$ define the instants t_m at which $c(t)$ is sampled. The TED transfer characteristic has the same shape as the clock signal. In Fig. 10.18, $c(t)$ is assumed to have a sawtooth shape so as to maximize the linear range of the transfer characteristic. The clock is lagging with respect to the data, and all samples χ_m are positive. Their polarity would reverse if the clock were leading.

An efficient discrete-time implementation of the loop of Fig. 10.17 for a system with run-length-limited modulation code is described in [59]. Problem 10.21 is concerned with an application to Bi-phase.

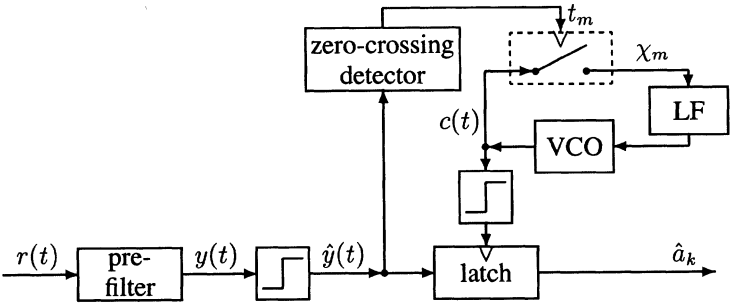


Figure 10.17: Delay-tracking loop with sampling TED.

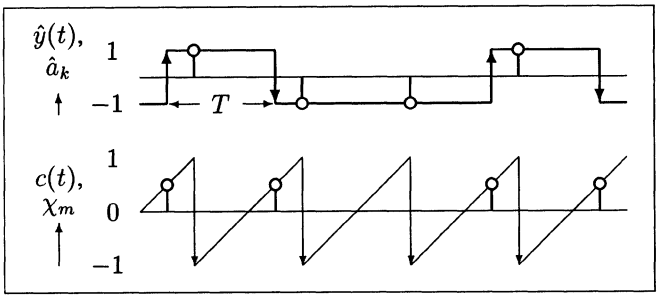


Figure 10.18: Example waveforms in the delay tracking loop of Fig. 10.17. Decision instants are determined by the downgoing clock transitions.

In practice the sampler of Fig. 10.17 is often replaced by a track-and-hold circuit. Instead of a discrete-time TED output χ_m this yields a continuous-time output $\chi(t)$ that changes only at the instants t_m. In effect, the hold operation forms part of the action of the loop filter (see also Problem 10.14).

The TED may be simplified by using a square wave for $c(t)$ so that the sampler in Fig. 10.17 degenerates into a D-type flip-flop. The TED output then becomes binary so that it still conveys information about the sign of the timing error but no longer about its magnitude. As a result it becomes difficult to assign a gain K_d to the TED and this complicates loop design and analysis.

Delay-difference TED [32, 34] In this TED, the incoming signal $\hat{y}(t)$ passes through two successive D-type flip-flops (Fig. 10.19). The signal is latched into these flip-flops at the upgoing and downgoing edges of the VCO clock $c(t)$, respectively. That clock is a square wave with a duty cycle of 50 percents. Its frequency coincides with the bit rate $1/T$, i.e. $c(t)$ has one upgoing and one downgoing transition every symbol interval T. The effective delay that the incoming transitions experience in passing through the first flip-flop depends on the location of these transitions relative to those of $c(t)$. The delay through the second flip-flop, by comparison, is exactly $T/2$ seconds because the input

of that flip-flop has already been retimed with respect to the VCO. A comparison of both delays yields a timing-error indicator $\chi(t)$. To this end, data input and output of each flip-flop are compared via an exclusive-or circuit, and the outputs of both exclusive-or circuits are differenced. This yields the desirable sawtooth-shaped transfer characteristic of Fig. 10.20. A further advantage is that the recovered data is available at the output of the second flip-flop, i.e. that the detection and clock-recovery paths are largely merged. This restricts delay mismatches between both paths that might otherwise have to be compensated for.

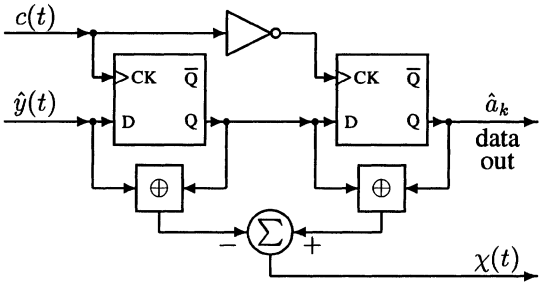


Figure 10.19: Delay-difference TED.

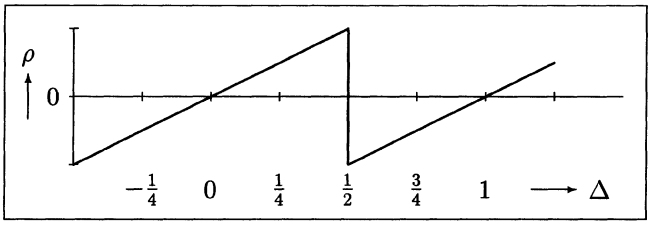


Figure 10.20: Transfer characteristic $\rho(\Delta)$ of the TED of Fig. 10.19.

10.3.8 Phase-and-frequency detectors

The foregoing TEDs all have a periodic transfer characteristic and their average output in the face of frequency errors is zero. It is sometimes possible to modify the TED in such a manner that it also responds to frequency errors. This usually involves the introduction of some form of hysteresis, and the resulting structure is said to be a *frequency-and-phase detector* (PFD). PFDs based on the TED of Fig. 10.19 and on the binary variant of the TED of Fig. 10.17 are developed in [56]. Another PFD for the latter binary TED is described in [49]. By way of illustration we discuss in this section a PFD that is used, among other purposes, for peak detection of run-length-limited data in digital magnetic recording. The origins of this circuit can be traced back to [62] and [13]. Improvements of it are described in [19] and [15].

Peak detection in digital magnetic recording In Section 2.3.5 we have seen that replay heads basically respond to recorded transitions at low recording densities. To account for this fact the data sequence is sometimes represented in so-called NRZI (non-return-to-zero-inverse) notation, i.e. a 1 denotes a transition (in either direction), and a 0 indicates no transition (see also Section 4.8). By way of illustration Fig. 10.21 depicts an NRZI data sequence a_k, the corresponding write signal $s(t)$, and the resulting replay signal $r(t)$. Clearly $s(t)$ has a transition whenever $a_k = 1$, while no transition occurs when $a_k = 0$. Every transition of $s(t)$ induces a pulse in the replay signal $r(t)$. Interference between successive pulses may be restricted by imposing a minimum run-length constraint on the recorded data. In Fig. 10.21, transitions of $s(t)$ are spaced at least two symbol intervals apart. In the nomenclature of Section 4.8 this corresponds to a minimum-run-length parameter $d = 1$.

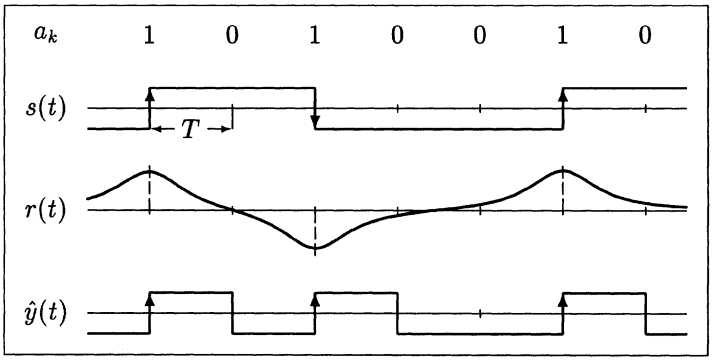


Figure 10.21: Illustration of peak detection in digital magnetic recording.

Data recovery may be accomplished via a *peak detector*, which basically locates the peaks of $r(t)$. The output of this detector is a logical signal $\hat{y}(t)$ that contains a rectangular pulse of width T for every detected peak. The upgoing edges of these pulses are aligned with the locations of the peaks. Intersymbol interference and noise together cause the peaks of $r(t)$ to be shifted with respect to their ideal locations. This phenomenon is called *peak shift*. A delay-tracking loop is needed to extract a clock signal that is phase-aligned to the average location of the peaks. The combination of such a loop and a bit detector is called *data separator* in disk drive jargon.

Data separator with PFD The signal $\hat{y}(t)$ of Fig. 10.21 is similar in nature to that of Figs. 10.8 and 10.9. The main difference is that successive pulses of width T are nominally separated by at least one symbol interval as a result of the minimum-run-length constraint. This fact is exploited in the PFD of Fig. 10.22.

The PFD contains two D-type flip-flops whose inverting outputs serve as outputs 'up' and 'down' of the TED. A logical AND circuit operates on the non-inverting outputs in such a manner as to ensure that 'up' and 'down' cannot be active jointly. As a result, the PFD can be in one of three states: 'up' (+1), 'down' (−1) and 'inactive' (0).

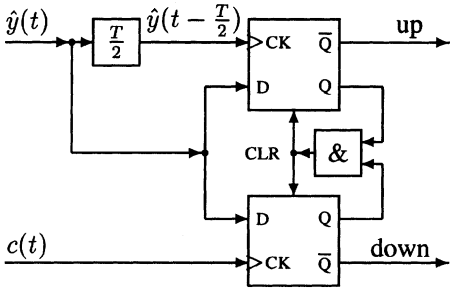


Figure 10.22: Phase-and-frequency detector (PFD) configured for operation on the logical signal $\hat{y}(t)$ of Fig. 10.21.

In the example waveforms of Fig. 10.23, these states are represented in the form of a single ternary signal $\chi(t)$. The PFD operates on the clock $c(t)$ and on a delayed version $\hat{y}(t-\frac{T}{2})$ of $\hat{y}(t)$. It is basically a ternary counter with outputs -1, 0, and $+1$. Upgoing transitions of $\hat{y}$ and c cause the counter to be increased and decreased by one step, respectively. In the absence of valid clock information (i.e. when $\hat{y}(t)$ is zero) the counter is forced to zero. Frequency errors between $\hat{y}$ and c will cause the counter to alternate between 0 and one of both extreme positions, depending on which frequency is largest. As a result, the average PFD output is proportional to the frequency error. This permits the timing-recovery loop to acquire more rapidly in the face of initial frequency errors than with a TED that only responds to phase errors.

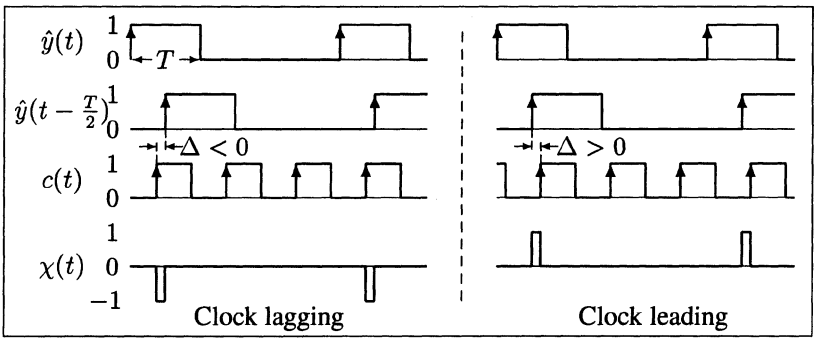


Figure 10.23: Example waveforms in the circuit of Fig. 10.22.

The output of the PFD is often connected to a *charge pump* [25]. This is basically a controllable current source with an associated capacitor that is charged, left untouched, or discharged depending on the three states that the PFD can be in. The net effect is that a voltage proportional to $\chi(t)$ in Fig. 10.23 is delivered to the loop filter. The PFD may be extended with provisions to improve acquisition in the face of frequency errors [19, 30]. Another extension pertains to a circuit for lock detection ([16],

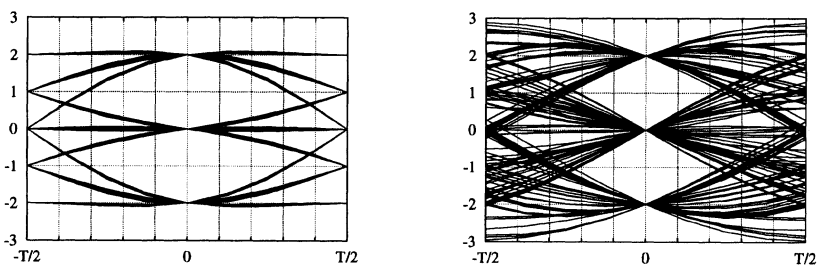
see Problem 10.22). It can be observed that only the upgoing transitions of $\hat{y}(t)$ are used to extract timing information. At a first glance, this might seem a disadvantage as compared to the schemes of Figs. 10.7 and 10.17. It should be noted, however, that the rectangular pulses of $\hat{y}(t)$ in Fig. 10.21 have a *fixed* width of T seconds. As a result, the downgoing transitions do not add any timing information to the upgoing transitions, and can be neglected without any performance penalty. Further properties of the circuit of Fig. 10.22 are explored in Problems 10.16 and 10.17.

10.3.9 Extension to partial-response reception

The foregoing techniques are easily extended to partial-response signaling. By way of example we consider Class IV. Here the prefiltered signal $y(t)$ satisfies

$$y(t) = \sum_{k=-\infty}^{\infty} b_k q(t - (k + \phi)T) + n(t), \tag{10.21}$$

where $b_k = a_k - a_{k-2}$ is the Class IV partial-response signal corresponding to the binary data a_k, $q(t)$ is a Nyquist-1 pulse and $n(t)$ is additive noise. Eye patterns are depicted in Fig. 10.24 for two raised-cosine pulses $q(t)$. Noise is assumed absent.



$\beta = 1$ $\beta = 0$

Figure 10.24: Eye patterns for a channel without noise that has been equalized to a Class IV raised-cosine characteristic with roll-off parameter β.

At the ideal sampling instants, the eye pattern has three possible values $b_k = -2, 0$ and 2. Since b_k may be zero for several symbol intervals in succession, zero crossings of $r(t)$ may be caused or strongly shifted by noise. Thus they do not convey much timing information. This precludes the use of a threshold at zero for timing recovery. A similar argument applies to thresholds at the other two nominal data levels. By contrast, thresholds at -1 and $+1$, i.e. midway between the nominal data levels, are only crossed by $r(t)$ in a transition from a given nominal data level to a different one. These crossings occur, upon average, halfway between the ideal decision instants. Threshold crossings all occur with a slope that differs significantly from zero. As a consequence, noise will

not shift the crossings significantly, and their location conveys a significant amount of timing information. For each threshold, a TED akin to that of Fig. 10.7 may be used. Both detectors have identical transfer characteristics, and an overall characteristic with twice the original amplitude is obtained by summing the individual outputs. By merging the common parts of both detectors we obtain the compound TED of Fig. 10.25. Similar compound TEDs may be based on the other TEDs above.

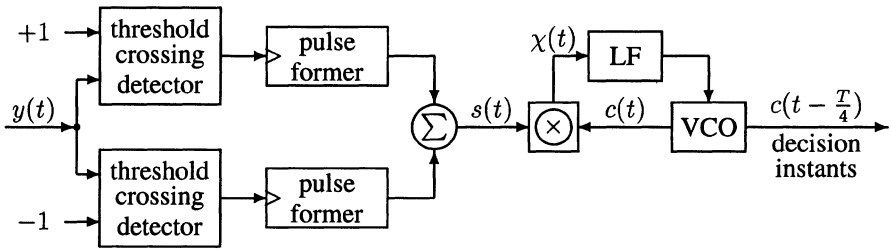


Figure 10.25: Delay-tracking loop with compound TED based on crossing of -1 and $+1$ levels in the eye patterns of Fig. 10.24.

For transitions between adjacent data levels (such as $+2$ and 0), only a single threshold is crossed. Conversely, the transitions $-2 \rightarrow +2$ and $+2 \rightarrow -2$ cause both thresholds to be crossed, thereby conveying a double amount of timing information. The first crossing occurs too early, and the second one too late (Fig. 10.24). The combined contribution of both crossings to the TED output is centered exactly halfway between the desired sampling instants and does not induce any pattern-dependent jitter. Similarly, any transition $0 \rightarrow 2$ or $0 \rightarrow -2$ must be followed in one of the next few symbol intervals by a transition in the opposite direction. Upon average, the first transition occurs somewhat too late, and the second one somewhat too early. This further reduces pattern-dependent jitter. It should be noted that this effect depends critically on the absence of phase distortion. In general, the performance of threshold-crossing-based timing recovery schemes is quite sensitive to asymmetries in the equalized system impulse response [55].

Upon average, there are 0.75 threshold crossings per symbol interval T. This is 50% more than for full-response signaling. A further advantage over full-response signaling is that the eye width degrades less dramatically as excess bandwidth vanishes. In fact, even without excess bandwidth the eye width at the threshold levels is still in the vicinity of 40%. This is distinctly favorable to the complete breakdown that is encountered for the squaring loops of Section 10.4 as excess bandwidth vanishes. We refer back to Fig. 9.15 for an illustration of the acquisition characteristics.

10.3.10 Zero-phase restart

The actual data sequence is often preceded by a preamble that serves, among other purposes, to facilitate acquisition. It is not uncommon to lock the PLL to an external

reference clock at the beginning of the preamble. This forces the VCO to run at a frequency which is very close to the bit frequency. At some point in time -perhaps after a certain number of transitions have been counted- the reference clock is disconnected, and the VCO is restarted in accurate phase alignment with the next data transition. This trick is sometimes called *zero-phase restart* and helps the PLL to acquire rapidly to the incoming data.

Exercise 10.5 Consider the loop of Fig. 10.7. Assume that the VCO is restarted with an upgoing transition of $c(t)$. How long after the data transition should this restart occur?

10.3.11 Effect of periodic data

Although we have so far restricted attention to random data, threshold-crossing-based timing recovery is equally applicable to other types of data. Of particular interest are periodic data sequences a_k as used in data preambles. For such sequences, the output of the channel will also be periodic when noise is absent, and threshold crossings occur with complete regularity. As a result, pattern-dependent jitter can be altogether absent.

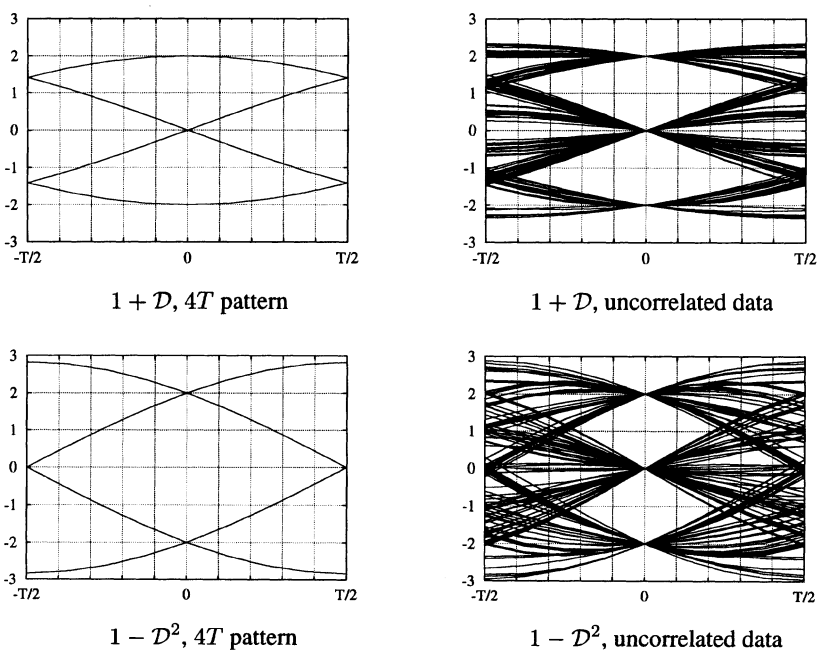


Figure 10.26: Eye patterns for duobinary $(1+\mathcal{D})$ and Class IV $(1-\mathcal{D}^2)$ partial-response systems without excess bandwidth. Noise is absent. Data a_k is binary; both the $4T$ pattern $\ldots + + - - + + - - \ldots$ and uncorrelated data are considered.

By way of example we consider the binary sinusoid $\ldots + + - - + + - - \ldots$ of period $4T$ when used in Duobinary $(1+\mathcal{D})$ and Class IV $(1-\mathcal{D}^2)$ partial-response systems without excess bandwidth. Fig. 10.26 depicts corresponding eye patterns. Also depicted are eye patterns for uncorrelated binary data. For $1+\mathcal{D}$, the zero crossings in the $4T$ eye pattern coincide with the instants that are optimum for detection of random data. For $1-\mathcal{D}^2$, by comparison, the zero crossings occur halfway between these decision instants. Thus, while zero crossings can serve in both cases as a basis for clock acquisition with a $4T$ preamble, different phase jumps may be needed at the end of the preamble to arrive at the optimum decision instants, or, equivalently, at a suitable initial clock phase for the timing recovery technique that is used beyond the preamble.

Further results in this context may be found in [40] and [50]. In particular for the $4T$ pattern, the zero crossings occur halfway between the optimum decision instants for $1\pm\mathcal{D}$ system responses, and are halfway in between for $1-\mathcal{D}^2$ and Nyquist-1 responses. Acquisition characteristics for $1-\mathcal{D}^2$ were illustrated in Fig. 9.16.

10.4 NONLINEAR SPECTRAL LINE METHODS

Received data signals do not normally have any content at the symbol frequency $1/T$. The basic idea behind nonlinear spectral line methods is to apply a nonlinearity in order to generate 'harmonics' of the received signal that include a timing tone at the symbol frequency. Timing information is extracted from this tone in a similar fashion as in Section 10.3. Nonlinear spectral line methods have long been popular because of their comparative simplicity; among the earliest references are [10] and [1]. Other noteworthy contributions are [21], [54], [39], [22] and [46]. In many instances the nonlinearity takes the form of a squarer. The corresponding timing recovery subsystem is said to be a *square-law delay-tracking loop* or just *squaring loop*. Because of their mathematical tractability, we will focus primarily on loops of this type. We will find that the performance of squaring loops depends critically on the amount of excess bandwidth. To a lesser extent, this remains true when other nonlinearities are used. It is worth noting that nonlinear spectral line methods all have a deductive nature. This renders delay differences between the timing recovery and detection paths a potential problem (see Section 9.2.2). This problem can be avoided in part by resorting to a digital implementation, of which Section 10.4.6 provides an example.

10.4.1 Basic square-law delay tracking loop

A starting point for this loop is the non-data-aided timing-error detector (NDA TED) of Fig. 10.6. This TED is optimum at very low SNRs if data is binary and the channel has a flat folded signal-to-noise ratio $\mathrm{FSN}(e^{j2\pi\Omega})$ (see Appendix 10A.3). The latter condition implies that the output $y(t)$ of the matched filter in Fig. 10.6 satisfies the Nyquist-1 criterion. To arrive at a tracking loop we feed the instantaneous estimates χ_k of the sampling phase error into a loop filter (LF), whose output controls the sampler via a voltage-controlled oscillator (VCO). The VCO output, of course, also controls

the sampler in the data path. Furthermore, we replace the matched filter by an arbitrary prefilter so as to account for the use of a variety of detection schemes. The resulting configuration is shown in Fig. 10.27. Corresponding noiseless eye patterns are shown in Fig. 10.28 for a prefilter in the form of a full-response linear equalizer. The equalized system response is a raised-cosine pulse with roll-off factor $\beta = 1$.

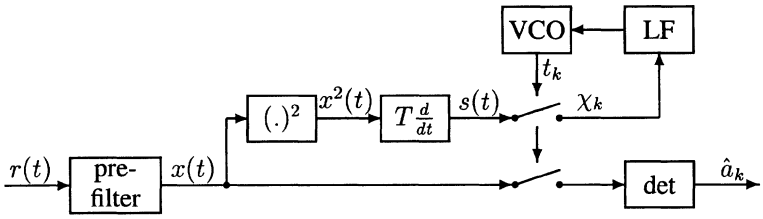


Figure 10.27: Basic square-law delay tracking loop.

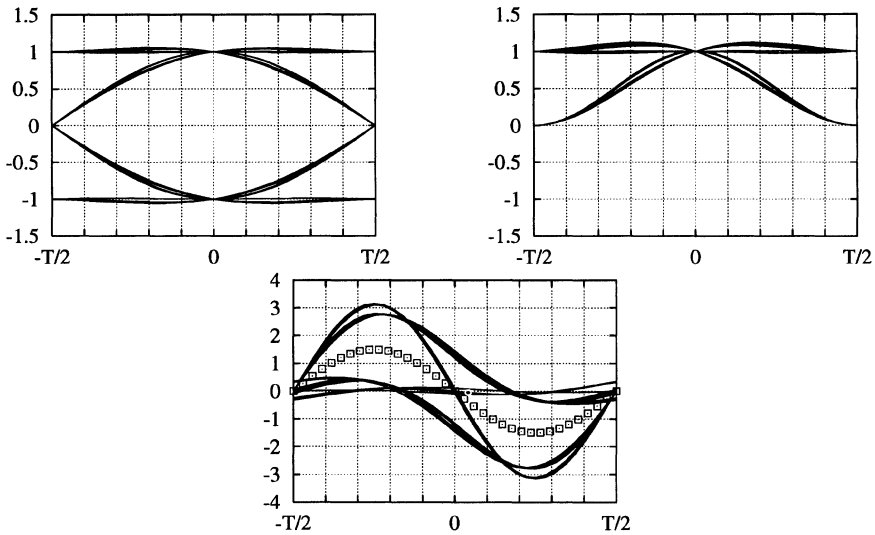


Figure 10.28: Eye patterns of three signals in the squaring loop of Fig. 10.27: $x(t)$ (top left), $x^2(t)$ (top right) and $s(t)$ (bottom; the squares indicate the average $\overline{s}(t)$).

At the ideal sampling instants, the output of the equalizer has no ISI, and all traces that make up the eye pattern have a more or less flat slope (Fig. 10.28, top left). This property is preserved by the squaring operation, which serves to make the signal unipolar (Fig. 10.28, top right). The differentiation transforms flat slopes into zero crossings (Fig. 10.28, bottom). Upon average, these occur at the ideal sampling instants and $T/2$ seconds away (see squares indicating $\overline{s}(t)$). Depending on the neighboring data symbols of a_k, the actual zero crossings may deviate from these average positions. As

a result, the output χ_k of the sampler will not be zero at the optimum sampling instants, and pattern-dependent jitter will arise. Any noise may displace the zero crossings still further. The output $s(t)$ of the differentiator is a cyclostationary timing wave with no DC. Its average $\overline{s}(t) \triangleq E[s(t)]$ may be regarded as a *timing tone* with period T that crosses zero at the optimum decision instants.

It is worth stressing that spectral lines arise beyond the squarer by virtue of the cyclostationarity of the data signal. Would $x(t)$ have been stationary then any nonlinear transformation of $x(t)$ would also be stationary and the only spectral line that could possibly emerge is one at DC.

10.4.2 TED transfer characteristic

The output χ_k of the sampler in Fig. 10.27 has a desired component $E[\chi_k]$ that constitutes the TED transfer characteristic $\rho(\Delta)$. Clearly

$$\rho(\Delta) = \overline{s}(t_k) = \overline{s}((k+\psi)T) = \overline{s}(\psi T) \tag{10.22}$$

where ψ denotes the VCO phase. The timing tone defines the shape as well as the amplitude of the TED transfer characteristic. This amplitude depends heavily on the excess bandwidth of the channel.

Example 10.8 The squarer in Fig. 10.27 essentially acts as a frequency doubler. The highest frequency components beyond the squarer are twice as high as those before it. For a channel without excess bandwidth, the output of the squarer has no data content at or above the symbol frequency $1/T$. This will also be true for the output $s(t)$ of the differentiator. Consequently $\overline{s}(t) = 0$.

This example shows that the squaring loop breaks down completely when excess bandwidth is absent. This is unlike the threshold-crossing-based technique of Section 10.3. To mitigate this problem, the squarer may be replaced by a fourth-power circuit [39, 20], a full-wave rectifier [10, 5], a ln cosh(.) circuit [22], or another symmetric nonlinearity.

We proceed to analyze the timing tone $\overline{s}(t)$ mathematically. Since $\overline{s}(t)$ is periodic with period T, it can be expanded into a two-sided Fourier series

$$\overline{s}(t) = \sum_{n=-\infty}^{\infty} s_n e^{j2\pi nt/T}. \tag{10.23}$$

The Fourier coefficients s_n describe the complex-valued amplitude of the timing tone at the frequency n/T. Now $\overline{s}(t)$ is a differentiated version of a tone $\overline{x^2}(t) \triangleq E[x^2(t)]$ at the output of the squarer. Let us denote the Fourier coefficients of $\overline{x^2}(t)$ by c_n. The spectral counterpart of differentiation is a multiplication by $j2\pi n$ and therefore

$$s_n = j2\pi n c_n. \tag{10.24}$$

Clearly $s_0 = 0$, i.e. $\overline{s}(t)$ has no DC. Example 10.8 reveals that $s_n = 0$ for $|n| \geq 2$ if the channel has less than 100% excess bandwidth. The coefficients c_n are easily determined

for a squarer input $x(t)$ of the form

$$x(t) = \sum_{k=-\infty}^{\infty} a_k q(t - kT) \tag{10.25}$$

where a_k is a random data sequence with zero mean and $q(t)$ is the system response up until the squarer input. For this input, it is shown in Appendix 10C.1 that

$$c_n = \frac{1}{T^2} \int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega}) Q(\Omega) Q^*(\Omega - n) d\Omega, \tag{10.26}$$

where $\mathcal{A}(e^{j2\pi\Omega})$ denotes the power spectral density of a_k and $Q(\Omega)$ denotes the Fourier integral of $q(t)$. This expression shows that the magnitude of c_n is determined by the amount of overlap between $Q(\Omega)$ and its shifted and transposed version $Q^*(\Omega - n)$. For systems with a restricted bandwidth, such overlap will exist only for small values of $|n|$. Problem 10.24 reveals that the addition of a stationary noise signal $n(t)$ to $x(t)$ can only affect c_0, provided that $n(t)$ is not correlated with $x(t)$. A direct implication is that $n(t)$ does not affect the timing tone $\overline{s}(t)$. If $q(t)$ is a symmetric function of time such that $q(-t) = q(t)$ then $\overline{s}(t)$ has zero-crossings for $t = 0, \pm\frac{T}{2}, \pm T, \pm\frac{3T}{2}, \ldots$.

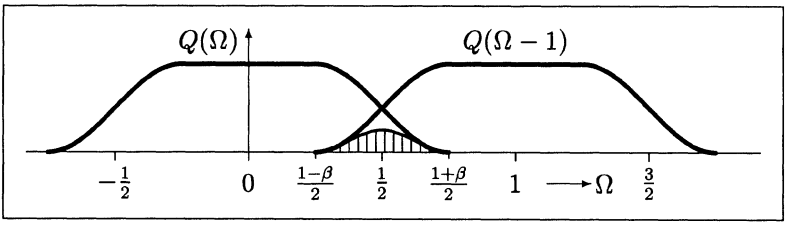


Figure 10.29: Raised-cosine characteristic $Q(\Omega)$ and its shifted version $Q(\Omega-1)$. The product of $Q(\Omega)$ and $Q(\Omega-1)$ is also depicted (thin). The shaded area determines the Fourier coefficient c_1 and thereby the amplitude of the timing tone.

Example 10.9 Let us assume that data a_k is uncorrelated and has unit power, and that $q(t)$ is a Nyquist-1 pulse from the raised-cosine family. This family has at most 100% excess bandwidth so that $c_n = 0$ for $|n| \geq 2$. Hence $s_n = 0$ for $|n| \neq 1$, i.e. the timing tone must be a pure sinusoid. Fig. 10.29 illustrates how the roll-off factor β affects c_1. The shaded area in Fig. 10.29 depends heavily on β, and becomes zero if there is no excess bandwidth ($\beta = 0$). In that case the timing tone will have zero amplitude, and the squaring loop will break down. This corroborates the conclusion of Example 10.8.

10.4.3 Jitter

Pattern-dependent jitter in the loop of Fig. 10.27 is induced by the data component η_k of χ_k. This component is analyzed in Appendix 10C.2. The analysis reveals that the power spectral density of η_k has a zero at DC whenever the system response $q(t)$ up to

the input of the squarer is a symmetric or antisymmetric function of time. This covers Nyquist-1 functions $q(t)$, most partial-response functions, and pulses appearing at the output of a matched filter. For responses of this type, pattern-dependent jitter is bound to be small. A similar statement seems to hold true for the fourth-power nonlinearity [39, 20].

Analysis of *noise-induced jitter* is complicated and would carry us too far afield. We refer to [21], [54] and [22] for results in this context. We have seen in Section 10.4.2 that the TED transfer characteristic has a small amplitude when excess bandwidth is small. In that case even modest amounts of noise will induce considerable amounts of jitter. For this reason, the squaring loop is best suited for systems with a comfortable excess bandwidth. Pre- and post-filters can be helpful to lower the effect of noise.

10.4.4 Squaring loop with prefilter and postfilter

Let us reconsider the raised-cosine system response $q(t)$ of Section 10.4.2. Frequencies outside the interval $\frac{1-\beta}{2} \leq |\Omega| \leq \frac{1+\beta}{2}$ do not contribute to the timing tone (compare Fig. 10.29). These frequencies may, however, contribute to pattern-dependent and noise-induced jitter. A band-pass prefilter centered around the Nyquist frequency $\Omega = 0.5$ may be used to suppress such jitter. Similarly, frequencies other than the symbol frequency ($\Omega = 1$) at the output of the squarer cannot contribute to the timing tone and may be suppressed by means of a band-pass postfilter centered around $\Omega = 1$. This leads to the topology of Fig. 10.30.

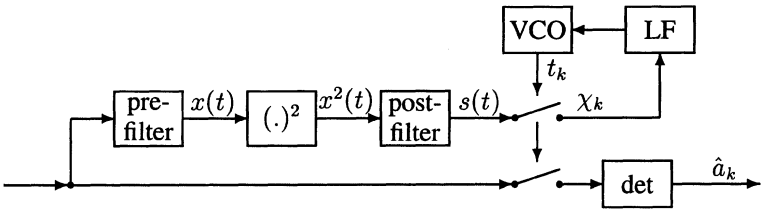


Figure 10.30: Bandpass pre- and post-filters may be added to the squaring loop of Fig. 10.27 in order to suppress pattern-dependent and noise-induced jitter.

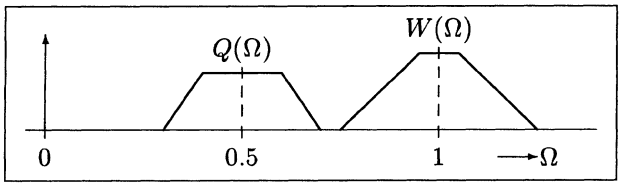


Figure 10.31: Pattern-dependent jitter can be eliminated via a prefilter that shapes the system characteristic $Q(\Omega)$ to be real-valued and symmetric about the Nyquist frequency. The transfer function $W(\Omega)$ of the postfilter must be real-valued and symmetric about the symbol frequency. There may be no spectral overlap between Q and W.

Let $Q(\Omega)$ denote the prefiltered system characteristic, i.e. the Fourier integral of the system response $q(t)$ up until the prefilter. Furthermore, let $W(\Omega)$ be the transfer function of the postfilter. Then $Q(\Omega)$ will be centered around the Nyquist frequency ($\Omega = 0.5$) and $W(\Omega)$ will be centered around the symbol frequency ($\Omega = 1$) (see Fig. 10.31). In Appendix 10C.3 it is shown that pattern-dependent jitter vanishes at the ideal sampling instants when $Q(\Omega)$ and $W(\Omega)$ are real-valued and symmetric about their center frequencies, provided that there is no spectral overlap between Q and W. By way of example we show in Fig. 10.32 corresponding eye patterns. The figure pertains to functions Q and W that have a raised-cosine roll-off about their center frequencies, and have equal bandwidth such that there is just no spectral overlap.

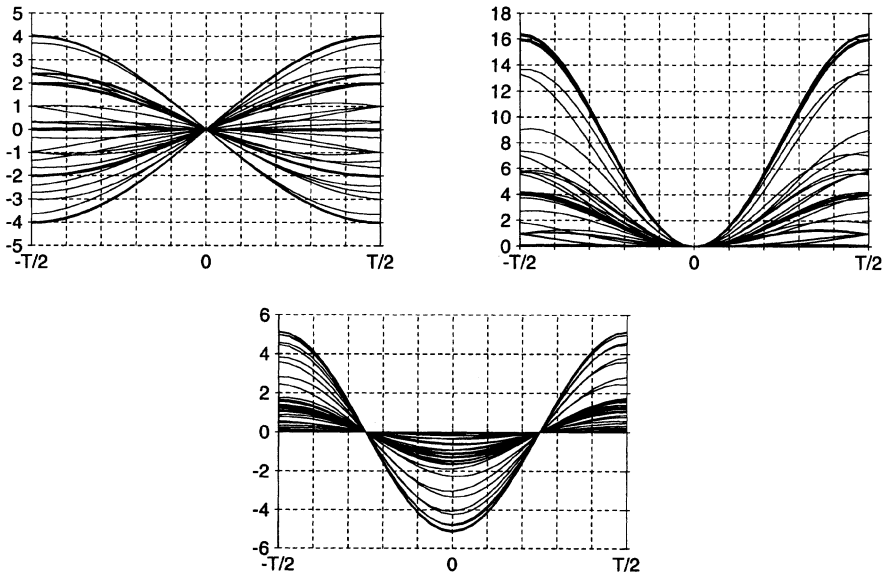


Figure 10.32: Eye patterns for a squaring loop with pre- and postfilter: prefilter output (top left), squarer output (top right), and postfilter output (bottom).

The prefilter output exhibits zero-crossings once every symbol interval T. These are preserved by the squaring operation. The postfilter fully suppresses DC and generates a timing wave with jitter-free $T/2$-spaced zero crossings that are at a fixed time offset ($T/4$) with respect to the desired sampling instants. Because of this offset, a quadrature output is needed in the VCO. Alternatively, the postfilter may be designed to incorporate a $\pi/2$ phase shift at the symbol frequency.

Detuning of the filters renders the zero crossings in Fig. 10.32 diffuse. For practical degrees of detuning, the resulting pattern-dependent jitter is likely to be modest when filter bandwidths are not too small [22]. When noise is modest as well, the zero-crossings in the output of the postfilter may directly drive the sampler in the data detec-

tion path of Fig. 10.30. In more demanding situations, a PLL remains needed for adequate jitter suppression. Noise at the output of the squarer is concentrated spectrally in the vicinity of the symbol rate $1/T$. It is suppressed in part by the postfilter, and for another part by the PLL. A noise analysis may be found in [21].

10.4.5 Timing-error detection via coherent demodulation

Sampling-phase information may be extracted from the timing wave $s(t)$ in several other manners than via the sampler of Fig. 10.27. For example, most of the techniques of Section 10.3 are applicable. It is also possible to use a coherent demodulator as in Appendix 9B (Fig. 10.33).

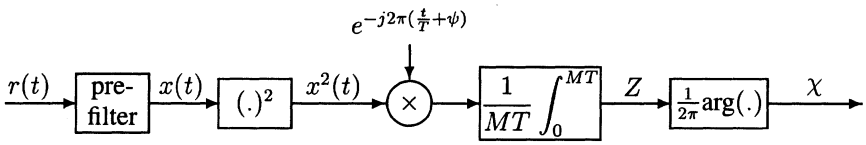


Figure 10.33: Timing-error detector for square-law timing recovery based on coherent demodulation.

The squarer output is multiplied by a complex-valued phasor $e^{-j2\pi(t/T+\psi)}$, where ψ represents the sampling phase that is used for detection (normalized in symbol intervals T). The aim is to shift the components of $x^2(t)$ at the symbol frequency $1/T$ towards DC. The phase information is then extracted by averaging across M symbol intervals T. The result is a variable

$$Z = \frac{1}{MT}\int_0^{MT} x^2(t)e^{-j2\pi(t/T+\psi)}dt = e^{-j2\pi\psi}\frac{1}{MT}\int_0^{MT} x^2(t)e^{-j2\pi t/T}dt. \quad (10.27)$$

The integral in this expression has expected value

$$\frac{1}{MT}\int_0^{MT} \overline{x^2}(t)e^{-j2\pi t/T}dt = \frac{1}{T}\int_0^{T} \overline{x^2}(t)e^{-j2\pi t/T}dt. \quad (10.28)$$

This may be recognized as the Fourier coefficient c_1 of the tone $\overline{x^2}(t)$ (see also (10.26) and Appendix 10C.1). Correspondingly $E[Z] = e^{-j2\pi\psi}c_1$.

Let us consider a system response $q(t)$ that is symmetric about $t = 0$. Then c_1 will be real-valued (see Problem 10.23). Furthermore, the optimum sampling phase will be zero, i.e. the sampling-phase error Δ just equals ψ. Thus

$$E[Z] = e^{-j2\pi\Delta}c_1. \quad (10.29)$$

This expression remains valid if the channel introduces an additional delay of ϕ symbol intervals (see Problem 10.25). Since c_1 is real-valued, an unbiased estimate χ of the sampling-phase error $\Delta = \psi - \phi$ may be formed by taking the argument of Z and

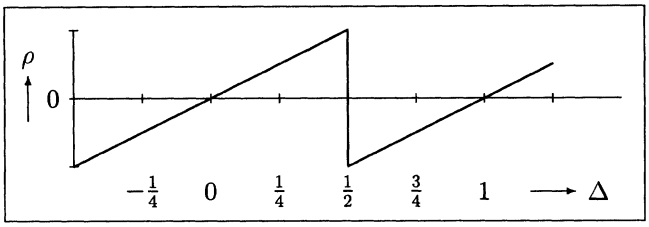


Figure 10.34: Transfer characteristic $\rho(\Delta)$ of the TED of Fig. 10.33.

scaling by $1/(2\pi)$. Because the argument of a complex phasor can only be determined modulo 2π, this estimate will exhibit jumps of unit magnitude for every symbol interval that Δ traverses. This leads to the sawtooth TED transfer characteristic of Fig. 10.34.

Exercise 10.6 What happens if $q(t)$ is antisymmetric about $t = 0$ rather than symmetric?

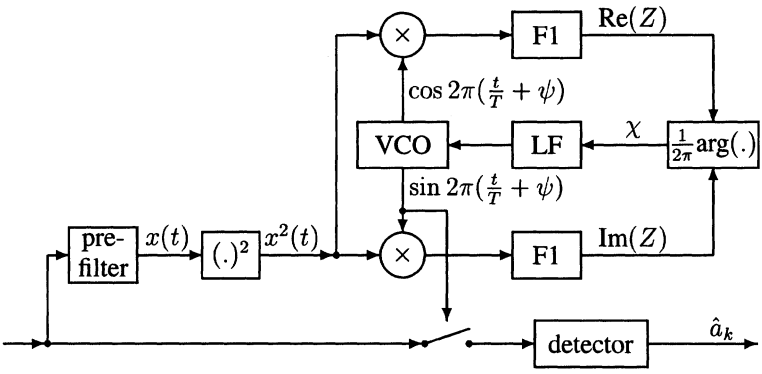


Figure 10.35: Square-law synchronizer based on the TED of Fig. 10.33.

A synchronizer based on the TED of Fig. 10.33 is shown in Fig. 10.35. To be able to track time-varying delays, the integrator is replaced by a filter F1 with a finite memory length. This filter is said to be *planar* because it operates on a complex-valued phasor. In Fig. 10.35, quadrature branches serve to separately determine the real and imaginary components of the filter output Z. Bandwidth of F1 is governed by the considerations of Appendix 9B.2.

The output χ of the TED controls the VCO via a loop filter (LF). When Δ is in the vicinity of any of the discontinuities in Fig. 10.34, the output of the arg-operator may jump up and down wildly, and the average TED output may be small even though the VCO phase ψ is nowhere near its desired value. As a consequence, the synchronizer may hang up at this erroneous phase for some time. This problem was already addressed in Appendix 9B.2, and the techniques to deal with it that were mentioned there are also applicable here.

In the tracking mode of operation, $|\Delta|$ will be small, whence $\arg(\Delta) \simeq -j2\pi\Delta$ and $E[Z] \simeq -j2\pi\Delta c_1$ (see (10.29)). The latter approximation suggests that we omit the cosine branch in Fig. 10.35, replace the arg-operator by a constant gain factor $1/(2\pi c_1)$, and combine the filters F1 and LF into a single loop filter. The resulting synchronizer is considerably simpler than the one of fig 10.35, but will have a poorer acquisition behavior.

10.4.6 Digital filter and square timing recovery

Nonlinear spectral line methods can also be implemented digitally. By way of example we discuss an all-digital full-response receiver [46, 8].

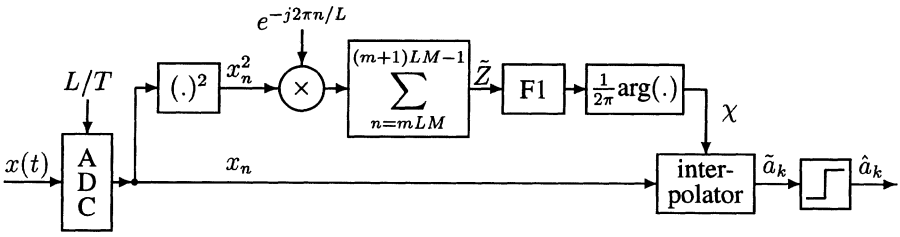


Figure 10.36: All-digital digital receiver with asynchronous input, squaring circuit with planar filtering, and interpolator for asynchronous-to-synchronous conversion.

In Fig. 10.36, an analog input signal $x(t)$ is converted into digital samples x_n at instants $t = nT/L$ that are demarcated by a free-running clock. Here L is an oversampling factor that remains to be determined. The sampled signal x_n is squared, and the component of x_n^2 at the symbol rate $1/T$ is extracted by means of coherent demodulation. To this end, x_n^2 is multiplied by a complex phasor $e^{-j2\pi n/L}$ and integrated across a block of M symbol intervals T. The result is a variable

$$\tilde{Z} = \sum_{n=mLM}^{(m+1)LM-1} x_n^2 e^{-j2\pi n/L} \tag{10.30}$$

where m is a block index. This may be recognized as the discrete Fourier transform of x_n^2 evaluated at the symbol frequency $1/T$. In the absence of aliasing distortion, $\tilde{Z}$ is equivalent to the variable Z of (10.27), and a measure χ of the sampling-phase error Δ is obtained by means of the arg-operator. A low-pass planar filter F1 is included before this operator in order to suppress jitter while avoiding hangup. The timing recovery scheme is of the open-loop type (see Section 9.2.2) and the planar filter can, therefore, not impede stability (see Appendix 9B.2).

In the data path, the input x_n is resampled under the control of χ. This is done by means of an interpolator with controllable phase so as to be able to handle fractional delays, i.e. delays of a fraction of T/L seconds (see Section 9.2.3). A simple linear

interpolator will do provided that L is sufficiently large (see Appendix 9A.2). Ideally, the output $\tilde{a}_k \triangleq x((k+\psi)T)$ of the interpolator is synchronous with the transmitted data. For the circuit of Fig. 10.36 to work properly, L must be large enough to avoid aliasing in the squarer. Because squaring doubles bandwidth, L should be at least twice as large as the oversampling factor that is minimally needed to avoid aliasing in the ADC. This compares favorably to other nonlinearities, which all have a considerably larger bandwidth expansion factor. In many instances the ADC input will have less than 100% excess bandwidth. Then $L = 4$ suffices, whence

$$\tilde{Z} = \sum_{n=4mM}^{4(m+1)M-1} (-j)^n x_n^2. \tag{10.31}$$

Hence computation of $\tilde{Z}$ requires only sign inversions and additions, and the fact that the synchronizer needs 4 samples per symbol interval is relevant virtually only to the ADC. Even so, this digital solution is not necessarily simpler than a properly crafted analog solution.

10.5 EARLY-LATE TIMING RECOVERY

These schemes, which originate in [38] and [28], are closely related to the data-aided maximum-likelihood timing-error detector of Section 10.2.2. In that TED, a matched filter is followed by a differentiator, whose output is sampled under the control of the VCO and multiplied with the data signal a_k (see Fig. 10.3). We arrive at the early-late TED upon replacing the differentiation by a time-difference operation (Fig. 10.37).

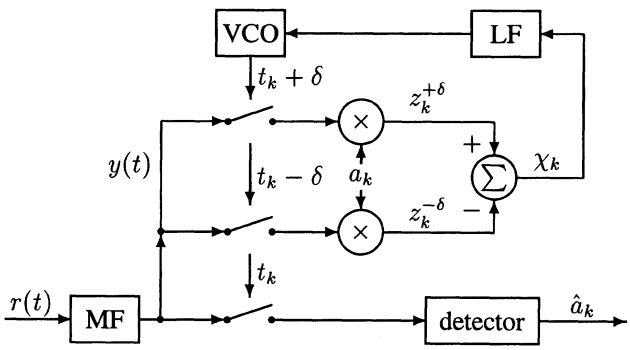


Figure 10.37: Data-aided early-late tracking loop.

The output $y(t)$ of the matched filter (MF) is sampled at instants $t_k \pm \delta$, i.e. slightly before and slightly after the instants t_k that are used for data detection. The difference $\frac{1}{2\delta}[y(t_k+\delta)-y(t_k-\delta)]$ will approximate the derivative of $y(t)$ at the instant t_k provided that the offset $|\delta|$ is small with respect to the symbol interval T. In Fig. 10.37, the scaling

factor $1/(2\delta)$ is incorporated in the loop filter. Furthermore, the samples $y(t_k + \delta)$ and $y(t_k - \delta)$ are both multiplied by a_k before being subtracted. This reversal of order does not affect the TED output χ_k. Performance will approach that of the DA ML TED (see Section 10.2.3) if $|\delta|$ is sufficiently small.

A disadvantage of the scheme of Fig. 10.37 is that it requires two samplers on top of the one that is needed for data detection. To save one sampler, the detector can be connected to one of both samplers in the timing recovery path. For sufficiently small $|\delta|$, the fact that decisions are taken $|\delta/2|$ seconds too early or late will not degrade detector performance seriously.

In [56] a related technique is discussed with only a single sampler that alternates between sampling slightly too early and slightly too late. That technique is of the MMSE type (see Section 10.6) and is applicable to linear and decision feedback equalizers.

Non-data-aided early-late tracking loop For binary data $a_k \in \{-1, 1\}$ it is easy to amend the loop of Fig. 10.37 into a non-data-aided (NDA) version. To this end, we note that the sampled output $y(t_k)$ may be viewed as a data component $a_k q(\Delta T)$ corrupted by noise and by ISI due to past and future symbols a_{k-i} $(i \neq 0)$. Let us assume that ISI is not severe, that noise is small, and that the loop is in the tracking mode of operation such that the actual sampling instants t_k are close to the desired ones. Then the eye behind the MF will be open in the vicinity of the instants t_k, i.e. $y(t_k \pm \delta)$ should have the same sign as a_k. Consequently $z_k^{\pm\delta} \triangleq a_k y(t_k \pm \delta) = |y(t_k \pm \delta)|$. Thus a full-wave rectifier suffices to produce $z_k^{\pm\delta}$; the data a_k itself is not needed. This suggests the NDA tracking loop of Fig. 10.38. This loop is only equivalent to its DA counterpart when ISI and noise are sufficiently small. Just as the squaring loop, it tends to break down when excess bandwidth vanishes.

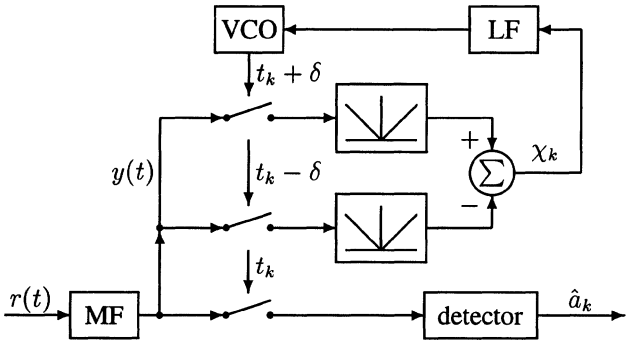


Figure 10.38: Non-data-aided early-late tracking loop.

In essence, the loop of Fig. 10.37 seeks the sampling phase ψ at which the MF output has maximum average amplitude. Virtually the same effect would be reached if the power of the MF output were maximized. This would necessitate squarers instead of

full-wave rectifiers. In fact, any symmetric nonlinearity can be used. Problem 10.27 serves to show for the configuration with squarers that the loop will settle down at the proper sampling phase for data detection. The only condition is that the system response $q(t)$ be symmetric with non-negative Fourier transform. This is ensured by the matched filter, and is also true for most Nyquist-1 functions. Thus the scheme of fig. 10.38 remains applicable when the MF is replaced by a full-response linear equalizer (FRLE), be it that tracking performance will suffer to a certain extent. A similar remark applies to the scheme of Fig. 10.37.

Wave difference method The discrete-time analog of the early-late technique is the *wave difference method* (WDM) [60, 4]. It requires an oversampling factor L that is high enough to avoid aliasing of the nonlinearly-processed signal. Typically $L = 4$ when use is made of squarers (compare section 10.4, page 533). This rather cumbersome factor may be halved with the aid of simple digital interpolation filters [4].

10.6 MMSE AND SAMPLED-DERIVATIVE TIMING RECOVERY

This data-aided technique is closely related to LMS adaptation techniques (see Section 8.6). Just as in LMS adaptation, an error e is formed that represents the difference between the actual detector input $\tilde{d}$ and the desired input d, and control is directed towards minimization of the power of e. MMSE timing recovery can be used with a variety of modulation codes and detection techniques, and performance can be close to optimum. The baseline MMSE scheme uses two samplers that operate on the output signal of a prefilter and the derivative of that signal. Both samplers must be accurately synchronized with respect to each other. To avoid this difficulty, the second sampler may be replaced by a discrete-time filter that operates on the output of the first sampler in order to approximate the output of the second one. This approach is simpler than the baseline scheme, yet performance can be comparable. A further simplification consists of replacing e by the desired input d. This leads to a category of schemes that encompasses the classical sampled-derivative variety. Our treatment extends upon much of the literature on MMSE timing recovery in that it accounts for both oversampling and error whitening (among the exceptions are [35] and [51]). These two ingredients are necessary and sufficient for MMSE timing recovery to achieve ML performance. In Section 10.7 we will see that zero-forcing (ZF) techniques provide an alternative and often cheaper means towards this end. For this reason our treatment of MMSE techniques is relatively concise. The baseline MMSE scheme and its approximation with a single sampler are discussed in Sections 10.6.1-10.6.3. Section 10.6.4 deals with sampled-derivative timing recovery, while Section 10.6.5 provides performance examples for idealized digital recording channels. We refer to [35], [50], [55], [28], [41], [43], [56], [51] and references therein for variations on the theme and for further details.

10.6.1 Development of MMSE timing recovery

In Fig. 10.39, a data sequence a_k of data rate $1/T$ is applied to a channel with impulse response $h(t)$, additive Gaussian noise $n(t)$ and an *a priori* unknown delay ϕ (expressed in symbol intervals T). Prior to detection, the receiver performs filtering and equalization operations that serve to suppress noise and condition ISI. A single continuous-time prefilter with impulse response $f(t)$ accounts for the compound effect of these operations. The prefilter output $y(t)$ is invariably bandlimited and can be sampled without loss of information at some integer oversampling factor L relative to the symbol rate $1/T$. Sampling occurs at instants $t_n = (\frac{n}{L} + \psi)T$ where ψ is a sampling phase normalized in units T. (The subscripts n and k are used for sequences of sampling rate L/T and $1/T$, respectively.) If excess bandwidth is negligible then baud-rate sampling ($L = 1$) will do.

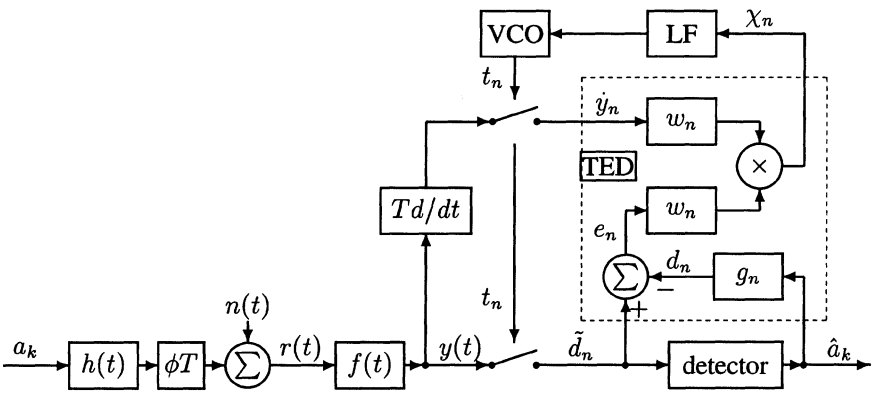


Figure 10.39: Data transmission system with MMSE timing-recovery scheme.

The sampled sequence $\tilde{d}_n$ of sampling rate L/T serves as a basis for timing recovery and symbol detection. The detector is meant for some prescribed target response g_n, i.e. it operates under the assumption that the sampled response of the system up until the detector is equal to g_n. Unless stated otherwise we shall assume that the channel is equalized perfectly, i.e. that the sampled system response $q_n^\Delta \triangleq (h * f)((\frac{n}{L} + \Delta)T)$ is equal to g_n when sampling occurs at the correct phase ($\Delta = 0$). The timing-recovery scheme is meant to align ψ with ϕ and in doing so will cause q_n^Δ to coincide with g_n. If sampling-phase errors and noise were absent, therefore, the detector would have input $d_n \triangleq (a * g)_n$. Following adaptive filtering conventions we call d_n the *desired input* of the detector. The error $e_n \triangleq \tilde{d}_n - d_n$ consists purely of filtered noise $v_n \triangleq (n * f)(t_n)$ if and only if equalization is perfect and sampling occurs at the correct phase. In order to synthesize local versions of d_n and e_n, the data sequence a_k must be known to the receiver. This knowledge may be derived in a data-aided (DA) fashion from a known preamble that precedes the actual data, or in a decision-directed (DD) fashion as in Fig. 10.39 once decisions have become sufficiently reliable. In the sequel we do

not make any distinction between DA and DD, i.e. we are not concerned with the effect of decision errors.

It can be observed that e_n is obtained from the prefilter output $y(t)$ via two reversible operations (alias-free sampling, subtraction of a known data sequence). Any information that is present in $y(t)$ must, therefore, still be present in e_n. This applies in particular to information with respect to ϕ. It follows that e_n is a sufficient statistic for maximum-likelihood (ML) timing recovery. The noise component v_n of e_n is Gaussian but is, in general, not white. To arrive at white noise we must apply a noise-whitening filter whose impulse response is denoted w_n. The whitening operation is reversible and the filtered error $(e * w)_n$ is, therefore, still a sufficient statistic for ML timing recovery. By a similar argumentation as in Section 10.2 (see also Appendix 3C) one readily establishes that ML timing recovery boils down to minimization of the power of the whitened error $(e * w)_n$.

Classical MMSE schemes act to minimize the power of e_n and use an oversampling factor $L = 1$ (see, for example, [41, 43]). ML performance will accrue if and only if excess bandwidth is negligible and the sampled noise component v_n is white. The MMSE approach is readily extended to minimization of the power of $(e * w)_n$ for an arbitrary oversampling factor L. To this end we observe that

$$\frac{\partial (e * w)_n^2}{\partial \psi} = 2(e * w)_n \frac{\partial (\tilde{d} * w)_n}{\partial \psi} = 2(e * w)_n (\dot{y} * w)_n \tag{10.32}$$

where $\dot{y}_n \triangleq Ty'(t_n)$. This expression suggests that we detect timing errors by multiplying $(e*w)_n$ with a noisy reference sequence $(\dot{y}*w)_n$ that is derived from $y(t)$. The loop acts to force the average value of the TED output $\chi_n \triangleq (e * w)_n (\dot{y} * w)_n$ towards zero and in doing so will approximately minimize the power of the whitened error $(e * w)_n$ (Fig. 10.39).

The target response g_n is typically a simple and causal function of short duration. For this reason computation of d_n and e_n is simple. The error sequence e_n may, in fact, already be available within the detector, or within the adaptation subsystem if one is present. Detection delays and implementation latencies may make it impossible to produce e_k without delay. Any delay of e_n must be matched in $(\dot{y} * w)_n$. The net effect will be a loop delay. Loop properties will not degrade as long as loop delay is kept well below the inverse of the loop bandwidth (see Chapter 11). Ideally the oversampling factor L is high enough to avoid aliasing, and the error is whitened by means of an error-whitening filter. To restrict complexity, one commonly operates the scheme at the baud rate ($L = 1$) and omits the error filter. Both simplifications result in a certain performance degradation.

Example 10.10 Assume that $L = 1$ and that noise is small. If the output of the prefilter in Fig. 10.39 is a rectangular NRZ signal (Fig. 10.40.a) then there is much timing information. This information is, however, fully concentrated in the transitions between data levels, and none of it is present near the optimum sampling instants (which are indicated by open circles). For all sampling-phase errors $|\Delta| < 0.5$ the sampled error e_k and the sampled derivative $\dot{y}_k$ consist only of noise and hence convey no timing information whatsoever. As a result, the scheme of Fig. 10.39 cannot work.

As excess bandwidth decreases, the signal gets 'rounded' (Fig. 10.40.b) and timing information gets more evenly spread across sampling phases. Thus, somewhat paradoxically, performance of the scheme of Fig. 10.39 is bound to improve even though less timing information is present. This is in marked contrast to deductive schemes, like those of Sections 10.3 and 10.4, whose performance invariably deteriorates as excess bandwidth decreases.

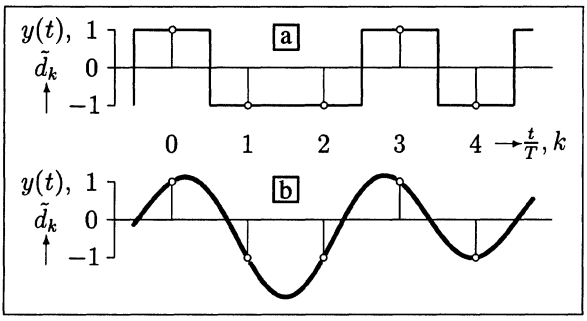


Figure 10.40: Binary data signal carried by rectangular NRZ pulses (a) and Nyquist-1 pulses with negligible excess bandwidth (b).

These observations carry over to other baud-rate schemes, including those of Section 10.7. In cases of practical interest there is often little if any excess bandwidth, especially under worst-case conditions. In that case it will suffice to operate the scheme at the baud rate ($L = 1$). For wideband systems it is advantageous to resort to oversampling. This need not increase complexity significantly.

Example 10.11 Consider a full-response system with raised-cosine roll-off characteristics. Since excess bandwidth can be at most 100%, an oversampling factor of 2 suffices to avoid any information loss. For a roll-off factor $\beta = 1$, the eye pattern is virtually horizontal at the decision instants, and most timing information is concentrated near the zero-crossings that occur halfway between the decision instants (see Fig. 10.12). It suffices, therefore, to compute χ_n for odd n. To this end we note that $\chi_{2k-1} = e_{2k-1}\dot{y}_{2k-1} = (y_{2k-1} - d_{2k-1})\dot{y}_{2k-1} = (y_{2k-1} - a_k g_{-1} - a_{k-1} g_1)\dot{y}_{2k-1}$. Upon realizing that $g_{-1} = g_1 = 0.5$ we arrive at the topology of Fig. 10.41. A related scheme is described in [51].

The lower sampler in Fig. 10.41 alternates between the upper and lower positions on successive sampling instants t_n; the upper sampler only closes halfway between the decision instants. Just as in baud-rate MMSE schemes, TED outputs χ and the preceding sequences $\dot{y}$, e, $\tilde{d}$ and d are produced only once per symbol interval.

10.6.2 Properties of MMSE timing recovery

Timing function This function is analyzed in Appendix 10D.1. It is similar in nature to the one for ZF timing recovery (see Example 10.13), and the effect of decision errors is also similar. In the absence of misequalization (i.e. when $q_n = g_n$), the function crosses zero at a sampling-phase error $\Delta = 0$ and the timing-recovery loop will settle

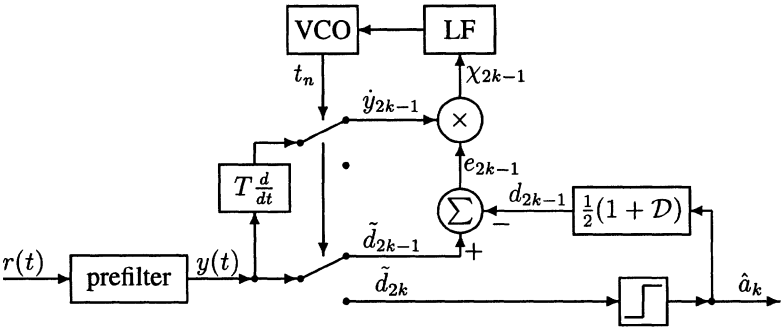


Figure 10.41: Oversampled MMSE timing recovery scheme for full-response system with roll-off factor $\beta = 1$.

down at the desired average phase $\psi = \phi$. In the presence of misequalization, the loop continues to minimize the MSE and this may lead to a steady-state sampling phase $\psi \neq \phi$.

Pattern-dependent jitter Provided that equalization is perfect ($q_n = g_n$), the error e_n in Fig. 10.39 consists purely of noise when sampling occurs at the correct phase ($\psi = \phi$). Then pattern-dependent jitter is negligible in the tracking mode of operation. Misequalization causes residual ISI which, in turn, induces pattern-dependent jitter. The amount of jitter depends on the degree and nature of misequalization. Residual amplitude distortion is generally less harmful than residual phase distortion.

Noise-induced jitter In the tracking mode of operation, sampling-phase errors are generally small, and equalization will normally be sufficiently accurate for e_k to consist mainly of noise. Jitter is then primarily induced by noise, and we may rate the scheme in terms of the *efficiency* γ that was defined in Section 9.3.3. Appendix 10D.1 analyzes this efficiency for the important practical case $L = 1$, and also optimizes it with respect to w_k. Not surprisingly, it is found that w_k should ideally be chosen in order for the filtered error $(e * w)_k$ to be white. This remains true when the sampling operation introduces aliasing, but ML performance is then beyond reach. At high SNRs, the optimum efficiency approaches a limit $\hat{\gamma}_{\text{BR}}$ that applies to any baud-rate scheme and is attainable for zero-forcing schemes (see Section 10.7). This limit is given by

$$\hat{\gamma}_{\text{BR}} \triangleq \frac{1}{\text{SNR}} \int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega}) \frac{|P(e^{j2\pi\Omega})|^2}{\mathcal{V}(e^{j2\pi\Omega})} d\Omega \tag{10.33}$$

where $\mathcal{A}(e^{j2\pi\Omega})$ and $\mathcal{V}(e^{j2\pi\Omega})$ denote the power spectral density of data a_k and sampled noise v_k, respectively, and where $P(e^{j2\pi\Omega})$ is the discrete-time Fourier transform of the sampled derivative $p_k \triangleq Tq'(kT)$ of the system response $q(t)$.

Error whitening will always yield optimum efficiency but may not always be needed. In particular, for sinusoidal data a_k it is demonstrated in Problem 10.30 that

the efficiency limit of (10.33) can be approached at high SNRs irrespective of w_k. The simplest possibility is to omit both filters with impulse response w_k. This is done in most MMSE schemes that are described in the literature.

Exercise 10.7 Show that $\hat{\gamma}_{\mathrm{BR}}$ coincides with the ML efficiency γ_{ML} of (10.15) when excess bandwidth is absent.

10.6.3 Approximate MMSE timing recovery

In the MMSE scheme of Fig. 10.39, timing-error information is produced by correlating the filtered error $(e * w)_n$ with a noisy reference sequence $(\dot{y} * w)_n$ that is derived from $y(t)$. This approach has various disadvantages:

1. Formation of $\dot{y}_n$ requires a second sampler that operates in exact synchronism with the one in the detection path. Even small delay mismatches can be detrimental.

2. Formation of $(e * w)_n(\dot{y} * w)_n$ is comparatively complicated when carried out digitally. In particular, digital multiplications may not be avoidable.

3. During periods in which no timing information is available (i.e. whenever the data sequence a_k exhibits long strings of identical symbols), the stochastic gradient of (10.32) will not be zero but will be fully induced by noise. As a result, the sampling phase may drift away randomly.

In Section 10.7 we will see that zero-forcing timing recovery takes aim at all three disadvantages. We now discuss an approximate MMSE scheme that avoids the first one. The trick is to approximate the differentiation in Fig. 10.39 by a discrete-time filtering operation on $\tilde{d}_n$. The resulting topology has only one sampler and is depicted for $L = 1$ in Fig. 10.42. For simplicity the filters with impulse response w_k have been omitted.

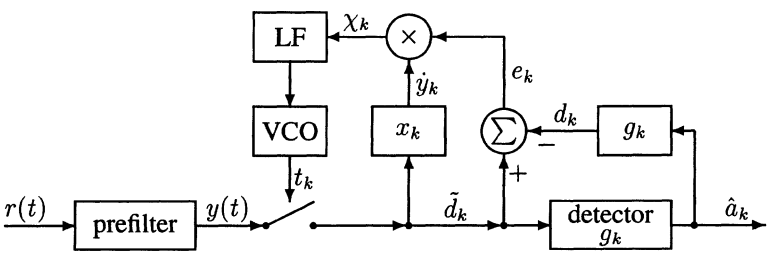


Figure 10.42: Data receiver with simplified MMSE timing-recovery loop.

In the absence of excess bandwidth, $\dot{y}_k$ can be computed from y_k to any desired degree of accuracy. This necessitates a filter whose impulse response x_k ideally satisfies

$$x_k \triangleq \begin{cases} 0, & k = 0, \\ \frac{(-1)^k}{k}, & k \neq 0. \end{cases} \tag{10.34}$$

For a derivation of this response we refer to Problem 10.38. In practice a finite-length approximation must be used for reasons of complexity. Even simple approximations may yield good results when excess bandwidth is not too large. Of particular interest is the two-term approximation $x_{-1} = 1$, $x_1 = -1$ and $x_k = 0$ otherwise [50, 2]. A delay of one symbol interval is needed to arrive at a causal implementation. The error sequence e_k must be delayed accordingly. The net effect will be a loop delay. We arrive at the configuration of Fig. 10.43. A related oversampled scheme is described in [51].

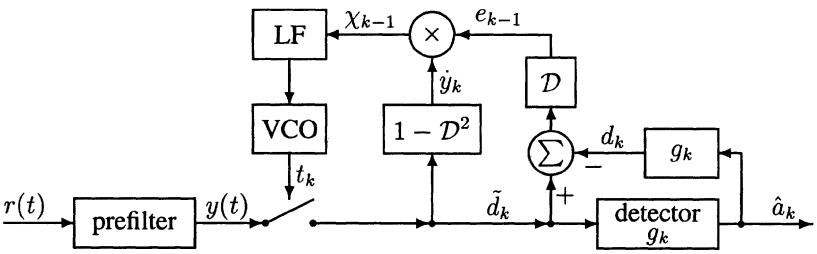


Figure 10.43: Variant of Fig. 10.42 due to Qureshi [50]. Here the filter with impulse response x_k performs a $1 - \mathcal{D}^2$ operation, where $\mathcal{D}$ represents a delay of one symbol interval.

Efficiency of the scheme of Fig. 10.42 is analyzed and optimized as a function of x_k in Appendix 10D.2. In the analysis it is assumed that v_k is negligible compared to the data component of $\tilde{d}_k$. Subject to this assumption it is found that x_k can always be chosen for efficiency to approach $\hat{\gamma}_{\mathrm{BR}}$, irrespective of the data and noise spectra. This means that tracking performance need not lag behind that of MMSE timing recovery even when noise is colored. For sinusoidal data a_k, many different responses x_k will do (see Problem 10.31).

10.6.4 Sampled-derivative timing recovery

The MMSE schemes of Figs. 10.39-10.43 all derive timing information via the error e or a filtered version of it. We will now see that it is often permissible to replace e_k by the desired detector input d_k. A disadvantage of this simplification is that pattern-dependent jitter arises, but in many instances this jitter tends to be small. By way of illustration we consider the loop of Fig. 10.39. Here $\chi_n = (\dot{y} * w)_n (e * w)_n = \xi_n - (\dot{y} * w)_n (d * w)_n$ where $\xi_n \triangleq (\dot{y} * w)_n (\tilde{d} * w)_n$. The loop forces an average of χ_n towards zero. The time constant of averaging depends on the loop filter but invariably amounts to many symbol intervals. Tracking properties are, therefore, mainly determined by the DC-component of χ_n. In Problem 10.33 it is demonstrated that ξ_n is DC-free irrespective of $y(t)$ and w_n provided that the oversampling factor L is large enough to avoid aliasing. Omission of ξ_n from χ_n will, then, affect neither the timing function $\rho(\Delta)$ nor the power spectral density of χ_n at DC. Hence, provided that loop bandwidth is sufficiently small, pattern-dependent jitter will be negligible and tracking properties will remain essentially unaffected. A baud-rate version of the implied

timing-recovery scheme is depicted in Fig. 10.44. For simplicity, the filters with impulse response w have again been omitted.

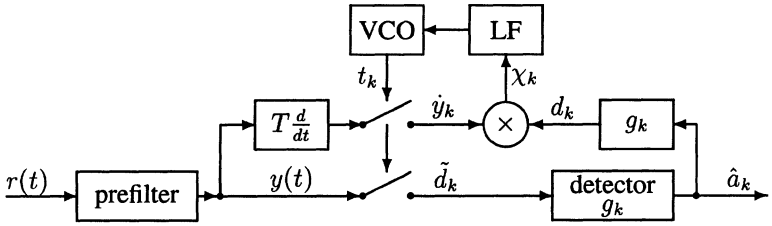


Figure 10.44: Data receiver with sampled-derivative timing-recovery scheme.

The topology of Fig. 10.44 becomes particularly simple for a full-response system. Here the detector takes the form of a slicer with zero threshold and the filter with impulse response g_k degenerates. It is in this form that sampled-derivative timing recovery originated [55] (see also [28]).

Replacement of e_k by d_k in Figs. 10.42 and 10.43 leads to a topology that we will reencounter for zero-forcing timing recovery (Fig. 10.62), and we refer ahead to Section 10.7.10 for a discussion.

10.6.5 Performance for idealized recording channels

Lorentzian channel equalized to Class IV We first consider the Lorentzian channel of Example 10.4 when equalized in a zero-forcing manner to Class IV partial response. Two equalization criteria are considered:

Minimum noise enhancement (MNE): Here the signal-to-noise ratio before detection is optimized. As a result, excess bandwidth of the equalized system is, coarsely speaking, proportional to that of the channel (see Chapter 6).

Zero excess bandwidth (ZEBW): Here all signal components beyond the Nyquist frequency are suppressed. This leads to the system response of Example 10.13 (Fig. 10.52).

Fig. 10.45 compares the baud-rate efficiency limit $\hat{\gamma}_{\mathrm{BR}}$ for these two responses. Also included is the the maximum-likelihood limit γ_{ML}. Both uncorrelated data and the binary 4T-pattern $\ldots++--++--\ldots$ are considered. The $2T$ data pattern $\ldots+-+-\ldots$ is impractical because it is completely rejected for both responses.

The Lorentzian channel has considerable excess bandwidth at low densities, and so does the MNE system response. We have argued in Example 10.10 that this tends to degrade the efficiency of baud-rate schemes. This is indeed visible in curves c. By comparison, the ZEBW response permits efficiencies within 1-2 dB from ML for densities of practical interest (say $D = 1 - 3$). Excess bandwidth of the channel vanishes at very high densities. Here MNE and ZEBW system responses coincide, and baud-rate schemes can achieve ML efficiency. Efficiencies for the $4T$ pattern are, on the whole, around 5 dB better than for uncorrelated data. This helps to explain the popularity of the

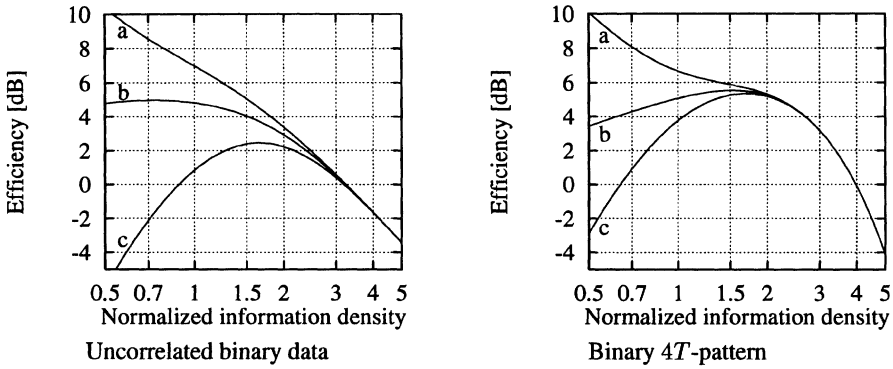


Figure 10.45: Efficiencies for a Lorentzian channel as a function of the normalized information density D. a. Maximum-likelihood (γ_{ML}); b, c: optimum baud-rate ($\hat{\gamma}_{\mathrm{BR}}$) when the channel is equalized to a Class IV system response with zero excess bandwidth (b.) or minimum noise enhancement (c.).

$4T$ training pattern [14]. For this pattern, the baud-rate version of the MMSE scheme of Fig. 10.39 approaches $\hat{\gamma}_{\mathrm{BR}}$ at high SNRs (see Problem 10.30). A similar statement applies to the approximate MMSE scheme of Fig. 10.42 (see Problem 10.31). For uncorrelated data, by comparison, some detraction from $\hat{\gamma}_{\mathrm{BR}}$ is generally inevitable. This is illustrated in Fig. 10.46 for the ZEBW response. Both considered schemes approach $\hat{\gamma}_{\mathrm{BR}}$ within a few dB in the important density range $D \in [1, 3]$.

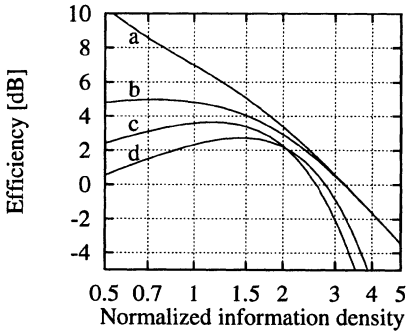


Figure 10.46: Efficiencies for a Lorentzian channel equalized to Class IV ($1 - \mathcal{D}^2$). Data is uncorrelated, excess bandwidth is absent. a. Maximum-likelihood (γ_{ML}); b. optimum baud-rate ($\hat{\gamma}_{\mathrm{BR}}$); c. MMSE scheme of Fig. 10.39; d. approximate MMSE scheme of Fig. 10.43.

Idealized optical recording channel equalized to Duobinary The duobinary ($1+\mathcal{D}$) target response is a natural one for the channel of Example 10.4. We consider only the variant without excess bandwidth and assume noise $n(t)$ to be white. Fig. 10.47 depicts

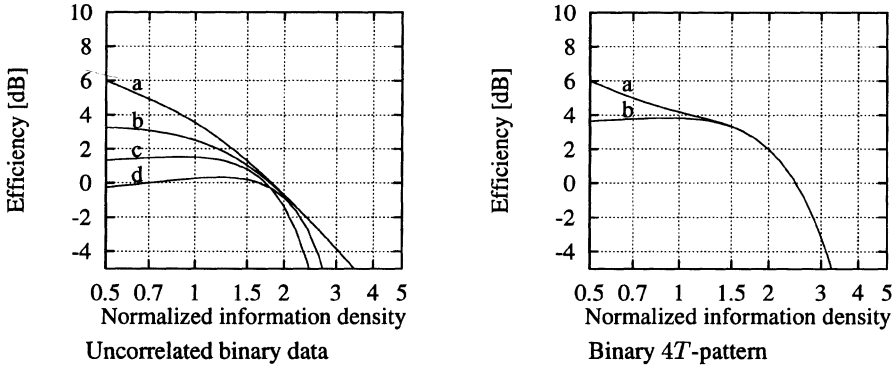


Figure 10.47: Efficiencies for an idealized optical recording channel equalized to a Duobinary $(1+\mathcal{D})$ target response without excess bandwidth. a. Maximum-likelihood (γ_{ML}); b. optimum baud-rate ($\hat{\gamma}_{\mathrm{BR}}$); c. MMSE scheme of Fig. 10.39 with baud-rate sampling ($L = 1$); d. approximate MMSE scheme of Fig. 10.43.

corresponding efficiencies. For uncorrelated data, both MMSE-type schemes approach $\hat{\gamma}_{\mathrm{BR}}$ within a few dB for densities of practical interest. Since duobinary has a zero at the Nyquist frequency, the $2T$ pattern is impractical. For the $4T$ pattern, $\hat{\gamma}_{\mathrm{BR}}$ is essentially within reach for both schemes, and is close to γ_{ML} except at the lowest densities.

10.7 ZERO-FORCING TIMING RECOVERY

10.7.1 Introduction

This class of data-aided techniques is closely related to the continuous-time maximum-likelihood technique of Section 10.2.1. It bears a strong resemblance to zero-forcing (ZF) adaptation techniques (see Section 8.5). Just as in ZF adaptation, an error e is formed that represents the difference between the actual detector input and the desired one, and control information is generated by cross-correlating e with a filtered data sequence. The control loop acts to eliminate this cross-correlation, and in doing so forces a combination of intersymbol interference components toward zero. This explains the adjective 'ZF'. Replacement of e by the detector input preserves the ZF nature of the scheme, but may induce a certain amount of pattern-dependent jitter. ZF timing recovery can be used with a variety of modulation codes and reception techniques, and is normally simple. Even so, tracking performance can be close-to-optimum. These advantages, along with the relative ease of analysis and the synergy with ZF adaptation, justify an extensive treatment of the approach.

The origins of ZF timing recovery can be traced back to the classical paper of Mueller and Müller [44]. The underlying principles have since been applied to a variety of codes, channels and detection techniques (see, for example, [18], [33], [26], [65], [7],

[14], [29], [11] and [12]). This section provides a unified treatment of these approaches. Section 10.7.2 derives the basic structure of the schemes. Subsequent sections provide a performance analysis and discuss various examples and extensions.

10.7.2 Structure of ZF schemes

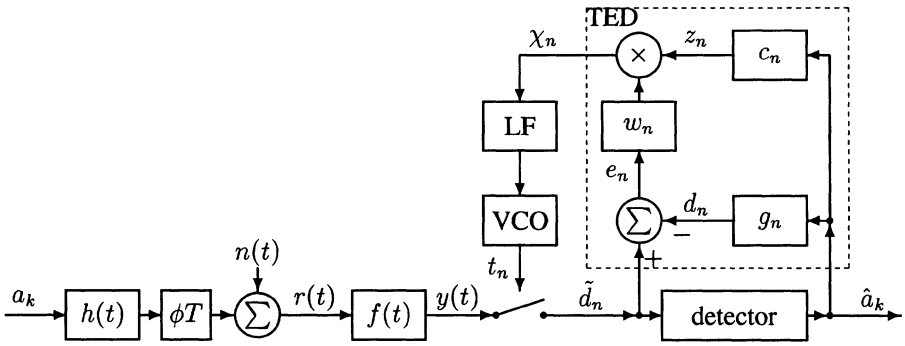


Figure 10.48: Data transmission system with zero-forcing timing-recovery scheme.

Zero-forcing (ZF) timing recovery may be developed along the same lines as MMSE timing recovery. Except for the timing-error detector, the system model of Fig. 10.48 is identical to the one of Fig. 10.39, and we refer back to Section 10.6.1 for a general explanation. Just as in Section 10.6.1, sampling occurs at instants $t_n = (\frac{n}{L} + \psi)T$ where L is an oversampling factor that is ideally large enough to avoid aliasing. The detector is meant for a target response g_n, i.e. it has a desired input $d_n = (a * g)_n$. The actual input $\tilde{d}_n$ may deviate from d_n as a result of misequalization, sampling-phase errors and noise. Unless stated otherwise we assume that equalization is perfect, i.e. that the sampled system response $q_n^\Delta \triangleq q((\frac{n}{L} + \psi)T)$ coincides with g_n when sampling occurs at the correct phase ($\Delta = \psi - \phi = 0$). At this phase the error $e_n = \tilde{d}_n - d_n$ will, then, consist purely of sampled noise $v_n = (n * f)(t_n)$. This noise is, in general, not white. A filter with impulse response w_n may be used for whitening purposes and we have argued in Section 10.6.1 that maximum-likelihood timing recovery boils down to minimization of the power of the whitened error $(e * w)_n$. In the MMSE scheme of Fig. 10.39, this minimization is accomplished by correlating $(e * w)_n$ with a noisy reference sequence $(\dot{y} * w)_n$ that is derived from $y(t)$ via a differentiator and sampler. This approach has certain disadvantages as listed in Section 10.6.3. ZF timing recovery mitigates or overcomes these disadvantages. The key idea is to replace the noisy reference sequence $(\dot{y} * w)_n$ by a nominal reference sequence z_n that would emerge in the absence of noise, misequalization and sampling-phase errors. This sequence may be generated locally by applying decisions $\hat{a}_k$ to an interpolating filter with

impulse response $c_n = (p * w)_n$ where

$$p_n \triangleq Tq'(\frac{nT}{L}) \tag{10.35}$$

is the sampled derivative of the system response $q(t) = (h * f)(t)$.

Exercise 10.8 Verify that z_n may indeed be generated in this manner.

The timing-error detector (TED) that springs forth from these considerations is depicted in Fig. 10.48. It has output $\chi_n \triangleq (e * w)_n z_n$ and forms part of the timing-recovery loop that is closed via a loop filter (LF) and VCO. The loop acts to eliminate all cross-correlation between $(e * w)_n$ and z_n and thereby forces a linear combination of ISI components towards zero (see Section 10.7.4). Thus it will not necessarily minimize the power of the error e_n, especially in the face of misequalization. If w_n and c_n are chosen as described above then ML performance will accrue. We will see later that other choices can also be attractive. In particular, w_n and c_n can be truncated to only a few nonzero components so as to simplify implementation at a modest expense to performance.

The scheme of Fig. 10.48 is highly similar to the continuous-time timing-recovery scheme of Fig. 10.2 as well as to ZF equalizer adaptation schemes. The latter schemes also correlate the error e with a filtered data sequence and force a linear combination of ISI components towards zero (see Chapter 8).

10.7.3 Equivalence classes

The timing-recovery loop of Fig. 10.48 acts to force an average of the TED output towards zero. The time constant of averaging depends on the LF but invariably amounts to many symbol intervals. TED properties are, therefore, largely determined by the DC-component of χ_n. This component amounts to

$$\sum_{n=-\infty}^{\infty} \chi_n = \sum_{n=-\infty}^{\infty} (e * w)_n (a * c)_n = \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} e_i a_j (w_- * c)_{i-jL} \tag{10.36}$$

where the subscript '$_-$' denotes time-reversal. We may apparently regard two TEDs with impulse responses (w_n^1, c_n^1) and (w_n^2, c_n^2) as equivalent whenever $(w_-^1 * c^1)_n = (w_-^2 * c^2)_n$ for all n. In $\mathcal{D}$-transform notation this equivalence condition reads

$$W_1(\mathcal{D}^{-1})C_1(\mathcal{D}) = W_2(\mathcal{D}^{-1})C_2(\mathcal{D}) \tag{10.37}$$

where $\mathcal{D}$ denotes a delay of one sampling interval (T/L seconds).

Example 10.12 Assume that w_n and c_n are chosen as in Section 10.7.2. Then

$$W(\mathcal{D}^{-1})C(\mathcal{D}) = \frac{P(\mathcal{D})}{\mathcal{V}(\mathcal{D})} \tag{10.38}$$

where $P(\mathcal{D})$ is the $\mathcal{D}$-transform of p_n and $\mathcal{V}(\mathcal{D})$ denotes the power spectral density of v_n (i.e. the $\mathcal{D}$-transform of the autocorrelation function of v_n). Any TED that satisfies (10.38) will yield ML performance. Two limiting options are attractive from an implementation standpoint:

1. The error filter is omitted $[W(\mathcal{D}) = 1, C(\mathcal{D}) = P(\mathcal{D})/\mathcal{V}(\mathcal{D})]$. The remaining filter operates on a data sequence a_k and can be implemented without any digital multiplications.
2. The data filter is omitted $[W(\mathcal{D}) = P(\mathcal{D}^{-1})/\mathcal{V}(\mathcal{D}), C(\mathcal{D}) = 1]$. Then z_n and χ_n will differ from zero for $n = kL$ only. As a result, the output of the error filter needs only be computed at the baud rate. A TED of this type was first proposed by Gardner [27].

It is worth observing that the optimum filters according to (10.38) do not depend on the modulation code. Performance, of course, does depend on the code (see Appendix 10E).

Latency and causality Detection delays and implementation latencies may make it impossible to produce the error e_n without delay. Any delay of e_n must be matched in z_n. The reason is that

$$[W(\mathcal{D}^{-1})D^{-N}][C(\mathcal{D})D^N] = W(\mathcal{D}^{-1})C(\mathcal{D}) \tag{10.39}$$

irrespective of the delay N. Insertion of identical delays in the error and data paths, therefore, leaves the DC-component of the TED output unaffected. The net effect of delaying e_n and z_n equally is a loop delay. Tracking properties will not degrade as long as loop delay is kept well below the inverse loop bandwidth. In cases where w_n and/or c_n differ from zero for $n < 0$, we can similarly use matched delays to obtain an equivalent yet causal TED.

10.7.4 Properties of ZF schemes

Timing function The average value of the TED output χ_n constitutes the timing function $\rho(\Delta)$. Appendix 10E shows that

$$\rho(\Delta) = \frac{1}{L}\sum_{k=-\infty}^{\infty} R_k^a((q^\Delta - g) * w * c_-)_{kL} \tag{10.40}$$

where $R_k^a \triangleq E[a_l a_{l+k}]$ is the data autocorrelation function. Nominally $q_n^0 = g_n$. Irrespective of R_k^a, c_n, w_n and g_n, $\rho(\Delta)$ will then cross zero at $\Delta = 0$ and the loop will settle down at the desired phase $\phi = \psi$. Misequalization causes q_n^0 to deviate from g_n and may cause the loop to settle down at a phase $\psi \neq \phi$. The resulting MSE can be poorer than that of an MMSE scheme. This is a potential disadvantage of ZF timing recovery. For uncorrelated data of unit power and baud-rate sampling ($L = 1$), (10.40) becomes

$$\rho(\Delta) = ((q^\Delta - g) * w * c_-)_0 = \sum_{k=-\infty}^{\infty} (w_- * c)_k (q_k^\Delta - g_k). \tag{10.41}$$

This is a linear combination of residual ISI components $(q_k^\Delta - g_k)$ with weights $(w_- * c)_k$. A timing function of this form was first proposed by Mueller and Müller [44]. In [44]

the coefficients $(w_- * c)_k$ are prescribed, and a somewhat involved analysis produces TEDs as in Section 10.7.11 that are efficient estimators of the timing function. The scheme of Fig. 10.48, by comparison, depends trivially on w_k and c_k, and we will see shortly that a choice for w_k and c_k involves a simple tradeoff of performance against complexity.

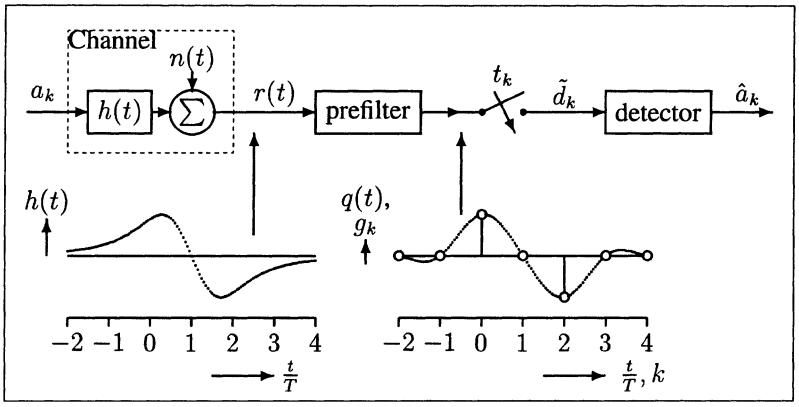


Figure 10.49: Class IV partial-response system.

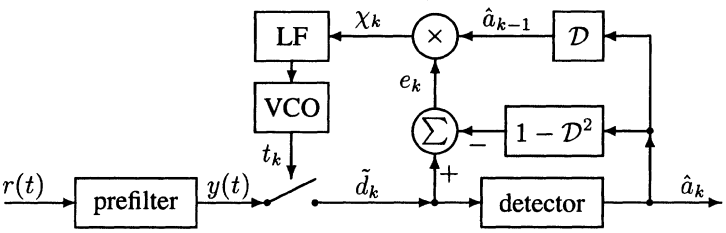


Figure 10.50: Class IV receiver with "poor-man's" ZF timing-recovery scheme.

Example 10.13 In Fig. 10.49, a received signal $r(t)$ is equalized to Class IV partial response, sampled at the baud rate ($L = 1$) and detected. In the vicinity of the optimum sampling phase ($|\Delta| \ll 1$), the sampled system response q_k^Δ will approach the Class IV response $G(\mathcal{D}) = 1 - \mathcal{D}^2$. The biggest deviation occurs for $k = 1$ as a result of the large slope of $q(t)$ in the vicinity of $t = T$. Also, the sign of q_1^Δ provides an indication of that of Δ. These considerations suggest that we take q_1^Δ as our timing function. According to (10.41), this function emerges when $W(\mathcal{D}^{-1})C(\mathcal{D}) = \mathcal{D}$. One option is to omit the error filer ($W(\mathcal{D}) = 1$). The data filter then degenerates into a delay of one sampling interval ($C(\mathcal{D}) = \mathcal{D}$). The implied TED is depicted in Fig. 10.50. For binary data a_k, computation of χ_k involves only additions, subtractions and bit delays. In spite of its simplicity, the scheme of Fig. 10.50 can be close to optimum (see Example 10.14 and Fig. 10.67 in Section 10.7.12). According to (10.41), the transfer characteristic has the shape of a Class IV pulse that is centered about $\Delta = 0$ (Fig. 10.51.a). For

random data a_k, data-aided schemes are often used in a decision-directed mode. Decision errors will abound once sampling errors become too large (say when $|\Delta| > 0.2$ to 0.3), and the TED ceases to deliver a meaningful output. In effect, the TED transfer function folds back toward zero; at the same time the disturbance component of the TED output increases rapidly. By way of illustration Fig. 10.51.b depicts a transfer function that was obtained by simulation for symbol-by-symbol detector. The transfer characteristic coincides with the one of Fig. 10.51.a for sampling-phase errors $|\Delta| < 0.2$. Beyond that point it levels off and becomes periodic with a period of one symbol interval. Decision-directed acquisition characteristics of the scheme of Fig. 10.50 are presented in Section 11.7.

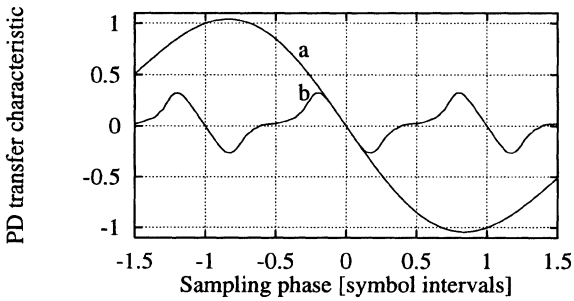


Figure 10.51: TED transfer characteristic $\rho(\Delta)$ for Class IV timing-recovery scheme of Fig. 10.50 with uncorrelated data a_k. a. Data-aided (DA) (no decision errors); b. decision-directed (DD).

Pattern-dependent jitter Provided that equalization is perfect ($q_n = g_n$), the error e_n in Fig. 10.48 consists purely of noise when sampling occurs at the correct phase ($\phi = \psi$). Then pattern-dependent jitter is negligible in the tracking mode of operation. Misequalization causes residual ISI which, in turn, induces pattern-dependent jitter. The amount of jitter depends on the degree and nature of misequalization. Just as for MMSE timing recovery, residual amplitude distortion is generally less harmful than residual phase distortion.

Noise-induced jitter Noise v_n induces a disturbance component $u_n \triangleq z_n(v * w)_n$ in the TED output of Fig. 10.48. This component, in turn, induces jitter. The loop averages u_n across many symbol intervals and noise-induced jitter is, therefore, mainly determined by the power spectral density $\mathcal{U}(e^{j2\pi\Omega/L})$ of u_n at DC ($e^{j2\pi\Omega/L} = 1$).[1] Appendix 10E analyzes $\mathcal{U}(1)$. The appendix also determines the efficiency γ_{ZF} of the TED, and optimizes γ_{ZF} with respect to $W(e^{j2\pi\Omega/L})$ and $C(e^{j2\pi\Omega/L})$. Not surprisingly, the optimum filters are found to obey (10.38). This is true irrespective of L. ML performance, of course, will only accrue if noise is Gaussian and L is large enough to avoid

[1] We recall that u_n is sampled at a rate L/T. Thus $\mathcal{U}$ is periodic in Ω with a period L. The argument $e^{j2\pi\Omega/L}$ reflects this periodicity.

aliasing. Baud-rate operation ($L = 1$) will do if excess bandwidth is absent. For $L = 1$ the optimum efficiency $\hat{\gamma}_{\mathrm{ZF}}$ coincides with the baud-rate limit $\hat{\gamma}_{\mathrm{BR}}$ of (10.33).

In practice the error filter is usually omitted ($W(e^{j2\pi\Omega/L}) = 1$) and data is often approximately uncorrelated. If noise is uncorrelated as well, then according to (10.38) the optimum function c_n is just a scaled version of the sampled derivative p_n. Denote the total energy of p_n by E_p and that of c_n by E_c (i.e. $E_p \triangleq \sum_{n=-\infty}^{\infty} p_n^2$ and $E_c \triangleq \sum_{n=-\infty}^{\infty} c_n^2$). Noise causes Δ to fluctuate about $\Delta = 0$. Denote by σ_Δ^2 the jitter variance for a given loop bandwidth and a given set of coefficients c_n, and by $\hat{\sigma}_\Delta^2$ the variance for the same bandwidth and the optimum coefficients $\hat{c}_n$. With the aid of (10E.2) and (10E.4) one establishes that a suboptimum choice for c_n causes a jitter increment

$$\mathcal{L} \triangleq \frac{\sigma_\Delta^2}{\hat{\sigma}_\Delta^2} = \frac{\hat{\gamma}}{\gamma} = \frac{E_c E_p}{(\sum_{n=-\infty}^{\infty} p_n c_n)^2}. \tag{10.42}$$

This expression remains valid in the presence of aliasing, but even the optimum coefficients $\hat{c}_n$ will then no longer permit ML performance. Clearly $\mathcal{L} \geq 1$, with equality holding if and only if $c_n = \lambda p_n$ for all n, where λ is an arbitrary constant. In the particular event that c_n is a truncated version of p_n, (10.42) simplifies to $\mathcal{L} = E_p/E_c$, i.e. $\mathcal{L}^{-1}$ is just the fraction of all energy of p_n that is included in c_n.

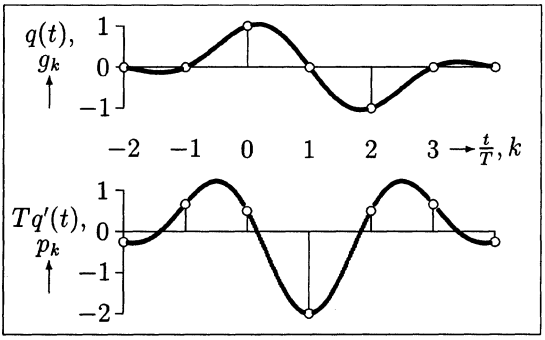


Figure 10.52: Class IV ($1 - \mathcal{D}^2$) response $q(t)$ without excess bandwidth and its derivative $Tq'(t)$. The corresponding sampled responses g_k and p_k are indicated with open circles.

Example 10.14 Let us reconsider the Class IV system of Example 10.13 for the situation that noise v_k is white and that $q(t)$ has no excess bandwidth. The sampled derivative p_k then amounts to $x_k - x_{k-2}$ where x_k is given by (10.34). The coefficient $p_1 = -2$ has largest magnitude by far (Fig. 10.52). Truncation of c_k to only this coefficient yields the scheme of Fig. 10.50 if we accommodate the constant gain factor $c_1 = -2$ in the loop filter. Clearly $\mathcal{L} = E_p/E_c = [(2\pi^2)^2/3 - 1]/4 \simeq 1.395$. This corresponds to a loss of only 1.45 dB. Inclusion in c_k of the two next-largest components $p_{-1} = p_3 = 2/3$ reduces $\mathcal{L}$ to 0.57 dB. With two additional coefficients $p_0 = p_2 = 1/2$ we arrive at a negligible loss of around 0.15 dB.

In this example, the magnitude of coefficients p_k determines the order in which new coefficients c_k are added. This approach will minimize $\mathcal{L}$ for any prescribed number of coefficients c_k, provided that data and noise are both uncorrelated.

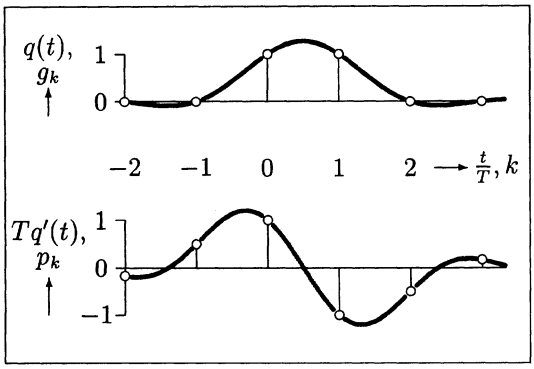


Figure 10.53: Duobinary response $q(t)$ without excess bandwidth and its derivative $Tq'(t)$. The corresponding sampled responses g_k and p_k are indicated with open circles.

Example 10.15 Consider a Duobinary $(1 + \mathcal{D})$ system response without excess bandwidth (Fig. 10.53). Here baud-rate sampling ($L = 1$) will do. The sampled derivative p_k has total energy $E_p = \frac{2\pi^2}{3} - 4$. The largest coefficients are $p_0 = 1$ and $p_1 = -1$; their combined energy falls 1.11 dB short of E_p. Inclusion of the two next-largest coefficients $p_{-1} = 0.5$ and $p_2 = -0.5$ in c_k reduces this loss to only 0.14 dB. The corresponding timing-recovery scheme is depicted in Fig. 10.54.

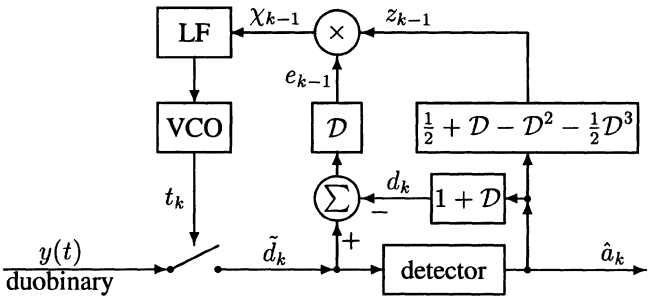


Figure 10.54: Near-optimum zero-forcing timing-recovery scheme for Duobinary $(1+\mathcal{D})$ system with white noise and no excess bandwidth.

In the decision-directed mode of operation, computation of $c_{-1}\hat{a}_{k+1}$ would involve a decision that has not yet been taken. A pure delay of one symbol interval in the computation of z_k serves to overcome this problem. The error sequence e_k must be delayed accordingly. The net effect will be a loop delay of 1 symbol interval.

10.7.5 Effect of the modulation code

By (10.38), the optimum dimensioning of ZF schemes is fully determined by the channel characteristics. The data spectrum $\mathcal{A}(e^{j2\pi\Omega})$ does not come into play, i.e. the applied modulation code is irrelevant. Efficiency, on the other hand, does depend on $\mathcal{A}(e^{j2\pi\Omega})$ and is normally best when the spectral content of the data is concentrated in the region where the channel is most reliable (as assessed, for example, by the spectral signal-to-noise ratio $\mathrm{SN}(\Omega)$). A limiting case is that of a periodic preamble. Here $\mathcal{A}(e^{j2\pi\Omega})$ becomes a line spectrum, and efficiencies γ_{ZF} and $\hat{\gamma}_{\mathrm{ZF}}$ become fully determined by C evaluated at the discrete set of frequencies where $\mathcal{A}(e^{j2\pi\Omega})$ is nonzero (see (10E.6) and (10.33)). At any other frequency we are free to choose W and C without affecting efficiency.

In the particular case that a_k is a sampled sinusoid of normalized frequency Ω_0, the optimum efficiency $\hat{\gamma}_{\mathrm{ZF}}$ is achieved whenever the phase of C coincides with that of P at frequency Ω_0 (see Problem 10.45). The amplitude of C is irrelevant (but should, of course, be nonzero). Even simple filters may meet this condition.

Example 10.16 Assume that the system response $q(t)$ is antisymmetric about $t = 0$. Then $p_k = Tq'(kT)$ will be symmetric about $k = 0$, i.e. $P(e^{j2\pi\Omega})$ will be real-valued. For any data frequency Ω_0 it suffices for $C(\Omega_0)$ to be real-valued. This is so, for example, for a degenerate filter in the form of a scaling factor c that is set to achieve the desired loop gain. If the point of symmetry is shifted in time across N symbol intervals then a pure delay of N symbol intervals must be added to c_k. The Class IV $(1 - \mathcal{D}^2)$ scheme of Fig. 10.50 is a case in point with $N = 1$.

Problem 10.44 is concerned with the situation that $q(t)$ is symmetric about $t = NT$. This is the case, for example, for typical Nyquist-1 functions. Problem 10.47 deals with responses that are symmetric about $t = (N + \frac{1}{2})T$, such as typical Duobinary $(1 + \mathcal{D})$ responses.

10.7.6 Oversampled ZF timing recovery

For systems with significant excess bandwidth, baud-rate sampling ($L = 1$) inevitably results in a significant detraction from ML performance. The only remedy is to resort to oversampling. The following two examples serve to show that this need not increase complexity dramatically.

Example 10.17 A Nyquist-1 system with raised-cosine roll-off can have at most 100% excess bandwidth, i.e. an oversampling factor $L = 2$ will do for timing recovery. Fig. 10.55 depicts the system response $q(t)$ and its derivative $Tq'(t)$ for a roll-off factor $\beta = 1$. Among all coefficients p_n, p_{-1} and p_1 have largest magnitude by far. Restriction of c_n to only these two coefficients results in a loss $\mathcal{L}$ of only 0.22 dB. With this choice, χ_n becomes zero at the decision instants ($n = 2k$), while $\chi_n = e_{2k-1}z_{2k-1} = (\tilde{d}_{2k-1} - a_{k-1}g_1 - a_k g_{-1})(a_{k-1}p_1 + a_k p_{-1})$ at the intermediate sampling instants ($n = 2k - 1$). Upon realizing that $g_{-1} = g_1 = \frac{1}{2}$ and that $p_{-1} = -p_1$ we arrive at Fig. 10.56. The scheme of Fig. 10.56 remains applicable when $\beta < 1$. Losses $\mathcal{L}$ increase gradually as β decreases and are in the order of 3 dB for $\beta = 0$.

The sampler in Fig. 10.56 alternates between the upper and lower positions on successive sampling instants t_n. Just as in baud-rate ZF schemes, TED outputs χ and the preceding se-

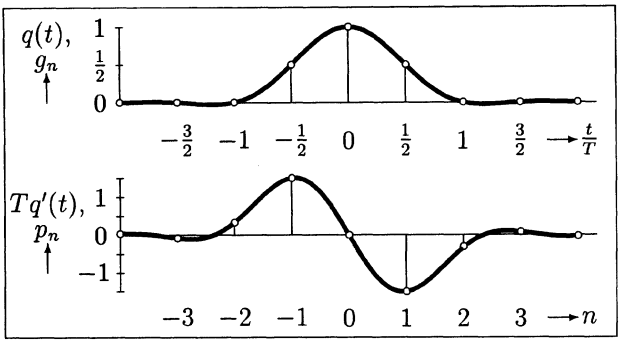


Figure 10.55: Raised cosine pulse $q(t)$ with roll-off factor $\beta = 1$ and its derivative $Tq'(t)$. Corresponding responses $g_n = q(nT/2)$ and $p_n = Tq'(nT/2)$ are indicated with open circles.

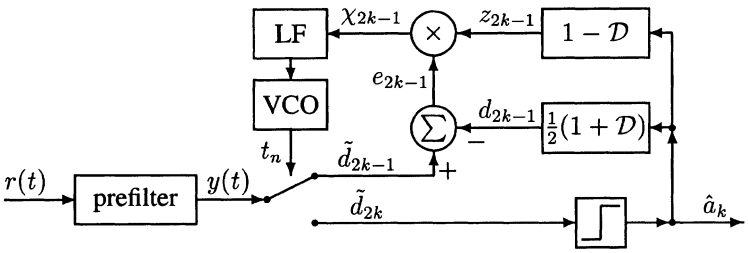


Figure 10.56: Near-optimum timing-recovery scheme for full-response system with roll-off factor $\beta = 1$ and white noise v_n. The operator $\mathcal{D}$ represents a delay of T seconds.

quences d, z and e are produced only once per symbol interval. Both z_{2k-1} and χ_{2k-1} are zero when $\hat{a}_{k-1} = \hat{a}_k$, i.e. on any data transition between the instants $k-1$ and k. This reflects the fact that timing information is mostly generated by transitions. More generally, nonzero TED outputs may be produced contingent upon the occurrence of data patterns for which significant timing information is expected. This idea is at the heart of the baud-rate selective-update schemes of [33] and [7].

Example 10.18 The Bi-phase modulation code, discussed earlier in Example 3.1 and Section 4.10, produces a transmitted signal $s(t) = \sum_{k=-\infty}^{\infty} a_k h_b(t - kT)$ with $h_b(t)$ as shown in Fig. 10.57. Due to bandwidth limitations, the system response $q(t)$ will be a 'rounded' version of $h_b(t)$. Bi-phase has spectral content well beyond the symbol rate $1/T$ and therefore an oversampling factor $L = 4$ is not overdone. Corresponding sampled responses $g_n = q(nT/4)$ and $p_n = Tq'(nT/4)$ are indicated in Fig. 10.57. Among all coefficients p_n, p_1 has the largest magnitude. This suggests that we restrict c_n to a single tap $c_1 = 1$, so that z_n and the TED output χ_n can be nonzero for $n = 4k+1$ only (Fig. 10.58). Decisions are taken at the instants $n = 4k$. Since $c_1 = 1$, the detected symbols $\hat{a}_k$ directly constitute the sequence z_n of Fig. 10.48 for $n = 4k+1$. From Fig. 10.57 it can be observed that $g_{4k+1} = 0$ for all k. Therefore $d_{4k+1} = 0$ and $e_{4k+1} = \tilde{d}_{4k+1}$. The VCO runs at $4/T$ and causes the lower and upper switches to be closed at the instants $n = 4k$ and $n = 4k+1$, respectively. Hence $y(t)$ is sampled in a non-uniform

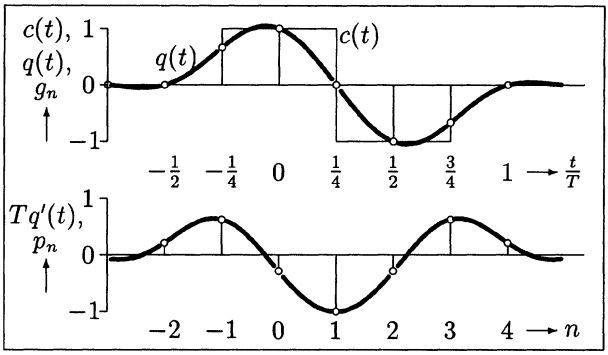


Figure 10.57: Symbol response $h_b(t)$ of Bi-phase along with system response $q(t)$ and its derivative $Tq'(t)$.

manner. The scheme can be simplified further at some expense to performance through addition of a slicer in the $\tilde{d}_{4k+1}$ path. Then e_{4k+1} is binary, and an exclusive-or gate suffices for the multiplication in Fig. 10.58 when a_k is binary. An all-digital variant of this simplified scheme, with added provisions to detect hangup, was first described by Wouda and Reijntjens [65].

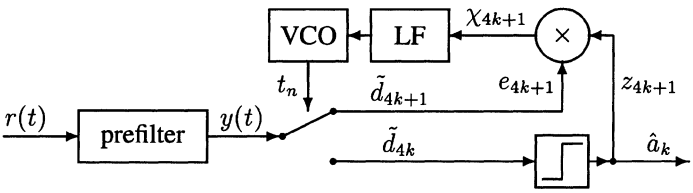


Figure 10.58: Zero-forcing timing-recovery scheme for Bi-phase.

10.7.7 Automatic adjustment of c_k

In practice the channel characteristics may not be accurately known *a priori*. A compromise choice for the coefficients c_k may yield good results when only $\mathcal{V}(e^{j2\pi\Omega})$ can vary (see Section 10.7.12). Variations of $P(e^{j2\pi\Omega})$ may be more cumbersome.

Example 10.19 Let $C(e^{j2\pi\Omega})$ be real-valued (c_k even), and assume that $P(e^{j2\pi\Omega})$ can become purely imaginary (p_k odd). Then the numerator in (10E.6) can vanish, and so can γ_{ZF}.

To avoid problems of this type, $C(e^{j2\pi\Omega})$ may be changed along with $P(e^{j2\pi\Omega})$. Let us consider a system without excess bandwidth. Then p_k becomes fully determined by g_k via the discrete-time analog of differentiation, i.e. $p_k = (g * x)_k$ where x_k is given by (10.34). A direct implication is that z_k may be formed by applying d_k to a filter with impulse response x_k. In practice a finite-length approximation of this impulse response must be used. Even simple approximations tend to yield good results. For

example, the differentiation may be replaced by a difference operation by retaining only the largest two coefficients $x_{-1} = 1$ and $x_1 = -1$. Correspondingly $z_k = d_{k+1} - d_{k-1}$ and $c_k = g_{k+1} - g_{k-1}$. A delay of one symbol interval is needed to arrive at a causal implementation. The error sequence e_k must be delayed accordingly (Fig. 10.59).

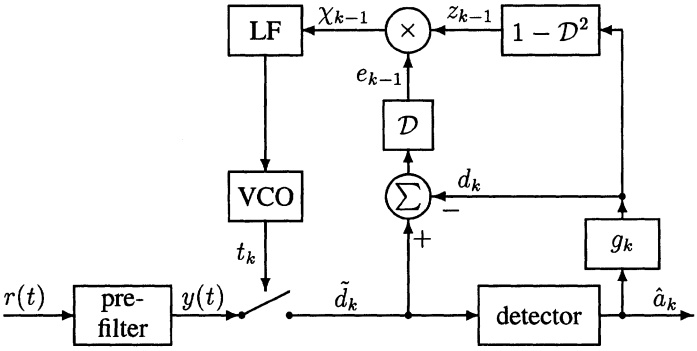


Figure 10.59: Approximation of the scheme of Fig. 10.48 for the situation that excess bandwidth is small and noise v_k is white.

It is worth noting that the sequence d_k is sometimes directly available at the output of the detector. This is so, for example, in partial-response systems with a precoder in the transmitter and a ternary threshold detector along with a memoryless inverse mapping in the receiver (see Chapter 6). The scheme of Fig. 10.59 becomes particularly simple for systems of this type.

Example 10.20 Let us consider the circuit of Fig. 10.59 for a full-response system with white noise v_k and negligible excess bandwidth. Here g_k degenerates ($g_k = \delta_k$, $p_k = x_k$), i.e. the circuit of Fig. 10.59 becomes a special case of that of Fig. 10.48, with two nonzero coefficients c_k ($c_{-1} = x_{-1} = 1$ and $c_1 = x_1 = -1$). With the aid of (10.42) it is seen that efficiency lags behind $\hat{\gamma}_{\mathrm{ZF}}$ by a factor $\mathcal{L} = [2\pi^2/3]/4 \simeq 1.645$. This corresponds to a loss of around 2.16 dB. Significantly smaller losses are encountered for partial-response systems (see Problem 10.39).

10.7.8 ZF timing recovery for decision feedback equalization

The decision feedback equalizer (DFE) of Chapter 6 contains a prefilter that suppresses noise and pre-cursive ISI (Fig. 10.60). The sampler output $\tilde{d}_k$ constitutes the input of a feedback detector and contains only trailing ISI when sampling occurs at the correct phase. In the detector, a feedback filter (FBF) operates on past symbol decisions $\hat{a}_{k-1}, \hat{a}_{k-2}, \ldots$ to produce an estimate $\hat{b}_k$ of the trailing ISI-component b_k of $\tilde{d}_k$. Subtraction of $\hat{b}_k$ from $\tilde{d}_k$ yields decision variables $\tilde{a}_k$, from which decisions $\hat{a}_k$ are obtained with a slicer. The desired sequence d_k is not available explicitly within the feedback detector. By definition, d_k would emerge at the input of this detector in the absence of noise and sampling-phase errors. The FBF output $\hat{b}_k$ constitutes the trailing component b_k of this desired input when past decisions are correct, as we assume. Similarly, the

current decision $\hat{a}_k = a_k$ constitutes the present component. There are no future components since pre-cursive ISI is absent. It follows that $d_k = \hat{a}_k + \hat{b}_k = a_k + b_k$ and $e_k = \tilde{d}_k - d_k = \tilde{a}_k - a_k$. This is true irrespective of the target response g_k, which is often determined via FBF adaptation and hence *a priori* unknown. The optimum impulse response c_k for the scheme of Fig. 10.48 is then also unknown. Availability of d_k allows us to construct a valid reference sequence z_k in spite of this uncertainty. Assume that noise is white and excess bandwidth is absent. Just as in Section 10.7.7, z_k may be obtained from d_k via the difference operator $\mathcal{D}^{-1} - \mathcal{D}$. In Fig. 10.60, a delay of one symbol interval is added for reasons of causality, and e_k is delayed accordingly. The net effect is a loop delay of one symbol interval.

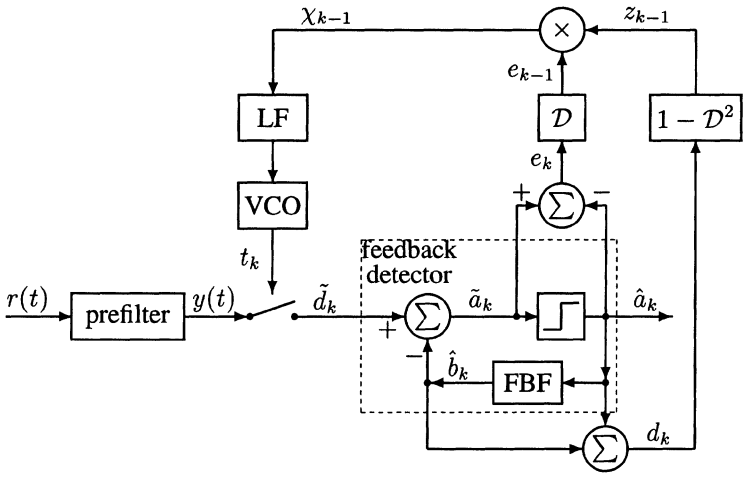


Figure 10.60: Decision feedback equalizer with baud-rate ZF timing-recovery scheme.

The scheme of Fig. 10.60 is not the only possible one. For example, any system response $q(t)$ that is suited for the DFE rises rapidly from 0 for $t = -T$ to a positive reference value for $t = 0$. Furthermore, $q(t)$ exhibits significant undershoot as t decreases from $-T$ when excess bandwidth is small. The first pre-cursor q_{-1}^{Δ} is then proportional to Δ for $|\Delta| \ll 1$. The implied timing function $\rho(\Delta) = q_{-1}^{\Delta} - g_{-1} = q_{-1}^{\Delta}$ is obtained with a single weight $c_{-1} = 1$ (compare (10.41)) and is at the heart of the *precursor-based* timing-recovery scheme that originates in [18] (see also [64, 3]). The scheme of Fig. 10.60 is somewhat more complicated than that of [18] but in return exploits basically all ISI components for timing recovery, as opposed to only the first precursor.

Example 10.21 By way of illustration we consider a Lorentzian recording channel with white noise (see Example 10.4). The prefilter in Fig. 10.60 is assumed to be a canonical whitened matched filter. This choice is optimum for the DFE and yields white noise v_k (see Chapter 6). The resulting responses $(h * f)(t)$, $g_k = (h * f)(kT)$ and $p_k = T(h * f)'(kT)$ are fully determined by the normalized information density t_{50}/T, and so is $\mathcal{L}$. We recall that $\mathcal{L}$ is the loss relative to an optimum ZF scheme with the same oversampling factor L. The factor $L = 1$ under consideration becomes suboptimum when excess bandwidth comes into play, i.e. at low

densities. In Fig. 10.61, losses are therefore re-normalized relative to an ML scheme, and an additional curve characterizes the loss of an optimum baud-rate ZF-scheme relative to ML. The scheme of Fig. 10.60 is clearly superior to precursor-based timing recovery and is virtually at par with the optimum baud-rate scheme. Losses with respect to ML are significant at low densities, where excess bandwidth is significant. Conversely, baud-rate operation permits close-to-ML performance at high densities.

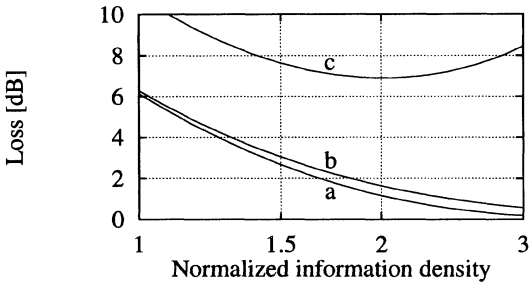


Figure 10.61: Loss relative to ML timing recovery versus normalized information density t_{50}/T for three ZF timing-recovery schemes. a. Optimum baud-rate ($L = 1$); b. Scheme of Fig. 10.60; c. Precursor-based timing recovery.

Exercise 10.9 Assume that the feedback filter and/or the phase characteristics of the prefilter are adapted under the control of e_k. Argue that interaction between adaptation and timing recovery may occur.

This exercise suggests that the scheme of Fig. 10.60 works best if the FBF is fixed. This undesirable constraint restricts the applicability of the scheme.

10.7.9 Selective update

Long runs of identical symbols a_k do not convey any timing information. As argued in Section 10.6.3, it is desirable that the TED ceases to deliver any output when such runs occur in order to prevent the sampling phase from drifting away randomly. This is possible by constraining c_n to have a zero at DC. In similar vein, c_n might have a zero at the Nyquist frequency $1/(2T)$ for systems that reject this frequency. (Cases in point are Duobinary $(1+\mathcal{D})$ and Class IV $(1-\mathcal{D}^2)$ partial-response systems.) What results is effectively a selective-update scheme akin to those of [33] and [7]. The accruing added robustness is only relevant when long worst-case patterns can indeed occur. In many instances the modulation code precludes such patterns (see Chapter 4). Appendix 10E.2 derives the finite-length impulse response c_n that minimizes $\mathcal{U}(0)$ subject to constraints on the TED gain K_d and the DC-component of c_n. The analysis is readily extended to constraints at other frequencies, but may not be needed when c_n is designed in the more intuitive fashion of the foregoing sections. For example, c_n is already DC-free in the scheme of Fig. 10.56, and has zeros at both DC and $1/(2T)$ in the schemes of Figs. 10.59 and 10.60.

10.7.10 Non-error-based schemes

For loop bandwidths much smaller than the symbol rate $1/T$, the loop of Fig. 10.48 is mainly driven by the DC-component of the TED output χ_n. Now $\chi_n = (e*w)_n z_n = (\tilde{d}*w)_n z_n - \xi_n$ where $\xi_n \triangleq (d*w)_n z_n = (a*g*w)_n(a*c)_n$. Appendix 10E.3 reveals that ξ_n is DC-free irrespective of a_k if and only if $(g_-*w_-*c)_{kL}$ is antisymmetric about $k=0$. Example 10.22 below serves to illustrate that this condition is often met. Omission of ξ from χ as in Fig. 10.62 will, then, affect neither $\rho(\Delta)$ nor the power spectral density of χ_n at DC. Hence, provided that loop bandwidth is much smaller than $1/T$, pattern-dependent jitter will be negligible and tracking properties will remain essentially unaffected.

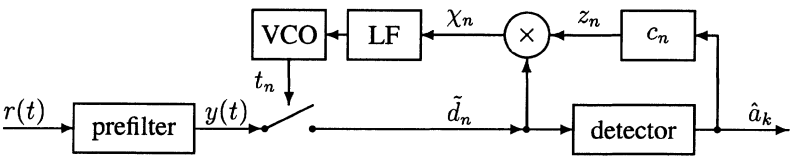


Figure 10.62: Non-error-based ZF timing-recovery scheme.

A related issue is that of gain variations as induced, for example, by a non-perfect AGC. Assume that the system has gain $1+\alpha$ where α accounts for a gain mismatch. Denote the resulting error by e_n^α. Clearly $e_n^\alpha = (1+\alpha)\tilde{d}_n - d_n = (1+\alpha)e_n + \alpha d_n$ where e_n is the nominal error (i.e. the error for $\alpha = 0$). Correspondingly $\chi_n^\alpha = (1+\alpha)\chi_n + \alpha\xi_n$. Gain variations, therefore, affect the TED gain K_d linearly and induce a pattern-dependent disturbance component $\alpha\xi_n$ whose effect is bound to be small when $(g_* * w_* * c)_{kL}$ is an odd function of k.

Example 10.22 Examples 10.13 and 10.18 describe schemes without error filter for which g is antisymmetric about a given instant and c is symmetric about the same instant. In the schemes of Examples 10.15 and 10.17 these symmetries are reversed. The baud-rate scheme of Fig. 10.60 has $c_k = g_{k+1} - g_{k-1}$, whence $(g_* * w_* * c)_k = (g_* * c)_k = (g_* * g)_{k-1} - (g_* * g)_{k+1}$. In all of these cases $(g_* * w_* * c)_{kL}$ is antisymmetric about $k=0$. This condition is not met, for example, by the precursor-based scheme of Section 10.7.8.

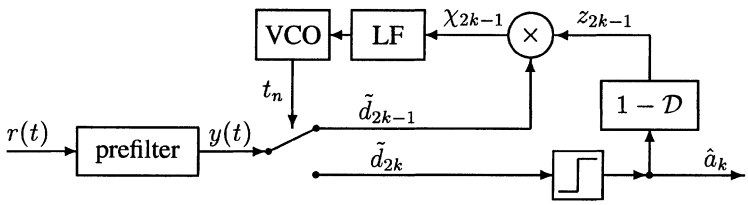


Figure 10.63: Non-error-based oversampled ZF timing-recovery scheme for full-response system.

Example 10.23 The non-error-based counterpart of the scheme of Fig. 10.56 is depicted in Fig. 10.63. The roots of this scheme can be traced back to work of Lindsey, Tausworthe and Simon [58] (see also [48]). The version of Fig. 10.63 and a non-decision-directed variant are usually accredited to Gardner [26]. Both were later recognized as near-ML schemes [45]. A numerical study that includes the effect of pattern-dependent jitter may be found in [6].

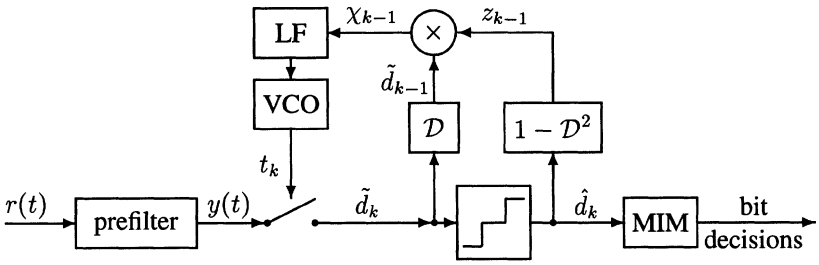


Figure 10.64: Class IV partial-response receiver with ternary threshold detector and non-error-based ZF timing-recovery scheme based on the one of Fig. 10.59.

Example 10.24 Consider a Class IV partial-response system with a precoder in the transmitter and a ternary threshold detector in conjunction with a memoryless inverse mapping (MIM) in the receiver (see Chapter 6 for more details about this topology). Then decisions $\hat{d}_k$ with respect to the partial-response sequence $d_k = a_k - a_{k-2}$ are directly available at the output of the detector, and the non-error-based counterpart of Fig. 10.59 becomes particularly simple (Fig. 10.64). The scheme of Fig. 10.64 is closely related to the one of [14], which is studied in Problem 10.52.

10.7.11 Baud-rate schemes of [44]

The idea of using some feature of the sampled system response as a timing function that is estimated by correlation techniques can be traced back to the classical paper of Mueller and Müller [44]. In [44] the TED output χ_k is obtained by combining N consecutive samples y_k or e_k in a matrix-type operation with N consecutive data symbols a_k. For $N = 1$ we are in the realm of the ZF techniques considered so far. In order to convey the flavor of the approach we will treat the case $N = 2$ by means of an example. Larger values of N are less practical because the schemes become progressively more complicated to analyze and implement, while performance is no better than that of the foregoing ZF schemes.

Example 10.25 Let us consider a full-response system with a system response $q(t)$ in the form of a raised-cosine pulse with roll-off factor $\beta = 0$. This pulse and its derivative $Tq'(t)$ are shown in Fig. 10.65. The sampled derivative $p_k = Tq'(kT)$ has largest magnitude for $k = \pm 1$. This suggests that we take $p_{-1}q_{-1}^{\Delta} + p_1 q_1^{\Delta}$ or, equivalently, $q_{-1}^{\Delta} - q_1^{\Delta}$ as our timing function. (We recall that q_k^{Δ} represents the kth sample of the system response when sampled at a normalized phase error Δ, i.e. $q_k^{\Delta} = q((k+\Delta)T)$.) This is the same function that underlies the full-response system of Example 10.20 (Fig. 10.59). In Section 10.7.4 we have seen that an instantaneous

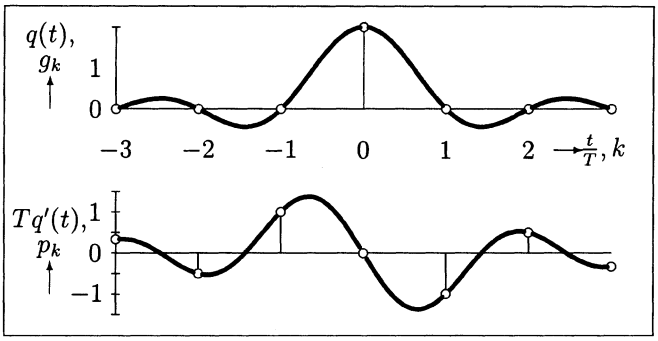


Figure 10.65: Raised-cosine pulse $q(t)$ with roll-off factor $\beta = 0$ and its derivative $Tq'(t)$. The corresponding sampled responses $g_k = q(kT)$ and $p_k = Tq'(kT)$ are depicted with open circles.

estimate $\tilde{q}_1^\Delta$ of q_1^Δ may be formed according to $\tilde{q}_1^\Delta = e_k a_{k-1}$. Similarly, the product $e_k a_{k+1}$ may serve as an instantaneous estimate of q_{-1}^Δ. For reasons of causality we must delay this product across one symbol interval to arrive at an instantaneous estimate $\tilde{q}_{-1}^\Delta = e_{k-1} a_k$. The timing function may, therefore, be estimated according to $\chi_k = \tilde{q}_{-1}^\Delta - \tilde{q}_1^\Delta = e_{k-1} a_k - e_k a_{k-1}$. This function can be recast in terms of the sampler output $\tilde{d}_k$ by noting that $e_k = \tilde{d}_k - a_k$ and $e_{k-1} = \tilde{d}_{k-1} - a_{k-1}$. After some elementary manipulations it follows that

$$\chi_k = \tilde{d}_{k-1} a_k - \tilde{d}_k a_{k-1}. \tag{10.43}$$

The estimate of (10.43) was first proposed in [44]. A TED that is based on it is shown in Fig. 10.66. In the figure, decisions $\hat{a}_k$ and $\hat{a}_{k-1}$ replace the actual data symbols a_k and a_{k-1}. Performance is comparable to that of the scheme of Example 10.20. A very similar TED for Class IV partial-response systems was described in [14]. Problem 10.52 is concerned with the timing function and efficiency of that TED.

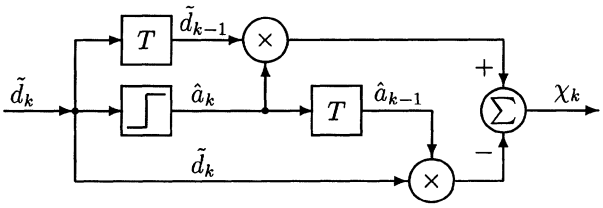


Figure 10.66: Timing-error detectror according to [44] for baud-rate timing recovery in full-response system.

10.7.12 Performance comparison

We reconsider the idealized recording systems of Example 10.4 (see also Section 10.6.5). For periodic data in the form of a sampled sinusoid, baud-rate

efficiencies γ_{ZF} approach $\hat{\gamma}_{BR}$ at high SNRs subject to mild conditions on c_k and w_k (see Section 10.7.5). For this reason the numerical results for periodic data of Section 10.6.5 remain applicable, and we restrict attention here to uncorrelated data a_k.

Lorentzian channel equalized to Class IV: We only consider the target response without excess bandwidth. Fig. 10.67 illustrates how the simple schemes of Figs. 10.50 and 10.59 match up against γ_{ML} and $\hat{\gamma}_{BR}$. Even though their dimensioning is fixed, both schemes approach $\hat{\gamma}_{BR}$ within a few dB across $D \in [1, 3]$.

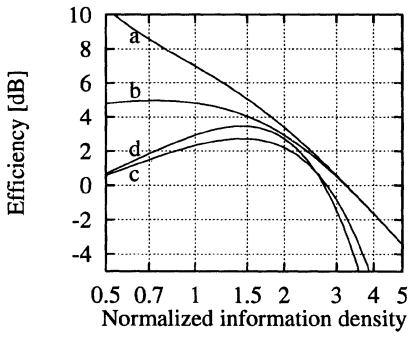


Figure 10.67: Efficiencies for a Lorentzian channel equalized to Class IV. Data is uncorrelated, excess bandwidth is absent. a. Maximum-likelihood (γ_{ML}); b. optimum baud-rate ($\hat{\gamma}_{ZF}$); c. scheme of Fig. 10.50; d. scheme of Fig. 10.59.

Idealized optical recording channel equalized to Duobinary: Just as in Section 10.6.5, we consider only the target response without excess bandwidth. Fig. 10.68 depicts corresponding efficiencies. Again even the simplest schemes approach $\hat{\gamma}_{ZF}$ within a few dB for densities of practical interest.

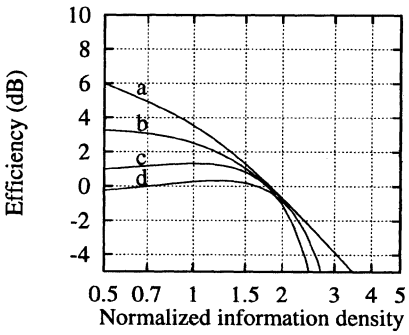


Figure 10.68: Efficiencies for an idealized optical recording channel equalized to Duobinary $(1 + \mathcal{D})$. Data is uncorrelated, excess bandwidth is zero. a. Maximum-likelihood (γ_{ML}); b. optimum baud-rate ($\hat{\gamma}_{ZF}$); c. scheme of Fig. 10.54; d. scheme of Fig. 10.59.

Learning curves: We finally illustrate the dynamic behavior of some of the foregoing schemes for a Lorentzian system at a normalized information density $D = 2.5$ and an SNR of 10 dB. A first-order loop as in Example 9.7 is used. For each scheme and type of data, K_t is adjusted via K_l to 0.02. The initial sampling-phase error Δ is set at 10% of a symbol interval and the evolution of Δ over time is recorded. This experiment is repeated 1024 times. Mean $\overline{\Delta}$ and variance σ_Δ^2 of Δ across all experiments are then computed for each instant k. Some of the resulting learning curves are compared in Fig. 10.69. As anticipated, the three considered schemes have virtually the same learning curves. The dashed lines in the σ_Δ curves were computed based on (9.2) along with Figs. 10.46 and 10.67. Close agreement is observed between theory and simulations.

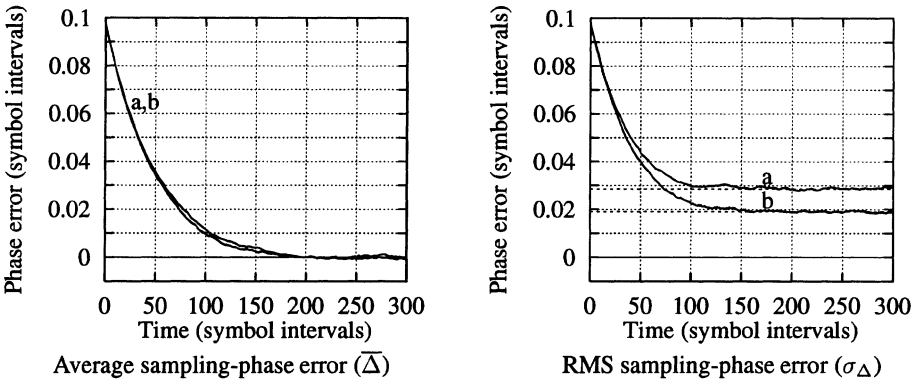


Figure 10.69: Learning curves of several timing-recovery schemes for a Lorentzian recording channel equalized to a Class IV $(1 - \mathcal{D}^2)$ partial response with no excess bandwidth. The channel has normalized information density $D = 2.5$ and an SNR of 10 dB. Covered are the schemes of Fig. 10.59, Fig. 10.42 with two coefficients $w_{-1} = 1, w_1 = -1$, and Fig. 10P.3. Both uncorrelated data (a.) and a binary $4T$ pattern (b.) are considered. The dashed lines in the σ_Δ picture depict theoretical RMS jitter levels.

APPENDIX 10A

Analysis of maximum-likelihood timing recovery

10A.1 Analysis of the data-aided maximum-likelihood timing-error detector

We first determine the transfer characteristic $\rho(\Delta) = E[\chi_k]$ of the TED of Fig. 10.3 and its slope K_d in the origin ($\Delta = 0$). From (10.6) one has $E[\chi_k] = (R^a * q_-^\Delta)_0$ where R_k^a denotes the data autocorrelation function and the subscript '$_-$' denotes time-

reversal. In frequency-domain notation this becomes

$$\rho(\Delta) = \int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega}) Q^{\Delta}(e^{j2\pi\Omega}) d\Omega \tag{10A.1}$$

where $\mathcal{A}(e^{j2\pi\Omega})$ denotes the data PSD and $Q^{\Delta}(e^{j2\pi\Omega})$ is the discrete-time Fourier transform of q_k^{Δ}. Now q_k^{Δ} is a sampled version of the system response $s(t)$ up until the matched filter according to $q_k^{\Delta} = s((k+\Delta)T)$. Accordingly, $Q^{\Delta}(e^{j2\pi\Omega})$ is a folded version of the Fourier integral $S(\Omega)$ of $s(t)$:

$$Q^{\Delta}(e^{j2\pi\Omega}) = \frac{1}{T} \sum_{n=-\infty}^{\infty} S(\Omega + n) e^{j2\pi\Delta(\Omega+n)}. \tag{10A.2}$$

Upon invoking the fact that $\mathcal{A}(e^{j2\pi\Omega})$ is periodic in Ω with period 1 we find with (10A.1)

$$\rho(\Delta) = \frac{1}{T} \int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega}) S(\Omega) e^{j2\pi\Delta\Omega} d\Omega. \tag{10A.3}$$

It remains to specify the system transfer function $S(\Omega)$. This function is the product of the channel transfer function $H(\Omega)$, the transfer function $H^*(\Omega)/\mathcal{N}(\Omega)$ of the matched filter, and a factor $j2\pi\Omega$ that accounts for scaling and differentiation. Thus $S(\Omega) = j2\pi\Omega|H(\Omega)|^2/\mathcal{N}(\Omega) = j2\pi\Omega T\mathrm{SN}(\Omega)$ where the last step is due to (10.8). Hence

$$\rho(\Delta) = \int_{-\infty}^{\infty} j2\pi\Omega \mathcal{A}(e^{j2\pi\Omega}) \mathrm{SN}(\Omega) e^{j2\pi\Delta\Omega} d\Omega. \tag{10A.4}$$

Now $\mathcal{A}$ and SN are both real-valued and symmetric functions of Ω. Correspondingly

$$\rho(\Delta) = -\int_{-\infty}^{\infty} 2\pi\Omega \mathcal{A}(e^{j2\pi\Omega}) \mathrm{SN}(\Omega) \sin(2\pi\Delta\Omega) d\Omega. \tag{10A.5}$$

This is an antisymmetric function of Δ. The slope K_d in the origin ($\Delta = 0$) is

$$K_d = -\int_{-\infty}^{\infty} (2\pi\Omega)^2 \mathcal{A}(e^{j2\pi\Omega}) \mathrm{SN}(\Omega) d\Omega. \tag{10A.6}$$

We next consider the noise component $u_k = a_k v_k$ of the TED output. The PSD of u_k at DC amounts to $\mathcal{U}(1) = \int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega}) \mathcal{V}(e^{j2\pi\Omega}) d\Omega$ where $\mathcal{V}(e^{j2\pi\Omega})$ denotes the PSD of v_k. Since v_k is a sampled version of the noise behind the differentiator, its PSD is a folded version of the noise PSD $\mathcal{N}(\Omega)|j2\pi\Omega H^*(\Omega)/\mathcal{N}(\Omega)|^2 = (2\pi\Omega)^2 T\mathrm{SN}(\Omega)$ behind the differentiator, i.e.

$$\mathcal{V}(e^{j2\pi\Omega}) = \sum_{n=-\infty}^{\infty} [2\pi(\Omega + n))^2 \mathrm{SN}(\Omega + n]. \tag{10A.7}$$

Upon realizing that $\mathcal{A}(e^{j2\pi\Omega})$ is periodic in Ω with period 1 we find after some elementary manipulations that

$$\mathcal{U}(1) = \frac{1}{T^2}\int_{-\infty}^{\infty}(2\pi\Omega)^2\mathcal{A}(e^{j2\pi\Omega})\mathrm{SN}(\Omega)d\Omega. \tag{10A.8}$$

It remains to note that the input-referred noise $\theta_k = u_k/K_d$ has PSD $P_\theta(e^{j2\pi\Omega}) = \mathcal{U}(e^{j2\pi\Omega})/K_d^2$. With (10A.6), (10A.8) and (9.3) this validates (10.15).

10A.2 Efficiency factors for data-aided maximum-likelihood timing recovery on a Lorentzian channel

From Chapter 3, we recall that the Lorentzian model for digital magnetic recording channels has a spectral signal-to-noise ratio $\mathrm{SN}(\Omega) = (1/N_0)\sin^2(\pi\Omega)\exp\{-2\pi D|\Omega|\}$ and an average signal-to-noise ratio

$$\mathrm{SNR} = \int_{-\infty}^{\infty}\mathrm{SN}(\Omega)d\Omega = \frac{1}{N_0}\frac{1}{2\pi D}\frac{1}{1+D^2}, \tag{10A.9}$$

where N_0 denotes the power spectral density of the noise and D is a normalized measure of the information density. For this model, the efficiencies of (10.16) and (10.17) may be determined analytically. In particular, twofold differentiation of $\mathrm{SN}(\Omega)$ with respect to D gives a function $C\Omega^2\mathrm{SN}(\Omega)$, where C is a constant. The integral of this function, which determines γ for uncorrelated data, may be computed by differentiating (10A.9) two times with respect to D. After some elementary manipulations it is found that

$$\gamma = \frac{2+6D^2+12D^4}{D^2(1+D^2)^2}. \tag{10A.10}$$

Similarly, twofold differentiation with respect to D of the folded signal-to-noise ratio $\mathrm{FSN}(e^{j2\pi\Omega}) \triangleq \sum_{n=-\infty}^{\infty}\mathrm{SN}(\Omega+n)$ leads to a variant of the infinite sum of (10.17). In Appendix 3A.1 it was shown that $\mathrm{FSN}(e^{j2\pi\Omega}) = [\sin^2\pi\Omega\cosh 2\pi D(0.5 - |\Omega|)]/(N_0\sinh\pi D)$. Upon differentiating this expression two times with respect to D we find that

$$\gamma = \frac{D(1+D^2)(1+\cosh^2(\pi D/2)}{(2/\pi)^3\sinh^3(\pi D/2)} \tag{10A.11}$$

for the $1/(4T)$ pattern '$\ldots++--++--\ldots$', and

$$\gamma = \frac{2D(1+D^2)(1+\cosh^2\pi D)}{\sinh^3\pi D/\pi^3} \tag{10A.12}$$

for the $1/(2T)$ pattern '$\ldots+-+-+-+-\ldots$'.

10A.3 Derivation of the non-data-aided maximum-likelihood timing-error detector

Let us assume that a_k is a *finite* message of N symbols $a_0, \ldots, a_{N-1}$, and that the timing-recovery subsystem may observe the received signal $r(t) = \sum_{k=0}^{N-1} a_k h(t-(k+\phi)T) + n(t)$ as long as it desires before producing an estimate $\tilde{\phi}$ of ϕ. Because it knows the encoding rule, the subsystem may form a list of all M sequences $a_k^0, \ldots, a_k^{M-1}$ $(k = 0, \ldots, N-1)$ that may have been transmitted. For each of these, it may compute a conditional likelihood function $L(a_k^i, \psi)$ that will be specified soon. This function indicates how likely the phase ψ is for the received signal $r(t)$ at hand if a_k^i were transmitted. Since the actual data sequence a_k is unknown, the best misadjustment indicator that the subsystem can construct is the average of L over all M sequences a_k^i, or any monotonic function of this average. For Gaussian noise one often arrives at comparatively simply expressions by considering the *log-likelihood function*, which is the logarithm of this average. The log-likelihood function is given by

$$\Lambda(\psi) \triangleq \ln\{\sum_{i=0}^{M-1} p_i L(a_k^i, \psi)\}, \tag{10A.13}$$

where p_i is the *a priori* probability of a_k^i. For uncorrelated binary data we have $M = 2^N$ and $p_i = 2^{-N}$ for all i, whence

$$\Lambda(\psi) = \ln\{2^{-N} \sum_{i=0}^{2^N-1} L(a_k^i, \psi)\}. \tag{10A.14}$$

Assume that the channel has white noise $n(t)$ with power spectral density N_0. In [28] it is shown for any channel impulse response $h(t)$ and any data sequence a_k^i, $k = 0, \ldots, N-1$ that

$$L(a_k^i, \psi) = \exp\{-\frac{1}{2N_0} \int_{-\infty}^{\infty} |r(t) - x^i(t - \psi T)|^2 dt\}, \tag{10A.15}$$

where

$$x^i(t) = \sum_{k=0}^{N-1} a_k^i h(t - kT) dt. \tag{10A.16}$$

If a_k^i were transmitted and the channel would have delay ψT, then the noiseless channel output $x^i(t - \psi T)$ would typically match $r(t)$ closely, and L would be large. This is intuitively reasonable.

Exercise 10.10 Generalize (10A.14) and (10A.16) to arbitrary channels with a flat folded signal-to-noise ratio $\mathrm{FSN}(e^{j2\pi\Omega})$.

We now restrict attention to uncorrelated binary data with $a_k \in \{-1, 1\}$ and a channel with a flat folded signal-to-noise ratio $\mathrm{FSN}(e^{j2\pi\Omega}) = \sum_{n=-\infty}^{\infty} \mathrm{SN}(\Omega + n)$. Let

$y(t)$ be the output of a matched filter operating on $r(t)$, i.e. $y(t) = (r * w)(t)$ with $W(\Omega) = H^*(\Omega)/\mathcal{N}(\Omega)$, and assume that $y(t)$ is sampled at instants $t_k = (k + \psi)T$. For this situation we will show that a log-likelihood function for non-data-aided phase estimation is

$$\Lambda(\psi) = \sum_{k=0}^{N-1} \ln \cosh y(t_k). \tag{10A.17}$$

We first consider the situation that noise is white with power spectral density $N_0 = 1$ and that the channel autocorrelation function $(h * h_-)(t)$ is a Nyquist-1 function. Expansion of the integral in (10A.15) yields

$$\int_{-\infty}^{\infty} |r(t) - x^i(t - \psi T)|^2 dt = \int_{-\infty}^{\infty} r^2(t)dt - 2\int_{-\infty}^{\infty} r(t)x^i(t - \psi T)dt + \int_{-\infty}^{\infty} (x^i)^2(t)dt. \tag{10A.18}$$

The first integral on the right hand side does not depend on i. It is not difficult to show that this is also true for to the third integral when $(h * h_-)(t)$ is Nyquist-1, as assumed. Upon denoting the sum of both integrals as a constant C and using (10A.16) to recognize the second integral as the sampled output $y(t_k) = (r * h_-)((k+\psi)T)$ of a matched filter, we get

$$\int_{-\infty}^{\infty} |r(t) - x^i(t - \psi T)|^2 dt = C - 2\sum_{k=0}^{N-1} a_k^i y(t_k). \tag{10A.19}$$

With (10A.15) this allows the conditional likelihood function $L(a_k^i, \psi)$ to be written as

$$L(a_k^i, \psi) = C' \exp\{\sum_{k=0}^{N-1} a_k^i y(t_k)\}, \tag{10A.20}$$

where $C' \triangleq \exp\{-C/2\}$. Upon using this result in (10A.14) we find after some elementary manipulations that

$$\Lambda(\psi) = \ln[\sum_{i=0}^{2^N-1} \prod_{k=0}^{N-1} \frac{1}{2} \exp\{a_k^i y(t_k)\}], \tag{10A.21}$$

where an irrelevant additive constant $\ln C'$ has been dropped. It remains to write the term in square brackets as a product of cosh terms. This is easily done by induction with respect to N, making use of the fact that the summation over i covers all binary sequences of length N. The result is (10A.17). Extension of this reasoning to channels with colored noise and a flat folded signal-to-noise ratio is left as an exercise to the reader. A likelihood indicator conforming to (10A.17) is shown in Fig. 10A.1.
To find the ML NDA TED, we differentiate (10A.17) with respect to ψ. The result is

$$\dot{\Lambda}(\psi) = T\sum_{k=0}^{N-1} y'(t_k) \tanh y(t_k), \tag{10A.22}$$

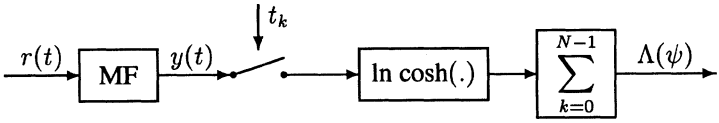


Figure 10A.1: Non-data-aided likelihood indicator for systems with uncorrelated binary data and a flat folded signal-to-noise ratio.

where "$\dot{}$" denotes time-derivative. Of particular interest is the behavior of $\dot{\Lambda}(\psi)$ in the vicinity of the optimum sampling phase, i.e. for $\psi \simeq \phi$. In Section 3.10.1 we have seen that for channels with a flat folded signal-to-noise ratio FSN($e^{j2\pi\Omega}$), the system impulse response $(h * w)(t)$ is a Nyquist-1 function with amplitude $(h * w)(0) = \text{SNR}$. Consequently $y((k+\psi)T) \simeq a_k\text{SNR}$ when $\psi \simeq \phi$ and noise is sufficiently small (SNR $\gg$ 1). Now $\tanh(\pm\text{SNR}) \simeq \pm\frac{\pi}{2}$ for any SNR of practical interest. For example, even for SNR = 6 dB, the error in replacing $\tanh(a_k\text{SNR})$ by $\frac{\pi}{2}\text{sgn}(a_k)$ is less than 0.1%. We may, then, replace the tanh operator in (10A.22) by the sign operator that produces decisions $\hat{a}_k$ with respect to a_k (the constant gain factor $\pi/2$ may be neglected because it affects signal and noise equally). For the system under study this bit-by-bit detector is in fact optimum because ISI is absent and the noise component of the detector input signal is uncorrelated (see Section 3.10.1). We arrive at the circuit of Fig. 10A.2, which should be compared with the DA TED of Fig. 10.3. As anticipated, the only difference is that locally generated decisions $\hat{a}_k$ are used instead of actual data symbols a_k. Whenever decisions are predominantly correct, i.e. for SNRs of practical interest, performance will coincide with that of the DA scheme, as analyzed in Section 10.2.3. For SNRs that are too low to be practical (say SNR $\leq$ 0 dB) we may use the approximation $\tanh y \simeq y$ in (10A.22). Upon realizing that $y(t)dy(t)/dt = \frac{1}{2}dy^2(t)/dt$ for any signal $y(t)$, we find that

$$\dot{\Lambda}(\psi) \simeq \frac{T}{2}\sum_{k=0}^{N-1}\frac{dy^2(t)}{dt}\Big|_{t=t_k}. \tag{10A.23}$$

The corresponding TED is shown in Fig. 10.6.

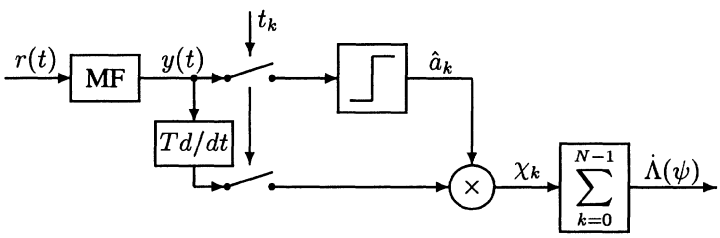


Figure 10A.2: Decision-directed maximum-likelihood timing-error detector for systems with uncorrelated binary data and a flat folded signal-to-noise ratio.

For the sake of completeness we note that a similar analysis may be carried out for multilevel and Gaussian data [28, 22]. The resulting likelihood functions are slightly different from the one derived here, but the main conclusions do not change.

APPENDIX 10B

Analysis of timing recovery based on threshold crossings

10B.1 Transfer characteristic

In what follows we derive the transfer characteristic of the TED of Fig. 10.7. Particular attention is paid to how this characteristic is affected by the form of the clock signal $c(t)$ and of the pulses delivered by the pulse former. We first introduce some notation. Assume that the channel introduces a delay of ϕ symbol intervals T. Then the timing tone $s(t)$ may be written

$$s(t) = -1 + \sum_{k=-\infty}^{\infty} d_k g(t - (k + \phi)T). \tag{10B.1}$$

Here d_k is a transition indicator. Specifically, $d_k = 1$ whenever a transition occurs in the kth symbol interval (i.e. when $a_k = -a_{k-1}$); otherwise $d_k = 0$. The function $g(t)$ is nominally a rectangular pulse of width $T/2$ that starts at $t = 0$ and has amplitude 2. Then $s(t) \in \{-1, 1\}$ as in Fig. 10.8.

Exercise 10.11 Use Fig. 10.8 to verify (10B.1).

Denote by $\overline{d}$ the average number of zero crossings of $\hat{y}(t)$ per symbol interval. Clearly $0 \leq \overline{d} \leq 1$. For uncorrelated binary data one has $\overline{d} = 0.5$.

Denote the DC value of the clock signal $c(t)$ by $\overline{c}$. In Section 10.3.1 we have assumed that this clock signal has a phase that can be controlled. To make the role of this phase explicit, we express $c(t)$ as a shifted version of a reference clock $c_r(t)$ with a *fixed* phase according to $c(t) = c_r(t - \psi T)$, where ψ denotes the clock phase. Along with the pulse $g(t)$, this reference clock determines a function

$$\xi^{\Delta}(t) \triangleq g(t)c_r(t - \Delta T) \tag{10B.2}$$

where Δ denotes the clock phase error normalized in symbol intervals T (see Figs. 10.8 and 10.9).

Exercise 10.12 Show that $\xi^{\Delta}(t)$ can be at most $T/2$ seconds long for the situation of Fig. 10.8.

Let us assume that $\xi(t)$ has a duration of almost T seconds. Then the TED of Fig. 10.7 has transfer characteristic

$$\rho(\Delta) = -\overline{c} + \overline{d} \int_{-0.5}^{0.5} \xi^{\Delta}(\psi T) d\psi \tag{10B.3}$$

In order to validate this expression we first note that multiplication of $s(t)$ with the clock signal yields the TED output $\chi(t)$, whose average value is the TED transfer characteristic. Clearly

$$\chi(t) = s(t)c(t) = -c(t) + \sum_{k=-\infty}^{\infty} d_k g(t - (k+\phi)T)c_r(t - \psi T). \tag{10B.4}$$

Now c_r is periodic in t with period T. For this reason

$$\begin{aligned} \chi(t) &= -c(t) + \sum_{k=-\infty}^{\infty} d_k g(t - (k+\phi)T)c_r(t - (k+\psi)T) \\ &= -c(t) + \sum_{k=-\infty}^{\infty} d_k \xi(t - (k+\phi)T) \end{aligned} \tag{10B.5}$$

where the last step is due to (10B.2). The average value of z determines the TED transfer characteristic $\rho(\Delta)$. The first term in the right hand side has average $-\bar{c}$. The infinite summation describes a sequence of T-spaced pulses with random amplitude d_k. Since pulses are non-overlapping, it suffices to average ξ over a single period and to take the average of d. This leads to (10B.3).

10B.2 Slope of transitions

We now focus on the average slope μ of an upgoing transition at the ideal sampling instant $\hat{t}_k$ (see Fig. 10.14). In the absence of noise, the signal $y(t)$ according to (10.18) has derivative

$$y'(t) = \sum_{k=-\infty}^{\infty} a_k q'(t - kT), \tag{10B.6}$$

where $q'(t)$ denotes the derivative of $q(t)$. For $t = \hat{t}_k = (k - 0.5)T$ this becomes

$$y'(\hat{t}_k) = \sum_{i=-\infty}^{\infty} a_i q'((k - i - 0.5)T). \tag{10B.7}$$

Since we are considering an upgoing transition, we must have $a_{k-1} = -1$ and $a_k = 1$. The other data symbols are not prescribed, and their average effect on y' will be zero for most coding rules of practical interest. The average slope of y at $t = \hat{t}_k$ is then given by $\mu = q'(-0.5T) - q'(0.5T)$. In frequency-domain notation this may be written

$$\mu = \frac{2}{T}\int_{-\infty}^{\infty} \frac{2\pi\Omega}{T} \sin \pi\Omega Q(\Omega) d\Omega. \tag{10B.8}$$

Example 10.26 Let $q(t)$ be a raised cosine pulse with roll-off factor $\beta = 1$, so that

$$Q(\Omega) = \begin{cases} T\cos^2 \frac{\pi}{2}\Omega & \text{for } |\Omega| \le 1, \\ 0 & \text{for } |\Omega| \ge 1. \end{cases} \tag{10B.9}$$

Then $\mu = (8/T)\int_0^1 \pi\Omega \sin(\pi\Omega)\cos^2(\pi\Omega/2)d\Omega = 3/T$. A slope of this order is indeed visible in the $\beta = 1$ eye pattern of Fig. 10.12.

10B.3 Effect of threshold-crossing jitter on TED transfer characteristic

To analyze the effect of random shifts of the threshold crossings, we assume that the phase ϕ at which a crossing occurs is a random variable with some distribution function $p(\phi)$. The average phase $E[\phi]$ is denoted $\overline{\phi}$.

Since ϕ is a stochastic variable, any function of ϕ will also be stochastic. This applies in particular to the TED output ρ due to a single threshold crossing. This output has expected value $\overline{\rho} \triangleq \int_\phi \rho(\psi - \phi)p(\phi)d\phi$. This may be recognized as the convolution of $\rho(\phi)$ and $p(\phi)$. Since $\rho(\phi)$ is periodic in ϕ, $\overline{\rho}$ will be periodic in ψ, or, equivalently, in $\Delta \triangleq \psi - \overline{\phi}$. To stress this fact we shall henceforth add an argument Δ to $\overline{\rho}$.

Example 10.27 Let ϕ be uniformly distributed in an interval of width $W \in [0, 1]$ centered around $\overline{\phi}$. Then

$$\overline{\rho}(\Delta) = \frac{1}{W}\int_{\overline{\phi}-\frac{W}{2}}^{\overline{\phi}+\frac{W}{2}} \rho(\psi - \phi)d\phi = \frac{1}{W}\int_{\Delta-\frac{W}{2}}^{\Delta+\frac{W}{2}} \rho(x)dx. \tag{10B.10}$$

This integral is easily evaluated for triangular and sinusoidal characteristics $\rho(x)$. The result is shown in Fig. 10B.1.

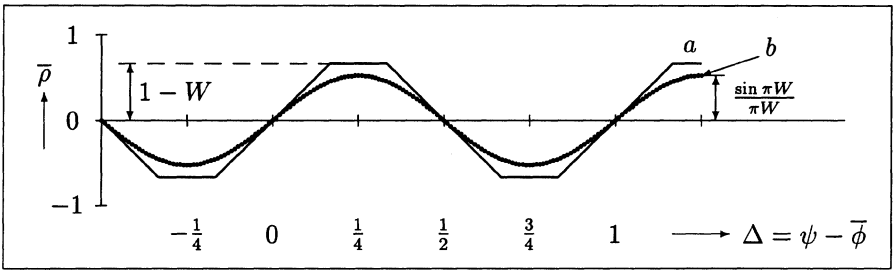


Figure 10B.1: Effective transfer characteristic $\overline{\rho}(\Delta)$ for a uniform probability distribution $p(\phi)$ of width W. a) triangular characteristic $\rho(\Delta)$; b) sinusoidal characteristic $\rho(\Delta)$.

For the triangular characteristic ρ, $\overline{\rho}$ is a clipped version of ρ. The clipping amplitude amounts to $1 - W$ (relative to an amplitude of 1 for ρ). The zero crossings of $\overline{\rho}(\Delta)$ coincide with those of $\rho(\Delta)$. The same applies to the slopes of both functions at the zero crossings. For the sinusoidal characteristic, $\overline{\rho}$ is also sinusoidal. Both the amplitude and the slope at the zero crossings are scaled by a factor $[\sin \pi W]/[\pi W]$ with respect to ρ. In the limiting case $W = 1$, zero crossings are uniformly distributed across a complete symbol interval T. Here $\overline{\rho}$ vanishes in both the sinusoidal and the triangular case. This reflects the fact that zero crossings no longer convey any timing information.

In the above example, the zero crossings of $\overline{\rho}(\Delta)$ coincide with those of $\rho(\Delta)$. Problem 10.19 shows that this is true subject to an elementary symmetry condition on $p(\phi)$ that will usually be met. For this reason the timing recovery subsystem is bound to

settle down at the appropriate phase even when $p(\phi)$ is not known explicitly. Problem 10.20 is concerned with the effective transfer characteristic for the situation that $p(\phi)$ is Gaussian. This subject is treated in considerable detail in Section 3.2.2 of [42] (see also [53]).

APPENDIX 10C

Analysis of square-law timing recovery

10C.1 Timing tone for square-law timing recovery

In this section we analyze the cyclostationary average of the output of a squarer when operating on a data signal $x(t)$ of the form (10.25). Expansion of $\overline{x^2}(t)$ yields

$$\overline{x^2}(t) = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} E[a_i a_j]q(t-iT)q(t-jT) = \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} \alpha_{i-j}q(t-iT)q(t-jT), \tag{10C.1}$$

where $\alpha_k \triangleq E[a_l a_{k+l}]$ denotes the autocorrelation function of a_k. Upon defining $p_k^t \triangleq q(t+kT)$ we get the more compact expression $\overline{x^2}(t) = (p^t * \alpha * p_-^t)_0$ where '$*$' denotes discrete-time convolution and the subscript '$_-$' denotes time-reversal. Spectrally, these operations boil down to multiplication and complex conjugation, respectively. Thus $\overline{x^2}(t)$ may be viewed as the zeroth term of a sequence $g_k \triangleq (p^t * \alpha * p_-^t)_k$ with discrete-time Fourier transform $G(e^{j2\pi\Omega}) = \mathcal{A}(e^{j2\pi\Omega})|P^t(e^{j2\pi\Omega})|^2$, where $P^t(e^{j2\pi\Omega})$ denotes the discrete-time Fourier transform of p_k^t. Correspondingly

$$\overline{x^2}(t) = g_0 = \int_{-0.5}^{0.5} G(e^{j2\pi\Omega})d\Omega = \int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega})|P^t(e^{j2\pi\Omega})|^2 d\Omega. \tag{10C.2}$$

Now p_k^t is a sampled version of $q(t)$ with sampling phase t/T. Thus $P^t(e^{j2\pi\Omega})$ is a folded version of $Q(\Omega)$ according to

$$P^t(e^{j2\pi\Omega}) = \frac{1}{T}\sum_{n=-\infty}^{\infty} Q(\Omega+n)e^{j2\pi(\Omega+n)t/T}. \tag{10C.3}$$

It follows that

$$\overline{x^2}(t) = \frac{1}{T^2}\int_{-0.5}^{0.5} \mathcal{A}(e^{j2\pi\Omega}) \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} Q(\Omega+n)Q^*(\Omega+m)e^{j2\pi(n-m)t/T}d\Omega. \tag{10C.4}$$

After a change of variables and exploitation of the periodicity of $\mathcal{A}$ this becomes

$$\begin{aligned}\overline{x^2}(t) &= \sum_{n=-\infty}^{\infty}[\frac{1}{T^2}\int_{-0.5}^{0.5}\sum_{k=-\infty}^{\infty} \mathcal{A}(e^{j2\pi(\Omega+k)})Q(\Omega+k)Q^*(\Omega+k-n)d\Omega]e^{j2\pi nt/T} \\ &= \sum_{n=-\infty}^{\infty}[\frac{1}{T^2}\int_{-\infty}^{\infty} \mathcal{A}(e^{j2\pi\Omega})Q(\Omega)Q^*(\Omega-n)d\Omega]e^{j2\pi nt/T}. \end{aligned} \tag{10C.5}$$

This validates (10.26).

10C.2 Pattern-dependent jitter for scheme with differentiator

Consider the squaring loop of Fig. 10.27 for the situation that the prefilter has output $x(t) = \sum_{k=-\infty}^{\infty} a_k q(t - kT)$, where a_k is the data sequence and $q(t)$ is the system response up until the prefilter. The signal $x(t)$ is squared, differentiated and sampled in order to obtain a TED output $\chi_k = \rho(\Delta) + z_k$, where $\rho(\Delta) = E[\chi_k]$ denotes the desired TED component and $z_k = \chi_k - \rho(\Delta)$ is a random data-dependent disturbance component that induces pattern-dependent jitter. We will show that the PSD of z_k at DC is zero when the system response $q(t)$ is even ($q(-t) = q(t)$) and sampling occurs at the ideal phase ($\Delta = 0$). To this end we first note that because of the symmetry of $q(t)$ about $t = 0$, the ideal VCO sampling phase ψ is zero (compare Fig. 10.28, where $q(t)$ is a symmetric Nyquist-1 function). The sampler has input $s(t) = Tdx^2(t)/dt = 2Tx(t)x'(t)$ where $x'(t)$ denotes the derivative of $x(t)$ with respect to t. At the ideal sampling phase ($\psi = 0$), the sampler output may be written $\chi_k = z_k = 2Tx(kT)Tx'(kT) = 2T(a*q)_k(a*c)_k$ where q_k and c_k denote the sampled versions of the system response and its derivative. This sequence has DC-component

$$\sum_{k=-\infty}^{\infty} z_k = 2T \sum_{k=-\infty}^{\infty} (a*q)_k(a*c)_k = 2T(a*a_-*q*c_-)_0. \tag{10C.6}$$

Since $q(t)$ is symmetric about $t = 0$, q_k and c_k must be even and odd functions of k, respectively. A direct consequence is that $(q*c_-)_k$ is also odd. Since $(a*a_-)_k$ is even, $(a*a_-*q*c_-)_k$ must be odd, i.e. z_k is DC-free irrespective of a_k. This validates the assertion.

Exercise 10.13 Show that the theorem continues to apply for an antisymmetric system response $q(t)$, for which $q(-t) = -q(t)$ for all t. What happens if the channel introduces an additional delay of ϕT seconds?

10C.3 Conditions for absence of pattern-dependent jitter in squaring loop with pre- and postfilters

Let $q(t)$ be the overall impulse response of the system up until the prefilter in Fig. 10.30, and let $w(t)$ be the impulse response of the postfilter. Denote the corresponding Fourier transforms by $Q(\Omega)$ and $W(\Omega)$, respectively. Let $Q(\Omega)$ be a bandpass characteristic centered around the Nyquist frequency $\Omega = 0.5$ of the form

$$Q(\Omega) = \frac{1}{2}[P(\Omega + 0.5) + P(\Omega - 0.5)], \tag{10C.7}$$

where $P(\Omega)$ is a real-valued and symmetric low-pass function of bandwidth $\Omega_0 < 0.25$ (i.e. $P(\Omega) = 0$ for $|\Omega| > \Omega_0$). Furthermore, let $W(\Omega)$ be a real-valued and symmetric

band-pass characteristic centered around $\Omega = 1$ that vanishes for $|\Omega - 1| > \Omega_1$. We will show that pattern-dependent jitter vanishes at the optimum sampling instants when $\Omega_1 < 1 - 2\Omega_0$. To this end we first take the inverse Fourier transform of (10C.7). It follows that $q(t) = p(t)\cos \pi t/T$ where $p(t)$ denotes the inverse Fourier transform of $P(\Omega)$. Because of the conditions imposed on $P(\Omega)$, $p(t)$ is real-valued and symmetric, i.e. $p(-t) = p(t)$ for all t. In terms of $p(t)$, the input signal $x(t)$ of the squarer in the absence of noise is

$$x(t) = \sum_{k=-\infty}^{\infty} a_k p(t-kT)\cos \pi(t/T-k) = \sum_{k=-\infty}^{\infty} a_k(-1)^k p(t-kT)\cos\frac{\pi t}{T}. \quad (10C.8)$$

The squarer output amounts to

$$x^2(t) = \frac{1}{2}(1+\cos\frac{2\pi t}{T})[\sum_{k=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} a_k a_n(-1)^{k+n}p(t-kT)p(t-nT)]. \quad (10C.9)$$

Now $P(\Omega)$ has no spectral content for $|\Omega| > \Omega_0$. Thus the term in square brackets is a baseband data signal with no spectral content at or above $|\Omega| = 2\Omega_0$. Because of the multiplication by $1+\cos 2\pi t/T$, $x^2(t)$ has a baseband component that vanishes for $|\Omega| > 2\Omega_0$, along with a passband component centered at the bit frequency, whose sidebands reach from $1-2\Omega_0$ to $1+2\Omega_0$. Since $\Omega_0 < 0.25$, these two components cannot overlap. The first component will be rejected by the postfilter when $1-\Omega_1 > 2\Omega_0$, as posited. The second one is passed without phase distortion because of the conditions imposed on $W(\Omega)$. This validates the assertion.

Exercise 10.14 Is it allowable or possible for $P(\Omega)$ to be symmetric but not real-valued? Is it allowable or possible for $p(t)$ to be real-valued but not symmetric?

APPENDIX 10D

Analysis of minimum mean-square error timing recovery

10D.1 Performance of the MMSE scheme of Fig. 10.39

For simplicity we restrict attention to an oversampling factor $L = 1$. Then $\chi_k = (\dot{y} * b)_k(e * b)_k = (\dot{y} * w)_k((\tilde{d} - (a * g)) * w)_k$. The timing function $\rho(\Delta)$ is the expected value of χ_k and is given by

$$\begin{aligned}\rho(\Delta) &= E[(\dot{y} * w)_k((\tilde{d} - (a * g)) * w)_k] \\ &= E[(a * p^{\Delta} * w)_k(a * (q^{\Delta} - g) * w)_k] \\ &= (p^{\Delta} * w * R^a * w_- * (q^{\Delta} - g)_-)_0 \end{aligned} \quad (10D.1)$$

where $p_k^\Delta \triangleq Tq'((k+\Delta)T)$. In the second step we have used the fact that the cross-correlation function of the noise components of $\dot{y}_k$ and $\tilde{d}_k$ is antisymmetric about the origin. The TED has gain $K_d \triangleq \partial\rho(\Delta)/\partial\Delta|_{\Delta=0} = (p * w * R^a * w_- * p_-)_0$ where $p_k \triangleq Tq'(kT) = p_k^0$. In frequency-domain notation this expression reads

$$K_d = \int_{-0.5}^{0.5} \mathcal{A}|P|^2|W|^2 d\Omega \tag{10D.2}$$

where we have suppressed the argument $e^{j2\pi\Omega}$ for notational convenience. At the ideal sampling phase ($\Delta = 0$), the error e_k consists purely of noise v_k, and χ_k consists purely of noise $u_k = (\dot{y} * w)_k(v * w)_k$. For simplicity we assume that the SNR is high enough for the noise component of $\dot{y}_k$ to be negligible compared to the data component; the forthcoming results become an upper bound when this assumption is not met. Subject to this assumption we have $u_k = (a * p * w)_k(v * w)_k$. This is a product of two terms without cross-correlation. Thus $R_k^u(k) = (p * w * R^a * w_- * p_-)_k(w * R^v * w_-)_k$, and

$$\mathcal{U}(1) = \sum_{k=-\infty}^{\infty} R^u(k) = \int_{-0.5}^{0.5} \mathcal{A}|P|^2|W|^4\mathcal{V} d\Omega \tag{10D.3}$$

where the second step is due to Parseval's equality. It follows that the TED has efficiency

$$\gamma_{\text{MMSE}} = \frac{1}{\text{SNR}}\frac{K_d^2}{\mathcal{U}(1)} = \frac{1}{\text{SNR}}\frac{[\int_{-0.5}^{0.5}\mathcal{A}|P|^2|W|^2 d\Omega]^2}{\int_{-0.5}^{0.5}\mathcal{A}|P|^2|W|^4\mathcal{V} d\Omega}. \tag{10D.4}$$

The optimum transfer function may be identified by minimizing $\mathcal{U}(1)$ subject to a constraint on K_d. This is possible by adding the constraint via a Lagrange multiplier. Along the usual lines one finds that $|W(e^{j2\pi\Omega})|^2$ should be proportional to $1/\mathcal{V}(e^{j2\pi\Omega})$. For this optimum choice, the efficiency of (10D.4) coincides with the baud-rate limit $\hat{\gamma}_{\text{BR}}$ of (10.33).

Exercise 10.15 Is it possible to replace the two filters with impulse response w_k by a single filter in one of both branches without affecting efficiency? If so, then how must the impulse response and transfer function of this single filter be chosen? Hint: Retrace the steps of Section 10.7.3.

10D.2 Performance of the approximate MMSE scheme of Fig. 10.42

We first determine K_d. For simplicity we neglect the noise component v_k of $\tilde{d}_k$. The effect of this component will be minor at practical SNRs. Then

$$\begin{aligned} E[\chi_k] &= E[(\tilde{d} * x)_k((\tilde{d} - (a * g)) * c)_k] \simeq E[(a * q^\Delta * x)_k(a * (q^\Delta - g))_k] \\ &= (q^\Delta * x * R^a * (q^\Delta - g)_-)_0. \end{aligned} \tag{10D.5}$$

Hence

$$K_d \triangleq \frac{\partial E[\chi_k]}{\partial \Delta}|_{\Delta=0} \simeq (g * x * R^a * p_-)_0 = \int_{-0.5}^{0.5} \mathcal{A}GXP^* d\Omega \tag{10D.6}$$

where we have again suppressed the argument $e^{j2\pi\Omega}$ for notational convenience. At the ideal sampling phase ($\Delta = 0$), the error e_k consists purely of noise v_k, and the TED output χ_k consists purely of noise $u_k = (\tilde{d} * x)_k v_k \simeq (a * q^0 * x)_k v_k = (a * g * x)_k v_k$. This is a product of two terms without cross-correlation. Hence $R^u(k) \simeq (g * d * R^a * d_- * g_-)_k R^v(k)$, and

$$\mathcal{U}(1) = \sum_{k=-\infty}^{\infty} R^u(k) \simeq \int_{-0.5}^{0.5} \mathcal{A}|G|^2|X|^2\mathcal{V} d\Omega. \tag{10D.7}$$

It follows that

$$\gamma \simeq \frac{1}{\text{SNR}} \frac{K_d^2}{\mathcal{U}(1)} = \frac{1}{\text{SNR}} \frac{[\int_{-0.5}^{0.5} \mathcal{A}GXP^* d\Omega]^2}{\int_{-0.5}^{0.5} \mathcal{A}|G|^2|X|^2\mathcal{V} d\Omega}. \tag{10D.8}$$

The optimum transfer function may be identified along the lines of Appendix 10E.1 and is found to be $\hat{X} = \mu P/[G\mathcal{V}]$ where μ is an arbitrary constant. For this choice we find that $\gamma \simeq \hat{\gamma}_{\text{BR}}$, with $\hat{\gamma}_{\text{BR}}$ given by (10.33). The approximation becomes tight at high SNRs. The optimum filter becomes independent of the system response when the channel has no excess bandwidth, and becomes a discrete-time differentiator (as characterized by (10.34)) when in addition noise is white.

APPENDIX 10E

Analysis of zero-forcing timing recovery

10E.1 Transfer function, efficiency

For simplicity we take $C(\mathcal{D}) = 1$, i.e. we assume that all TED filtering is concentrated in the error path. With the equivalence property (10.37) one readily extends the results to arbitrary functions $C(\mathcal{D})$. For $C(\mathcal{D}) = 1$ the reference sequence z_n and TED output χ_n are nonzero for $n = kL$ only. As a result, the TED transfer function becomes

$$\begin{aligned} \rho(\Delta) &= \frac{1}{L}E[\chi_{kL}] = \frac{1}{L}E[y_{kL}a_k] \\ &= \frac{1}{L}E[(a * (q^\Delta - g) * w)_{kL}a_k] = \frac{1}{L}\sum_{i=-\infty}^{\infty} R_i^a((q^\Delta - g) * w)_{iL} \end{aligned} \tag{10E.1}$$

The TED gain K_d is the slope of $\rho(\Delta)$ in the origin ($\Delta = 0$), i.e.

$$K_d \triangleq \frac{\partial \rho(\Delta)}{\partial \Delta}|_{\Delta=0} = \frac{1}{L}\sum_{i=-\infty}^{\infty} R_i^a(p * w)_{iL} \tag{10E.2}$$

where p_n was defined in (10.35). The noise component $u_n = z_n(v * w)_n$ at the TED output is nonzero for $n = kL$ only, and its time-averaged autocorrelation function may be written

$$\begin{aligned} R_k^u &\triangleq \frac{1}{L}E[u_{iL}u_{(k+i)L}] = \frac{1}{L}E[z_{iL}z_{(k+i)L}]E[(v*w)_{iL}(v*w)_{(k+i)L}] \\ &= \frac{1}{L}R_k^a(R^v * w * w_-)_{kL} \end{aligned} \tag{10E.3}$$

where R_n^v denotes the autocorrelation function of v_n. The loop reacts mainly to the DC-component of the TED output. For this reason, noise-induced jitter is mainly determined by the power spectral density $\mathcal{U}(e^{j2\pi\Omega/L})$ of u_n at DC ($\Omega = 0$). This density amounts to

$$\mathcal{U}(1) = \sum_{k=-\infty}^{\infty} R_k^u = \frac{1}{L}\sum_{k=-\infty}^{\infty} R_k^a(R^v * w * w_-)_{kL}. \tag{10E.4}$$

Efficiency is, therefore, given by

$$\gamma_{\text{ZF}} = \frac{1}{L}\frac{[\sum_{i=-\infty}^{\infty} R_i^a(p*w)_{iL}]^2}{\sum_{k=-\infty}^{\infty} R_k^a(R^v * w * w_-)_{kL}}. \tag{10E.5}$$

In particular for baud-rate sampling ($L = 1$), this may be recast in frequency-domain form as

$$\gamma_{\text{ZF}} = \frac{1}{\text{SNR}}\frac{[\int_{-0.5}^{0.5}\mathcal{A}(e^{j2\pi\Omega})P(e^{j2\pi\Omega})W(e^{j2\pi\Omega})d\Omega]^2}{\int_{-0.5}^{0.5}\mathcal{A}(e^{j2\pi\Omega})\mathcal{V}(e^{j2\pi\Omega})|W(e^{j2\pi\Omega})|^2 d\Omega}. \tag{10E.6}$$

Efficiency may be optimized with respect to w_n by minimizing $\mathcal{U}(1)$ subject to a constraint on K_d. The constraint may be added with a Lagrange multiplier and the optimizing transfer function is found to be $\hat{W}(e^{j2\pi\Omega/L}) = P^*(e^{j2\pi\Omega/L})/\mathcal{V}(e^{j2\pi\Omega/L})$ irrespective of L. For baud-rate sampling ($L = 1$) the optimum efficiency $\hat{\gamma}_{\text{ZF}}$ coincides with the baud-rate efficiency limit $\hat{\gamma}_{\text{BR}}$ of (10.33). ML efficiency accrues when L is high enough to avoid aliasing.

10E.2 Constrained optimization of c_n

Denote by S the set of indices n for which c_n is allowed to be nonzero. We seek to find the coefficient vector $\underline{c}$ with components c_n, $n \in S$, that minimizes $\mathcal{U}(1)$ subject to the constraints $K_d = 1$ and $\sum_{n\in S} c_n = 0$. To this end we define the Lagrangian $\Gamma \triangleq \mathcal{U}(1) - 2\mu_1 K_d - 2\mu_2 \sum_{n\in S} c_n$ where μ_1 and μ_2 are Lagrange multipliers. We first express Γ in matrix notation. Define a symmetric matrix $\mathbf{M}$ with components

$$\mathbf{M}_{i,j} \triangleq \frac{1}{L}\sum_{k=-\infty}^{\infty} R_k^a(R^v * w * w_-)_{kL+i-j} \text{ for } i,j \in S, \tag{10E.7}$$

a vector $\underline{v}$ with components

$$\underline{v}_i \triangleq \frac{1}{L} \sum_{k=-\infty}^{\infty} R_k^a (p * w)_{kL+i} \text{ for } i \in S, \tag{10E.8}$$

and the all-one vector $\underline{1}$ with components 1 for all $i \in S$. With Appendix 10E one establishes that $\Gamma = \underline{c}^T \mathbf{M} \underline{c} - 2\mu_1 \underline{c}^T \underline{v} - 2\mu_2 \underline{c}^T \underline{1}$ where '^{T}' denotes transposition. A standard derivation reveals that the optimum coefficient vector $\hat{\underline{c}}$ is given by $\hat{\underline{c}} = \hat{\mu_1} \mathbf{M}^{-1} \underline{v} + \hat{\mu_2} \mathbf{M}^{-1} \underline{1}$ where $\mathbf{M}^{-1}$ is the inverse of $\mathbf{M}$ and $\hat{\mu_1}$ and $\hat{\mu_2}$ are the solution of the matrix equation

$$\begin{pmatrix} \underline{v}^T \mathbf{M}^{-1} \underline{v} & \underline{v}^T \mathbf{M}^{-1} \underline{1} \\ \underline{1}^T \mathbf{M}^{-1} \underline{v} & \underline{1}^T \mathbf{M}^{-1} \underline{1} \end{pmatrix} \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{10E.9}$$

10E.3 Pattern-dependent jitter

Let a_k be an arbitrary data sequence of finite length and sampling rate $1/T$. Let $\xi_n \triangleq (a * g * w)_n (a * c)_n$ where g_n, w_n and c_n are impulse responses of sampling rate L/T. Then χ_n has DC-component

$$\begin{aligned} \sum_{n=-\infty}^{\infty} \xi_n &= \sum_{n=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} a_i a_j (g * w)_{n-iL} c_{n-jL} \\ &= \sum_{k=-\infty}^{\infty} (a * a_-)_k (g * w * c_-)_{kL} = (a * a_- * \eta)_0 \end{aligned} \tag{10E.10}$$

where $\eta_k \triangleq (g_- * w_- * c)_{kL}$. Now $(a * a_-)_k$ is an even function of k, i.e. $(a * a_-)_{-k} = (a * a_-)_k$ for all k. In order for $(a * a_- * \eta)_k$ to vanish at $k = 0$ irrespective of a_k we must require that η_k has no even component. Except for the trivial solution ($\eta_k = 0$ for all k) this leaves only odd functions η_k. □

PROBLEMS

10.1 Assume that the data sequence a_k in Fig. 10.3 has a finite length N. Show that the summation in (10.5) can be restricted to N terms.

10.2 Prove (10.10) and (10.11).

10.3 Consider a channel with white additive Gaussian noise of power spectral density N_0 whose symbol response $h(t)$ is a rectangular NRZ pulse of width T according to $h(t) = 1$ for $|t| < T/2$ and 0 else. The input of this channel is uncorrelated binary data $a_k \in \{-1, 1\}$, and the channel output $r(t)$ is filtered by a filter with rectangular impulse response $w(t) = 1/(\alpha T)$ for $|t| < \alpha T/2$, where $\alpha \in [0, 1]$ is a parameter that controls the bandwidth of the filter.

1. Sketch the system response $(h * w)(t)$ for $\alpha \to 0$, $\alpha = 0.5$ and $\alpha = 1$.
2. Sketch the eye pattern at the output of the filter for these three cases. Is there any intersymbol interference? In which parts of the eye patterns is timing information present?
3. The output of the filter is applied to a sampler that closes at the optimum sampling instants, which is followed by a bit-by-bit detector. For which value of α is the predetection signal-to-noise ratio largest? Is a further improvement possible by resorting to another filter or a different detection scheme?

These results illustrate two points: 1) As system bandwidths increase, timing information typically gets concentrated away from the optimum sampling instants for data detection; 2) the output of a matched filter concentrates an optimum amount of timing information in the vicinity of these optimum sampling instants.

10.4 Consider the system of Example 10.1. Use (10.11) to show that the TED transfer characteristic for the $4T$ pattern is purely sinusoidal for $\beta = 0$ and is a sum of two sinusoids for $\beta = 1$.

10.5 Consider two channels with transfer function

$$H_1(\Omega) = \begin{cases} T, & |\Omega| < 0.5 \text{ and} \\ 0, & |\Omega| > 0.5 \end{cases} \tag{10P.1}$$

and

$$H_2(\Omega) = \begin{cases} T(1 - |\Omega|), & |\Omega| < 1 \text{ and} \\ 0, & |\Omega| > 1, \end{cases} \tag{10P.2}$$

respectively. Assume that both channels have white noise $n(t)$ with power spectral density N_0.

1. Does either of the channels exhibit intersymbol interference?
2. Determine the signal-to-noise ratio SNR for both channels.
3. Determine the efficiencies γ_{ML} according to (10.15) for both channels. Which channel has largest efficiency? How large is the difference in dB? Is this consistent with Example 10.2?

10.6 Consider the Lorentzian system of Example 10.4 when used in conjunction with the Bi-Phase modulation code (see Appendix 3A). Use a development akin to that of Appendix 10A.2 along with expression (3A.10) to show that the merit factor $F \triangleq \int_{-\infty}^{\infty} (2\pi\Omega)^2 \text{SN}(\Omega) d\Omega$ amounts to

$$F = \frac{3}{2\pi N_0} \frac{2 + 30D^2 + 300D^4 + 680D^6 + 480D^8}{D^3(1 + D^2)^3(1 + 4D^2)^3}. \tag{10P.3}$$

What is the corresponding efficiency factor γ_{ML}? Sketch γ_{ML} as a function of the normalized information density D and comment on the difference with NRZ (see Fig. 10.5.a). Show that

$$\frac{F(\text{Bi-phase})}{F(\text{NRZ})} = \frac{3 + 45D^2 + 450D^4 + 1020D^6 + 720D^8}{(1 + 4D^2)^3(1 + 3D^2 + 6D^4)}. \tag{10P.4}$$

What is the significance of this ratio? (See Example 4.14 for an explanation and a graph.)

10.7 Consider a Lorentzian recording channel as in Example 10.4 with a normalized information density $D = 2$ and signal-to-noise ratio SNR of 10 dB. Assume that data $a_k \in \{-1, 1\}$ is uncorrelated and that timing recovery is done with a first-order phase-locked loop as in Example 9.8, based on a data-aided maximum-likelihood timing-error detector. Assume further that the loop is dimensioned for a time constant $1/K_t$ of 100 symbol intervals.

1. What conditions must be met for jitter Δ_k to be purely induced by noise when the loop is in the tracking mode of operation?
2. Determine the RMS jitter σ_Δ in this case. Hint: Use the results of Appendix 10A.2 and Example 9.8.
3. Variations of tape-to-head contact may, in effect, cause D to increase temporarily. At the same time SNR decreases in proportion to $1/D$ or even $1/D^2$. Let us assume that D can become as large as 3. What is the minimum RMS jitter σ_Δ that is to be expected in this worst-case condition? How does this compare to the result of the previous question?

10.8 This problem is concerned with pattern-dependent jitter in the data-aided maximum-likelihood timing-error detector of Fig. 10.3. This jitter is induced by the data component z_k of the TED output χ_k. We have shown in Appendix 10E.3 that z_k has no spectral content at DC. Since only frequencies near DC are passed by the loop, pattern-dependent jitter is bound to be insignificant. We now examine how system parameter variations affect this conclusion.

1. Assume that variations of channel and/or matched filter cause the system response $s(t)$ up until the differentiator to change. Does this always lead to significant pattern-dependent jitter? If not, then what condition must $s(t)$ meet for pattern-dependent jitter to remain negligible? Hint: Use the results of Appendix 10E.3.
2. Assume that only the amplitude characteristics of the system exhibit variations. Does this affect pattern-dependent jitter significantly? How about variations of the phase characteristics? Do such variations increase pattern-dependent jitter? What is the exception to this rule?

10.9 Assume that the -1 and $+1$ levels in Figs. 10.8 and 10.9 are represented by voltages of 0V and +5V, respectively. What is the DC-level of $\chi(t)$ at the ideal clock phase? Is this a serious problem?

10.10 What happens to the characteristic of Fig. 10.10 if signal levels are as in Problem 10.9? How does this affect loop properties? How can this problem be repaired?

10.11 Consider a system response $q(t)$ that satisfies the condition

$$q(t) = \begin{cases} 1 & \text{for } t = 0, \\ \alpha & \text{for } t = \pm T/2, \text{ and} \\ 0 & \text{for } t = nT/2, |n| \geq 2. \end{cases} \tag{10P.5}$$

1. Is $q(t)$ a Nyquist-1 pulse? At what instants must the prefilter output signal be sampled to obtain a two-level eye?
2. At what instants must the prefilter output signal be sampled to obtain a three-level eye?
3. Show that the prefilter output signal has zero-crossings exactly halfway between the ideal decision instants.
4. Verify that $q(t)$ has Fourier transform

$$Q(\Omega) \triangleq \begin{cases} 2T(1 + 2\alpha \cos \pi\Omega) & \text{for } |\Omega| \leq 1, \text{ and} \\ 0 & \text{for } |\Omega| > 1. \end{cases} \tag{10P.6}$$

10.12 Prove that there are no system characteristics $Q(\Omega)$ with less than 100% excess bandwidth that satisfy (10P.5). Hint: Translate (10P.5) to the frequency domain, and show that the resulting condition is met with the smallest possible bandwidth by the characteristic $Q(\Omega)$ of (10P.6).

10.13 Consider the zero-crossing timing-error detector of Fig. 10.7 for the situation that data a_k is uncorrelated.

1. Determine the autocorrelation function $R_b(k)$ of the indicator signal b_k of Section 10.3.5.
2. Refer to Fig. 10.15. Denote the autocorrelation function of v_k and u_k by $R_v(k)$ and $R_u(k)$, respectively. Use the fact that b_k and v_k are statistically independent to express the autocorrelation function $R_u(k)$ in terms of $R_v(k)$.
3. Find the power spectral density $\mathcal{U}(e^{j2\pi\Omega})$ of u_k.
4. Assume that noise v_k is white. What can you say about $\mathcal{U}(e^{j2\pi\Omega})$?
5. For what type of noise PSD does $\mathcal{U}(e^{j2\pi\Omega})$ have a depression near DC? Is a PSD of this kind likely to occur in practice? What is the consequence for noise-induced jitter?
6. Show that the efficiency γ of the TED may be expressed as

$$\gamma_{\text{ZC}} = \frac{1}{\text{SNR}} \frac{[\int_{-\infty}^{\infty} 2\pi\Omega \sin \pi\Omega Q(\Omega) d\Omega]^2}{2\int_{-0.5}^{0.5} \mathcal{V}(e^{j2\pi\Omega}) \sin^2 \pi\Omega d\Omega}. \tag{10P.7}$$

10.14 Consider the binary delay-tracking loop of Fig. 10.17 for the situation that the sampler is replaced by a track-and-hold circuit so that a continuous-time TED output $\chi(t)$ emerges. Assume that data a_k is such that successive transitions of $\hat{y}(t)$ can be spaced far apart. How will this affect loop performance? Does this problem also arise with a discrete-time TED output as in Fig. 10.17?

10.15 At what clock phase will the loop of Fig. 10.17 settle down? How can the decision instants be derived? Verify that this is done properly in Figs. 10.17 and 10.18.

10.16 Verify that the PFD of Fig. 10.22 operates as described in Section 10.3.8. Would it continue to work properly if the minimum run-length of the data were not constrained?

10.17 Verify that the PFD of Fig. 10.22 has the transfer characteristic of Fig. 9.18.

10.18 Assume that the function $g(t)$ of Appendix 10B is an NRZ pulse of width $T/2$ and that the clock is a square wave of frequency $1/T$ with 50% duty cycle and no DC. Verify that $\rho(\Delta)$ is triangular as in Fig. 10.10. How does $\rho(\Delta)$ change if $c(t)$ is a sinusoid rather than a square wave? How does it change when the duration of $g(t)$ deviates from $T/2$ seconds?

10.19 In this problem we are concerned with the effect of threshold-crossing jitter on the effective TED transfer characteristic for timing recovery based on zero crossings (see Appendix 10B.3). Let $\rho(\Delta) = 0$ for $\Delta = \Delta_0 + n/2$ for some Δ_0 and all $n \in \mathbb{Z}$, and let $\rho(\Delta)$ be antisymmetric about these zero-crossings. Furthermore, let $p(\phi)$ be symmetric about $\overline{\phi}$. Use symmetry arguments to show that $\overline{\rho} = 0$ for $\Delta = \Delta_0 + n/2$.

10.20 Let ϕ be a Gaussian random variable with mean $\overline{\phi}$ and variance σ_ϕ^2.

1. Determine the effective TED transfer characteristic $\overline{\rho}(\Delta)$ as defined in Appendix 10B.3 for a sinusoidal characteristic $\rho(\Delta)$ of amplitude 1. Show that $\overline{\rho}$ has

amplitude $A = e^{-2(\pi\sigma_\phi)^2}$.
Hint: Use the result of Problem 10.19 and the fact that a filtered sinusoid is a sinusoid of the same frequency.

2. How large is A when the RMS zero-crossing jitter amounts to 5% of a symbol interval? Does this cause a significant degradation of tracking properties?
3. Repeat the previous question for 10% jitter. Is the conclusion any different?

10.21 Bi-phase is a DC-free binary modulation that we have encountered previously in Appendix 3A as well as in Section 4.9 and Example 10.18. It has rate $R = 1/2$ and produces a binary signal $s(t) = \sum_{k=-\infty}^{\infty} a_k c(t - kT)$ where

$$c(t) = \begin{cases} 1, & 0 < t < \frac{T}{2}, \\ -1, & \frac{T}{2} < t < T, \text{ and} \\ 0, & \text{else.} \end{cases} \tag{10P.8}$$

In this problem we are concerned with timing recovery based on zero crossings for Bi-phase.

1. At which instants t is $s(t)$ certain to exhibit zero crossings? (These zero crossings are sometimes called *mid-bit transitions*.)
2. At which other instants t can $s(t)$ exhibit zero crossings?
3. Amend the scheme of Fig. 10.17 in such a way that it becomes suited for Bi-phase. How high must the clock frequency be?
4. How can the decision instants be derived from the clock? What ambiguity can arise? What property of $s(t)$ can be used to resolve this ambiguity?
5. How many transitions are there, upon average, per symbol interval T? How does this compare to NRZ? Comment on the relative performance of zero-crossing-based timing-recovery schemes for Bi-phase and NRZ.
6. Outline how decision instants may be derived from the mid-bit transitions in an open-loop fashion, i.e. without using a PLL. An early example of this approach may be found in [23].

10.22 This problem is concerned with lock-detection for the phase-and-frequency detector (PFD) of Fig. 10.22 (Section 10.3.7). Fig. 10P.1 depicts a circuit that was first described in [16].

The circuitry that is added to the PFD permits the detection of cycle slips. These are said to occur when the clock drops (cs^-) or adds (cs^+) one cycle of oscillation relative to the incoming signal. The loop is defined to be in lock if no cycle slips occur. (This is not the only possible definition. Alternative definitions may be found, for example, in [24].)

1. What condition must be met for the signal cs^- to be active? Use example waveforms as in Fig. 10.23 to illustrate this condition.
2. Repeat the previous question for a positive cycle slip (cs^+).
3. Verify that the 'lock detect' signal is active if and only if no cycle slips occur.

10.23 Consider the squaring loop of Section 10.4 (Fig. 10.27). Equation (10.26) describes the Fourier coefficients c_n of the tone $\overline{x^2}(t)$ at the output of the squarer. Show that $c_{-n} = c_n^*$ for all n when $q(t)$ is real-valued, and that c_n is real-valued for all n when $q(t)$ is symmetric about $t = 0$. What happens if $q(t)$ is antisymmetric about $t = 0$?

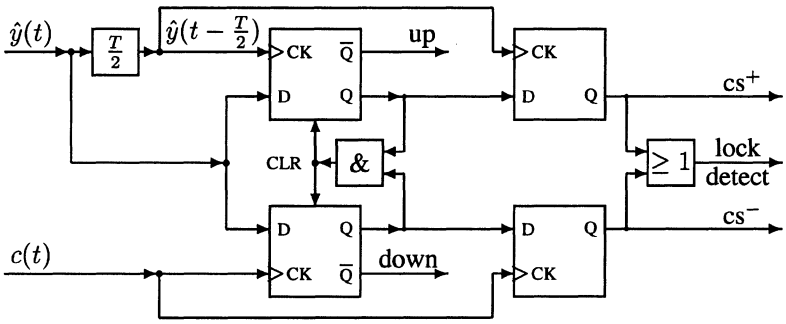


Figure 10P.1: Extension of the phase-and-frequency detector (PFD) of Fig. 10.22 for lock-detection and detection of cycle slips.

10.24 Assume that a stationary noise signal $n(t)$ is added to the signal $x(t)$ of (10.25), and that $n(t)$ is not correlated with $x(t)$. Argue that this affects only the coefficient c_0 in (10.26). Is the timing tone affected?

10.25 Consider the squaring loop of Fig. 10.27. Assume that $x(t)$ suffers an additional delay of ϕT seconds with respect to (10.25), i.e. that

$$x(t) = \sum_{k=-\infty}^{\infty} a_k q(t-(k+\phi)T). \tag{10P.9}$$

Show that $\overline{x^2}(t)$ now has Fourier coefficients c'_n that are related to the original ones according to

$$c'_n = e^{j2\pi\phi} c_n. \tag{10P.10}$$

10.26 Consider the squaring loop with pre- and post-filter of Section 10.4.4 (Fig. 10.30). Use the results of Appendix 10C.1 in order to express the amplitude A of the timing tone $\overline{s}(t)$ in terms of $Q(\Omega)$ and $W(\Omega)$. How does A depend on Ω_0 and Ω_1? Does prefiltering become more or less effective as the excess bandwidth of the channel grows?

10.27 This problem relates to the non-data-aided early-late timing-recovery scheme of Section 10.5 (Fig. 10.38) with squarers rather than full-wave rectifiers. Let the system response $q(t)$ up until the matched filter (MF) be symmetric with non-negative Fourier integral $Q(\Omega)$. Assume for simplicity that noise is absent.

1. Show that the power $\overline{y^2}(t) \triangleq E[y^2(t)]$ of the MF output is an even and periodic function of t with a global maximum for $t = kT$.
   Hint: expand the power $\overline{y^2}(t)$ of $y(t)$ into a Fourier series as in Section 10.4.2 and Appendix 10C.1, and show that all Fourier coefficients c_k are real-valued and non-negative.
2. Show that the loop will settle down at the ideal sampling phase for data detection.
3. What happens if the channel has no excess bandwidth? Would it help if the squarers were replaced by full-wave rectifiers or other symmetric nonlinearities?

10.28 Consider the non-data-aided early-late timing-recovery scheme of Fig. 10.38, with squarers instead of full-wave rectifiers. Show that this scheme follows from the squaring loop of Fig. 10.27 upon replacing the differentiation by a time-difference operation.
Hint: repeat the argumentation that led to Fig. 10.37.

10.29 Consider the system of Problem 10.3. Determine the average amplitude $\overline{|y(t)|}$ and power $\overline{y^2}(t)$ of the output of the matched filter in the absence of noise. At which sampling phase are $\overline{|y(t)|}$ and $\overline{y^2}(t)$ maximal? Verify that this is the optimum sampling phase for data detection.

10.30 In this problem we are concerned with the efficiency γ_{MMSE} according to (10D.4) of the MMSE timing-recovery scheme of Fig. 10.39 with baud-rate sampling ($L = 1$) for the situation that data a_k is sinusoidal with normalized frequency Ω_0.

1. Express γ_{MMSE} in terms of SNR, $P(e^{j2\pi\Omega_0})$ and $V(e^{j2\pi\Omega_0})$. Show that γ_{MMSE} does not depend on w_k.
2. Show that γ_{MMSE} coincides with the optimum baud-rate efficiency $\hat{\gamma}_{\text{BR}}$ according to (10.33).

10.31 In this problem we are concerned with the efficiency γ of the approximate MMSE timing-recovery scheme of Fig. 10.42 for the situation that data a_k is sinusoidal with normalized frequency Ω_0. We restrict attention to real-valued system responses $q(t)$, and assume that SNR is high enough for (10D.8) to be valid.

1. Express γ in terms of SNR, $G(e^{j2\pi\Omega_0})$, $X(e^{j2\pi\Omega_0})$, $P(e^{j2\pi\Omega_0})$ and $\mathcal{V}(e^{j2\pi\Omega_0})$. Hint: use (10D.8) and the fact that $q(t)$ is real-valued.
2. Express $G(e^{j2\pi\Omega_0})$, $X(e^{j2\pi\Omega_0})$ and $P(e^{j2\pi\Omega_0})$ in polar notation. Show that the amplitude characteristics of $X(e^{j2\pi\Omega_0})$ do not affect $\hat{\gamma}_{\text{MMSE}}$. What conditions must be imposed on the phase of $X(e^{j2\pi\Omega_0})$ in order for γ to approach the optimum baud-rate efficiency $\hat{\gamma}_{\text{BR}}$?
3. Assume that the system response $q(t)$ is symmetric about $t = NT$ for some integer N. Find a simple impulse x_k that will cause γ to approach $\hat{\gamma}_{\text{BR}}$ irrespective of Ω_0. Show that this same response remains applicable when $q(t)$ is antisymmetric about $t = NT$.

10.32 Prove that the efficiency of the scheme of Fig. 10.43 approaches that of the zero-forcing scheme of Fig. 10.59 at high signal-to-noise ratios. How do the complexities of these two schemes compare?

10.33 This problem is concerned with pattern-dependent jitter in the sampled-derivative timing-recovery system of Section 10.6.4. Central to this jitter is the random variable $\xi_n = (\tilde{d} * w)_n(\dot{y} * w)_n$ where $\tilde{d}_n = y(t_n)$ and $\dot{y}_n = Ty'(t_n)$.

1. Show that the DC-component of ξ_n may be expressed as $\sum_{n=-\infty}^{\infty} \xi_n = (\tilde{d} * w * w_- * \dot{y}_-)_0$ where $_-$ denotes time-reversal. Recast this expression in frequency-domain notation.
2. Assume that $y(t)$ is bandlimited so that the sampling operation causes no aliasing. Express $\sum_{n=-\infty}^{\infty} \xi_n$ directly in terms of the Fourier integral $Y(\Omega)$ of $y(t)$. Show that ξ_n has no DC-component irrespective of w_n.

10.34 Explain the periodicity and the shape of the decision-directed TED transfer characteristic of Fig. 10.51.

10.35 Consider a full-response system with negligible excess bandwidth. Tailor the timing-recovery scheme of Fig. 10.48 to this system in such a manner that it operates at the baud rate and exploits only the first precursor g_{-1}. Is the scheme causal? How can this problem be solved? (The resulting scheme was first described in [44].) What further changes are needed to exploit the first post-cursor g_1 as well? How much will this improve efficiency if noise at the output of the prefilter is white?

10.36 Consider a full-response system with negligible excess bandwidth. Develop a causal and near-optimum zero-forcing timing-recovery scheme akin to the one of Fig. 10.54 with four nonzero coefficients c_k. How much has efficiency improved relative to the schemes of Problem 10.35 if sampled noise at the output of the prefilter is white?

10.37 Consider the timing-recovery scheme of Fig. 10.54. Assume that only delayed decisions $\hat{a}_{k-\delta}$ for some $\delta > 0$ are available as a result of implementation latency and/or detection delay. Modify the scheme accordingly.

10.38 Let x_k be an impulse response with discrete-time Fourier transform $X(e^{j2\pi\Omega}) \triangleq j2\pi\Omega$ for $\Omega \in [0, 0.5]$. This is the discrete-time analog of differentiation.

1. How must $X(e^{j2\pi\Omega})$ be extended beyond $\Omega \in [0, 0.5]$? Does $X(e^{j2\pi\Omega})$ exhibit any discontinuities?
2. Determine the coefficients x_k by means of inverse Fourier transformation. Hint: make use of the integral identity

$$\int \sin kx dx = \frac{\sin kx}{k^2} - \frac{x \cos kx}{k}. \tag{10P.11}$$

Verify that x_k satisfies (10.34).

10.39 Consider the ZF timing-recovery scheme of Fig. 10.59 when applied to the $1 - \mathcal{D}^2$ system of Examples 10.13 and 10.14. How far does the efficiency γ_{ZF} of this scheme lag behind $\hat{\gamma}_{\text{ZF}}$? Answer the same question for the situation that the scheme is applied to the $1 + \mathcal{D}$ system of Example 10.15. Hint: Determine the coefficients c_k that are in effect, and apply equation (10.42).

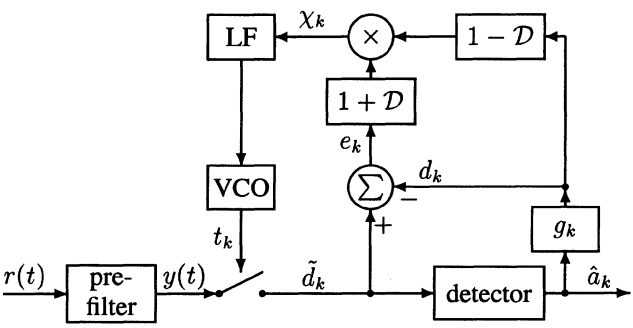


Figure 10P.2: Zero-forcing scheme equivalent to the one of Fig. 10.59 with filtering distributed across error and data paths.

10.40 Show that the ZF timing-recovery scheme of Fig. 10P.2 is equivalent to the one of Fig. 10.59 in the sense of Section 10.7.3 Discuss the relative merits of both schemes.

10.41 The schemes of Figs. 10P.2 and 10.59 are equivalent and are both based on the difference operator $C(\mathcal{D}) = \mathcal{D}^{-1} - \mathcal{D}$. A better discrete-time approximation to differentiation is obtained if we include two more terms according to $C(\mathcal{D}) = -0.5\mathcal{D}^{-2} + \mathcal{D}^{-1} - \mathcal{D} + 0.5\mathcal{D}^2$ (compare (10.34)). Derive two equivalent causal loops that are based on this approximation. Hint: make use of the factorization $-0.5 + \mathcal{D} - \mathcal{D}^3 + 0.5\mathcal{D}^4 = (-0.5 + \mathcal{D} - 0.5\mathcal{D}^2)(1 - \mathcal{D}^2)$.

10.42 Determine a causal ZF scheme equivalent to the ones of Figs. 10.59 and 10P.2 such that d_k is merely delayed before being multiplied with a filtered error signal. What is the impulse response of the error filter?

10.43 In [29], a ZF timing-recovery scheme is described that has $W(\mathcal{D}) = \mathcal{D}^{-1} - \mathcal{D}$ and $C(\mathcal{D}) = 1 - \mathcal{D}$. Find the impulse response c_k of an equivalent scheme without error filter (i.e. with $W(\mathcal{D}) = 1$). Sketch a system response g_k for which this scheme is near-optimum if noise is white.

10.44 Consider the ZF timing-recovery scheme of Fig. 10.48 for the situation that the system response $q(t) = (h * f)(t)$ is symmetric about $t = NT$. Find a simple impulse response c_k that will yield optimum efficiency $\hat{\gamma}_{\text{ZF}}$ for sinusoidal data, irrespective of the normalized frequency Ω_0. Show a complete receiver for full-response signaling that is based on this approach.

10.45 In this problem we are concerned with the efficiency of ZF timing-recovery schemes for sinusoidal data a_k with normalized frequency Ω_0. We restrict attention to system responses $q(t)$ and filter impulse responses c_k that are real-valued.

1. Express the efficiency γ_{ZF} of the ZF scheme of Fig. 10.48 in terms of SNR, $P(e^{j2\pi\Omega_0})$, $C(e^{j2\pi\Omega_0})$ and $\mathcal{V}(e^{j2\pi\Omega_0})$. Hint: make use of (10E.6) and the fact that $q(t)$ and c_k are both real-valued.
2. Express $P(e^{j2\pi\Omega_0})$ and $C(e^{j2\pi\Omega_0})$ in polar notation. Show that γ_{ZF} does not depend on $|C(e^{j2\pi\Omega_0})|$. How must the phase of $C(e^{j2\pi\Omega_0})$ be chosen in order to optimize γ_{ZF} ?

10.46 Use the result of Problem 10.45 to determine the efficiency of the poor-man's timing-recovery scheme of Fig. 10.50 when used for a Class IV system with white noise and no excess bandwidth in conjunction with the training pattern '$\ldots + + - - + + - - \ldots$' of normalized frequency $\Omega_0 = 0.25$.

10.47 In this problem we are concerned with the efficiency of ZF timing-recovery schemes when data a_k is sinusoidal and the system response $q(t)$ is real-valued and symmetric about $t = (N + \frac{1}{2})T$.

1. Verify that the Duobinary $(1 + \mathcal{D})$ system response with no excess bandwidth is of this type. How large is N?
2. Consider the sampled derivative $p_k = Tq'(kT)$. What symmetry condition does p_k meet? How does this condition translate spectrally? Show that only the phase characteristics of p_k are of concern.
3. Use the results of Problem 10.45 to find a category of simple impulse response c_k that will provide optimum efficiency $\hat{\gamma}_{\text{ZF}}$ irrespective of the data frequency.
4. Devise a simple ZF timing-recovery scheme with optimum efficiency for a system with sinusoidal data and a Duobinary system response with no excess bandwidth.

10.48 Redo Problem 10.47 for a system response $q(t)$ that is antisymmetric about $k = (N + \frac{1}{2})T$. Use a Bipolar $(1 - \mathcal{D})$ system response to illustrate the results. How large is N for this response?

10.49 Consider a Bipolar $(1 - \mathcal{D})$ system with system response $q(t) = f(t) - f(t - T)$ where $f(t)$ is a raised-cosine pulse with roll-off factor $\beta = 1$. Develop a near-optimum oversampled ZF timing-recovery scheme akin to the one of Fig. 10.56 for the situation that sampled noise at the output of the prefilter is white.

10.50 Compare the configuration of Fig. 10.41 with that of Fig. 10.56 and explain any differences. Comment on the relative performance and complexity of both schemes.

10.51 Consider the optimum ZF scheme according to (10.38). Show that $(g * w * c_-)_n$ is antisymmetric about $n = 0$ irrespective of $\mathcal{A}(e^{j2\pi\Omega})$ and $\mathcal{V}(e^{j2\pi\Omega/L})$ in each of the following cases:

1. the sampling operation in Fig. 10.48 causes no aliasing;
2. the continuous-time system response $q(t)$ is symmetric about $t = 0$ (i.e. $q(-t) = q(t)$ for all t);
3. $q(t)$ is antisymmetric about $t = 0$ (i.e. $q(-t) = -q(t)$ for all t).

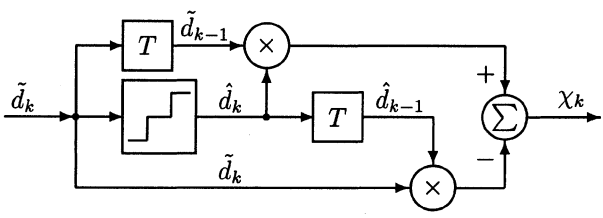


Figure 10P.3: Timing-error detector according to [14] for baud-rate timing recovery in Class IV partial-response system.

10.52 Fig. 10P.3 depicts a baud-rate timing-error detector for Class IV partial-response systems that was described in [14]. The TED contains a ternary bit-by-bit detector that serves to take instantaneous decisions $\hat{d}_k$ with respect to the Class IV partial-response signal $d_k = a_k - a_{k-2}$, where a_k is a binary data sequence.

1. Reverse the reasoning of Example 10.25 to derive the timing function that underlies this timing-error detector. In which ZF timing-recovery scheme for Class IV did we come across this same function? (In [12] it is shown that this ZF scheme has, in fact, the same efficiency as the one of Fig. 10P.3.)
2. Compare the topology of Fig. 10P.3 with that of the timing-error detector in Fig. 10.64 and explain the differences.

REFERENCES

[1] M.R. Aaron, 'PCM Transmission in the Exchange Plant', *Bell Syst. Tech. J.*, Vol. 41, pp. 99-141, Jan. 1962.

[2] W.L. Abbott and J.M. Cioffi, 'Timing Recovery for Adaptive Decision Feedback Equalization of the Magnetic Storage Channel', Proc. GLOBECOM '90, pp. 1794-1799, San Diego, CA, Dec. 2-5, 1990.

[3] T. Aboulnasr, M. Hage, B. Sayar and S. Aly, 'Characterization of a Symbol Rate Timing Recovery Technique for a 2B1Q Digital Receiver', *IEEE Trans. Commun.*, Vol. COM-42, No. 2/3/4, pp. 1409-1414, Feb./March/April 1994.

[4] O. Agazzi, C-P. J. Tzeng, D.G. Messerschmitt and D.A. Hodges, 'Timing Recovery in Digital Subscriber Loops', *IEEE Trans. Commun.*, Vol. COM-33, pp. 558-569, June 1985.

[5] N.A. D'Andrea and U. Mengali, 'A Simulation Study of Clock Recovery in QPSK and 9QPRS Systems', *IEEE Trans. Commun.*, Vol. COM-33, No. 10, pp. 1139-1142, Oct. 1985.

[6] N.A. D'Andrea and M. Luise, 'Design and Analysis of a Jitter-Free Clock Recovery Scheme for QAM Systems', *IEEE Trans. Commun.*, Vol. COM-41, No. 9, pp. 1296-1299, Sept. 1993.

[7] J. Armstrong, 'Symbol Synchronization Using Baud-Rate Sampling and Data-Sequence-Dependent Signal Processing', *IEEE Trans. Commun.*, Vol. COM-39, No. 1, pp. 127-132, Jan. 1991.

[8] G. Ascheid, M. Oerder, J. Stahl, and H. Meyr, 'An All Digital Receiver Architecture for Bandwidth Efficient Transmission at High Data Rates', *IEEE Trans. Commun.*, Vol. 37, No. 8, pp. 804-813, Aug. 1989.

[9] M. Banu and A.E. Dunlop, 'Clock Recovery Circuits with Instantaneous Clocking', *Electron. Lett.*, Vol. 28, No. 23, pp. 2127-2130, Nov. 5, 1992.

[10] W.R. Bennett, 'Statistics of Regenerative Data Transmission', *Bell Syst. Tech. J.*, Vol. 37, pp. 1501-1542, Nov. 1958.

[11] J.W.M. Bergmans and H-W Wong-Lam, 'A Class of Data-Aided Timing-Recovery Schemes', *IEEE Trans. Commun.*, Vol. COM-43, No. 2/3/4, pp. 1819-1827, Feb./March/Apr. 1995.

[12] J.W.M. Bergmans, 'Efficiency of Data-Aided Timing Recovery Schemes', *IEEE Trans. Inform. Theory*, Vol. 41, No. 5, pp. 1397-1408, Sept. 1995.

[13] J.I. Brown, 'A Digital Phase and Frequency-Sensitive Detector', *Proc. IEEE*, Vol. 59, p. 717, Apr. 1971.

[14] R.D. Cideciyan, F. Dolivo, R. Hermann, W. Hirt and W. Schott, 'A PRML System for Digital Magnetic Recording', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 38-56, Jan. 1992.

[15] R.C. den Dulk, 'Improved Circuit Implementation of Adaptive Phase Comparators', *Electron. Lett.*, Vol. 24, No. 10, pp. 180-181, May 12, 1988.

[16] R.C. den Dulk, 'Digital PLL Lock-Detection Circuit', *Electron. Lett.*, Vol. 24, No. 14, pp. 880-882, Jul. 1988.

[17] D.L. Duttweiler, 'The Jitter Performance of Phase-Locked Loops Extracting Timing from Baseband Data Waveforms', *Bell Syst. Tech. J.*, Vol. 55, pp. 37-58, Jan. 1976.

[18] C.A. Ehrenbard and M.F. Tompsett, 'A Baud-Rate Line Interface for Two-Wire High-Speed Digital Subscriber Loops', Proc. GLOBECOM'82, pp. 931-935.

[19] J. Eijselendoorn and R.C. Den Dulk, 'Improved Phase-Locked Loop Performance with Adaptive Phase Comparators', *IEEE Trans. on Aerospace and Electronic Systems*, Vol. AES-18, No. 3, pp. 323-332, May 1982.

[20] T.T. Fang, 'Analysis of Self-Noise in a Fourth-Power Clock Regenerator', *IEEE Trans. Commun.*, Vol. COM-39, No. 1, pp. 133-140, Jan. 1991.

[21] L.E. Franks and J.P. Bubrouski, 'Statistical Properties of Timing Jitter in a PAM Timing Recovery Scheme', *IEEE Trans. Commun.*, Vol. COM-22, No. 7, pp. 913-920, Jul. 1974.

[22] L.E. Franks, 'Carrier and Bit Synchronization in Data Communication - A Tutorial Review', *IEEE Trans. Commun.*, Vol. COM-28, pp. 1107-1121, Aug. 1980.

[23] A. Gabor, 'High-Density Recording on Magnetic Tape', *Electronics*, Vol. 32, pp. 72-75, Oct. 16, 1959.

[24] F.M. Gardner, *Phaselock Techniques*, 2nd ed. New York: Wiley, 1979.

[25] F.M. Gardner, 'Charge-Pump Phase-Lock Loops', *IEEE Trans. Commun.*, Vol. COM-28, No.11, pp. 1849-1858, Nov. 1980.

[26] F.M. Gardner, 'A BPSK/QPSK Timing-Error Detector for Sampled Receivers', *IEEE Trans. Commun.*, Vol. COM-34, No. 5, pp. 423-429, May 1986.

[27] F.M. Gardner, 'Demodulator Reference Recovery Techniques Suited for Digital Implementation', ESA Final Report, ESTEC Contract No. 6847/86/NL/DG, Aug. 1988.

[28] R.D. Gitlin and J. Salz, 'Timing Recovery in PAM Systems', *Bell Syst. Tech. J.*, Vol. 50, No. 5, pp. 1645-1669, May-June 1971.

[29] N.H. Gottfried, 'Low Complexity Viterbi Detector for Magnetic Disc Drives', *IEE Proc.-E*, Vol. 140, No. 1, pp. 78-80, Jan. 1993.

[30] J.C. Haartsen and R.C. Den Dulk, 'Improved Circuit Implementation of Adaptive Phase Comparators', *Electron. Lett.*, Vol. 24, No. 10, pp. 575-576, May 1988.

[31] C.W. Helstrom, *Statistical Theory of Signal Detection*. Oxford: Pergamon press, 1960.

[32] C.R. Hogge, Jr., 'A Self Correcting Clock Recovery Circuit', *IEEE J. Lightwave Technol.*, Vol. LT-3, No. 6, pp. 1312-1314, Dec. 1985.

[33] A. Jennings and B.R. Clarke, 'Data-Sequence Selective Timing Recovery for PAM Systems', *IEEE Trans. Commun.*, Vol. COM-33, pp. 729-731, Jul. 1985.

[34] M. Kawai, 'Smart Optical Receiver With Automatic Decision Threshold Setting and Retiming Phase Alignment', *IEEE J. Lightwave Technol.*, Vol. LT-7, No. 11, pp. 1634-1640, Nov. 1989.

[35] K. Kobayashi, 'Simultaneous Adaptive Estimation and Decision Algorithm for Carrier Modulated Data Transmission Systems', *IEEE Trans. Commun. Techn.*, Vol. COM-19, pp. 268-280, June 1971.

[36] E.A. Lee and D.G. Messerschmitt, *Digital Communication*. Boston: Kluwer Academic Publishers, 1988.

[37] W.C. Lindsey and C.M. Chie, 'A Survey of Digital Phase-Locked Loops', *Proc. IEEE*, Vol. 69, pp. 410-431, Apr. 1981.

[38] P. Mallory, 'A Maximum Likelihood Bit Synchronizer', International Telemetring Conf., Proc., IV (1968), pp. 1-16.

[39] J.E. Mazo, 'Jitter Comparison of Tones Generated by Squaring and by Fourth-Power Circuits', *Bell Syst. Tech. J.*, Vol. 57, pp. 1489-1498, May-June 1978.

[40] J.E. Mazo, 'Optimum Sampling Phase of an Infinite Equalizer', *Bell Syst. Tech. J.*, Vol. 54, pp. 189-201, Jan. 1975.

[41] U. Mengali, 'Joint Phase and Timing Acquisition in Data-Transmission', *IEEE Trans. Commun.*, Vol. COM-25, No. 10, pp. 1174-1185, Oct. 1977.

[42] H. Meyr and G. Ascheid, *Synchronization in Digital Communications, Volume 1*. New York: Wiley, 1990.

[43] M. Moeneclaey, 'A Comparison of Two Types of Symbol Synchronizers for Which Self-Noise is Absent', *IEEE Trans. Commun.*, Vol. COM-31, No. 3, pp. 329-334, March 1983.

[44] K.H. Mueller and M. Müller, 'Timing Recovery in Digital Synchronous Data Receivers', *IEEE Trans. Commun.*, Vol. COM-24, No. 5, pp. 516-531, May 1976.

[45] M. Oerder, 'Derivation of Gardner's Timing-Error Detector from the Maximum Likelihood Principle', *IEEE Trans. Commun.*, Vol. COM-35, No. 6, pp. 684-685, June 1987.

[46] M. Oerder and H. Meyr, 'Digital Filter and Square Timing Recovery', *IEEE Trans. Commun.*, Vol. COM-36, No. 5, pp. 605-612, May 1988.

[47] E. Panayirci, 'Jitter Analysis of a Phase-Locked Digital Timing Recovery System', *IEE Proc.-I*, Vol. 139, No. 3, pp. 267-275, June 1992.

[48] A.E. Payzin, 'Analysis of a Digital Bit Synchronizer', *IEEE Trans. Commun.*, Vol. COM-31, No. 4, pp. 554-560, Apr. 1983.

[49] A. Pottbäcker, U. LLangmann, and H.-U. Schreiber, 'A Si Bipolar Phase and Frequency Detector IC for Clock Extraction up to 8 Gb/s', *IEEE J. Solid-State Circuits*, Vol. 27, No. 12, pp. 1747-1751, Dec. 1992.

[50] S.U.H. Qureshi, 'Timing Recovery for Equalized Partial-Response Systems', *IEEE Trans. Commun.*, Vol. COM-24, pp. 1326-1331, Dec. 1976.

[51] R. Raheli, A. Pincin and R. Colombo, 'On MMSE Timing Recovery Using Oversampling', *Proc. 1990 Int. Zürich Seminar on Digital Communications*, March 5-8, 1990, ETH Zürich, Switzerland, pp. 474-494.

[52] C.R. Rao, 'Minimum Variance and the Estimation of Several Parameters', *Proc. Cambridge Phil. Soc.*, Vol. 43, p. 280, 1947.

[53] W. Rosenkrantz, 'Phase-Locked Loops with Limiter Phase Detectors in the Presence of Noise', *IEEE Trans. Commun.*, Vol. COM-30, No. 10, pp. 2297-2304, Oct. 1982.

[54] E. Roza, 'Analysis of Phase-Locked Timing Extraction Circuits for Pulse-Code Transmission', *IEEE Trans. Commun.*, Vol. COM-22, pp. 1236-1249, Sept. 1974.

[55] B.R. Saltzberg, 'Timing Recovery for Digital Synchronous Data Transmission', *Bell Syst. Tech. J.*, Vol. 46, pp. 593-622, March 1967.

[56] H. Sari, L. Desperben and S. Moridi, 'Minimum Mean-Square Error Timing Recovery Schemes for Digital Equalizers', *IEEE Trans. Commun.*, Vol. COM-34, No. 7, pp. 694-702, Jul. 1986.

[57] H. Sari, H. Houeix, and G. Karam, 'Phase and Frequency Detectors for Clock Synchronization in High-Speed Optical Transmission Systems', *European Trans. on Telecommun. and Related Technologies*, Vol. 5, No. 5, pp. 635-641, Sept./Oct. 1994.

[58] M.K. Simon, 'Optimization of the Performance of a Digital-Data-Transition Tracking Loop', *IEEE Trans. Commun. Technol.*, Vol. COM-18, pp. 686-689, Oct. 1970.

[59] E.F. Stikvoort and J.A.C. van Rens, 'An All-Digital Bit Detector for Compact Disc Players', *IEEE J. Selected Areas Commun.*, Vol. SAC-10, No. 1, pp. 191-200, Jan. 1992.

[60] T. Suzuki, H. Takatori, M. Ogawa and K. Tomooka, 'Line Equalizer for a Digital Subscriber Loop Employing Switched Capacitor Technology', *IEEE Trans. Commun.*, Vol. COM-30, pp. 2074-2082, Sept. 1982.

[61] M. Tanaka and N. Watanabe, *Bittaktsignal-Rückgewinnungsschaltung*, German patent DE 31 16 054 C 2, issued Dec. 12, 1991.

[62] R.C.E. Thomas, 'Frequency Comparator Performs Double Duty', *EDN*, pp. 29-33, Nov. 1, 1970.

[63] Y.P. Tsividis and J.O. Voorman (Eds.), *Integrated Continuous-Time Filters*. New York: IEEE Press, 1993.

[64] C.-P.J. Tzeng, D.A. Hodges and D.G. Messerschmitt, 'Timing Recovery in Digital Subscriber Loops Using Baudrate Sampling', *IEEE J. Selected Areas Commun.*, Vol. SAC-4, pp. 1302-1311, Nov. 1986.

[65] K.J. Wouda and W. Reijntjens, *Arrangement for Generating a Clock Signal*, European patent No. 0 162 505 (filed 18.04.85).

11

PHASE-LOCKED LOOPS

11.1 INTRODUCTION

The timing-recovery subsystem of the receiver determines the sampling instants that are used for data detection. Since these instants are discrete, any phase-locked-loop (PLL) that is embedded in the timing-recovery subsystem has fundamentally a discrete-time nature. Discrete-time PLLs have received relatively little attention in the literature (see [16] for an overview). Fortunately, their properties approach those of analog PLLs in many cases of practical interest, as we will see. This renders a range of books and review articles on analog PLLs relevant [3], [4], [8], [10], [14], [15], [20], [6], [18].

In the PLL literature it is customary to work with angular frequencies. In this chapter we adhere to this custom wherever meaningful. For similar reasons we shall use the term *phase detector* (PD) in lieu of 'timing-error detector' (TED) throughout the chapter, and use z-transforms instead of $\mathcal{D}$-transforms. (We recall that $z = \mathcal{D}^{-1}$.)

The chapter is organized as follows. In Section 11.2 the basic structure of the discrete-time PLL is described, and its components are reviewed. Section 11.3 analyzes the linear behavior of a first-order PLL and its relation to the continuous-time first-order PLL. A similar analysis is given in Section 11.4 for the high-gain second-order PLL, and more briefly in Section 11.5 for the general second-order PLL. Nonlinear PLL behavior (acquisition, unlock) is reviewed in Section 11.6. By way of illustration, Section 11.7 outlines the design of a data-aided timing-recovery loop for a Class IV partial-response system.

11.2 PLL STRUCTURE

The phase-locked loop (PLL) consists of three basic building blocks: a phase detector (PD), a loop filter (LF), and a voltage-controlled oscillator (VCO) (Fig. 11.1).

The input signal $r(t)$ has a phase ϕ, and the VCO output has a phase ψ. The PD output χ is ideally a measure of the phase error $\Delta \triangleq \phi - \psi$. Just as in Chapters 9 and 10, ϕ, ψ and Δ are normalized in symbol intervals T, i.e. a value $\Delta = 0.1$ corresponds to a

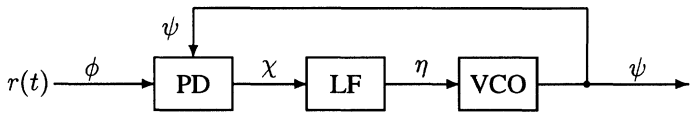


Figure 11.1: Phase-locked loop.

sampling-phase error of 10 percents of a symbol interval. The PD output χ is filtered by a loop filter (LF) that determines the order of the loop and various other loop properties. The frequency of the VCO is controlled by the filtered error η. Whenever the PLL is in lock, the VCO phase ψ will track ϕ. Before deriving PLL properties we review the basic PLL components.

Phase detector In the PD model of Fig. 11.2, the PD output χ_k depends on the phase error Δ_k according to $\chi_k = \rho(\Delta_k) + u_k + z_k$ where ρ is the PD transfer function while u_k and z_k are noise- and data-induced disturbance components. We recall from Chapters 9 and 10 that ρ describes the *average* output of the PD across the ensemble of possible data sequences. For random data, the instantaneous PD output tends to vary dynamically depending on the 'local' timing content of the data pattern as measured, crudely speaking, by the presence or absence of data transitions. The PLL effectively performs an averaging operation on the PD output and thus suppresses these dynamic variations. For this reason it is the *average* output ρ that primarily determines PLL properties.

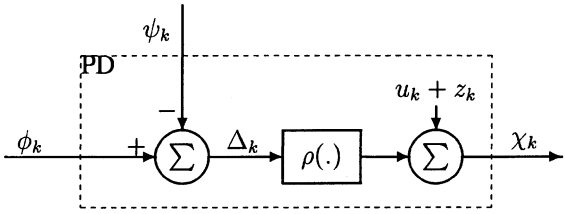


Figure 11.2: Phase-domain PD model.

A well-designed PD has a continuous transfer function $\rho(\Delta)$ that crosses zero for $\Delta = 0$. For small phase errors we may linearize ρ according to $\overline{\rho}(\Delta) \simeq K_d\Delta$ where K_d is said to be the *PD gain*. This linearization allows us to refer the noise $u_k + z_k$ back to the PD input. We arrive at the linearized PD model of Fig. 11.3, which is applicable when the PLL is in lock.

The error $\Delta = \phi - \psi$ is corrupted by input-referred phase noise $\theta_k \triangleq (u_k + z_k)/K_d$ before being scaled by the PD gain. The loop is desired to track ϕ_k while rejecting θ_k. These two requirements conflict. A practical compromise is to choose PLL bandwidth only slightly larger than that of ϕ_k (see also Section 11.3.4).

Loop filter, voltage-controlled oscillator Just as for the PD, implementation approaches for these two components are manifold. They can be analog, digital, or a mix-

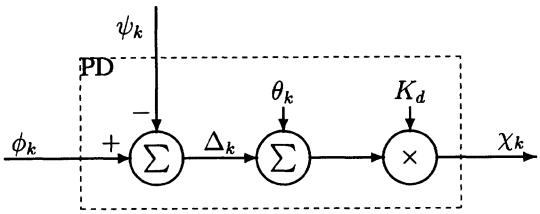


Figure 11.3: Linearized PD model.

ture of the two, and are sometimes so intermingled as to be hardly distinguishable. In all cases, however, the model of Fig. 11.4 applies at least approximately.

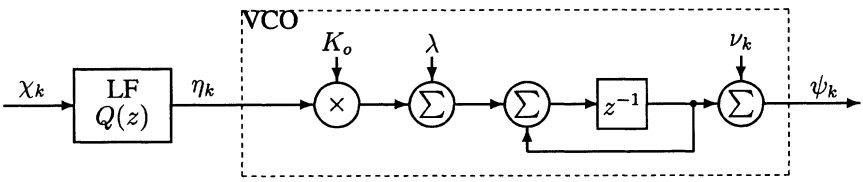


Figure 11.4: Discrete-time model of loop filter and VCO.

The discrete-time LF of Fig. 11.4 is followed by a phase-domain model of the VCO. At the heart of this model is an ideal integrator, which represents the frequency-to-phase conversion of the VCO. The DC offset λ at the integrator input accounts for the offset between the *free-running* or *quiescent* frequency of the VCO and the symbol rate $1/T$. This offset is normalized with respect to the symbol rate $1/T$, i.e. a value $\lambda = 0.01$ means that the free-running frequency of the VCO is 1% above the data rate.

The *sensitivity* of the VCO determines the constant gain factor K_o. The VCO output is inevitably corrupted by VCO phase noise ν_k. This noise is typically small for crystal-controlled and LC-type VCOs, but may be significant for integrated oscillators. In accordance with physical reality, changes of the VCO input voltage η do not affect its output phase ψ instantaneously; the delay in Fig. 11.4 amounts to 1 symbol interval.

As to the LF, 3 structures are frequently encountered:

1. a linear weighing factor K_p (Fig. 11.5.a). This degenerate filter, along with the first-order filtering by the VCO, leads to a *first-order PLL*. Main disadvantage of this loop is its inability to handle frequency offsets λ with zero error.

2. a proportional-plus-integral filter with weighing factors K_p and K_f (Fig. 11.5.b). This leads to a *second-order PLL* of the *high-gain* type. The added integration helps to handle frequency offsets.

3. a proportional part plus a leaky integrator (Fig. 11.5.c). This model arises, for example, when a passive analog LF is used in conjunction with an analog VCO

(see Appendix 11A). When leakage is significant, we have a *low-gain second-order PLL*. We will devote little attention to this PLL because its performance is comparatively poor.

PLLs of third or higher order are seldom applied for timing recovery. They do, however, show up in other applications [8].

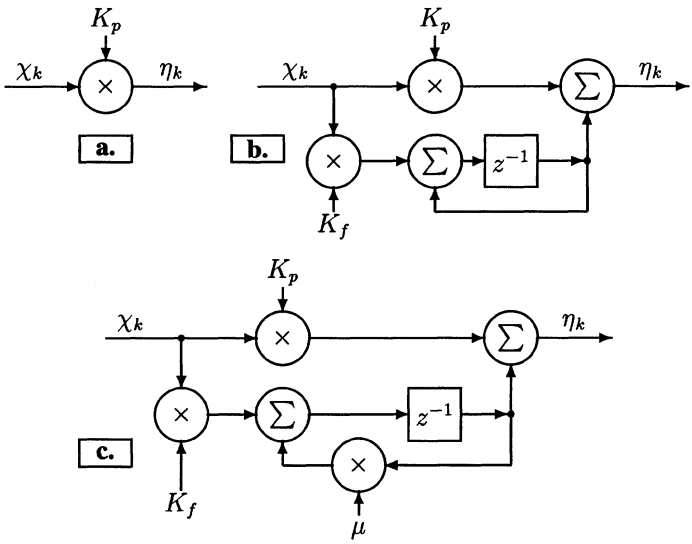


Figure 11.5: Three discrete-time loop filters. a. Proportional part only; b. proportional plus integral part; c. proportional part plus leaky integrator.

As an example, Appendix 11A derives the equivalent discrete-time model of Fig. 11.4 for an analog LF/VCO cascade. The three analog LF types that are considered each lead to one of the above discrete-time LFs. The appendix can be helpful to design a continuous-time LF and VCO in accordance with any discrete-time PLL specifications. The digital counterpart of the VCO is the numerically-controlled oscillator (NCO) as discussed in Section 9.2.3 and Appendix 9A.2.

The following sections concentrate on linear PLL behavior for the above three LFs. It is assumed that the PLL is in lock, so that the linearized PD model of Fig. 11.3 applies. Conditions of validity of this assumption are reviewed in Section 11.6, which focuses on nonlinear PLL behavior.

11.3 LINEAR ANALYSIS OF FIRST-ORDER PLL

First-order loops are of restricted practical interest because of their poor ability to handle frequency errors. Nevertheless, their analysis is relevant for the light it sheds on higher-order loops.

For the first-order PLL, the LF degenerates into a constant gain factor as in Fig. 11.5.a. For simplicity we lump the constant gain factors K_d, K_p and K_o into a total open-loop gain $K_t = K_d K_p K_o$. For the time being we neglect the input-referred phase noise θ_k, the frequency offset λ, and the phase noise ν. This leads to Fig. 11.6.

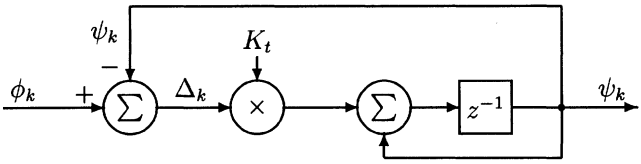


Figure 11.6: Phase-domain model of first-order PLL.

11.3.1 Phase and error transfer functions

Denote the z transforms of ϕ_k, ψ_k and Δ_k by $\Phi(z)$, $\Psi(z)$ and $\Delta(z)$. The phase- and error transfer functions $G_\psi(z) \triangleq \Psi(z)/\Phi(z)$ and $G_\Delta(z) \triangleq \Delta(z)/\Phi(z)$ are given by

$$G_\psi(z) = \frac{K_t}{(z-1)+K_t} \tag{11.1}$$

and

$$G_\Delta(z) = 1 - G_\psi(z) = \frac{z-1}{(z-1)+K_t}. \tag{11.2}$$

To obtain the transfer magnitudes we evaluate these expressions on the unit circle ($z = e^{j\omega T}$). The result is

$$|G_\psi(e^{j\omega T})|^2 = \frac{K_t^2}{K_t^2 + 4(1-K_t)\sin^2\frac{\omega T}{2}} \tag{11.3}$$

and

$$|G_\Delta(e^{j\omega T})|^2 = \frac{4\sin^2\frac{\omega T}{2}}{K_t^2 + 4(1-K_t)\sin^2\frac{\omega T}{2}}. \tag{11.4}$$

These functions are depicted in Fig. 11.7. For $K_t \ll 1$, the case of greatest practical interest, their shape becomes independent of K_t and $\omega_c \triangleq K_t/T$ demarcates the angular -3 dB cut-off frequency. The PLL exhibits peaking for $K_t \in (1,2)$. In this region, inputs at all frequencies are amplified and the PLL cannot function properly. For $K_t < 1$, the loop performs a low-pass filtering on phase inputs. The error response has a high-pass nature; that is the loop tracks low-frequency changes but cannot track high frequencies.

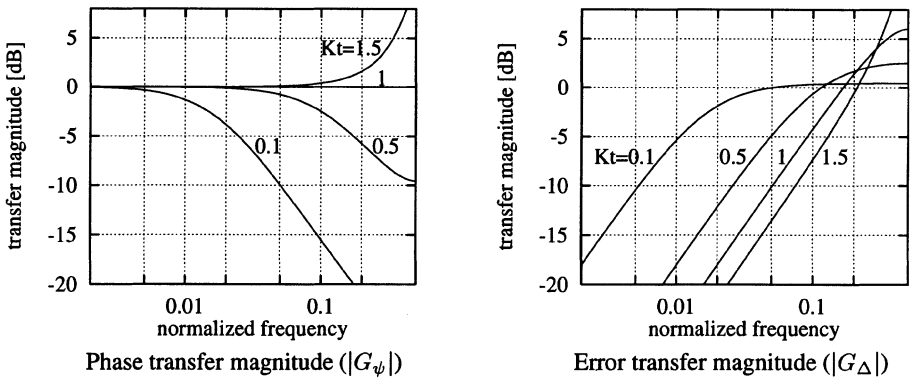


Figure 11.7: Transfer magnitudes of first-order discrete-time PLL as a function of the normalized frequency $\Omega = \omega T/(2\pi)$.

11.3.2 Stability range

Looking at (11.1) and (11.2), we see that the PLL is stable whenever the zero of $z - 1 + K_t$ is inside the unit circle, i.e. when $0 \leq K_t \leq 2$. The baseline PLL model of Fig. 11.6 does not account for a pure delay or *latency* that is sometimes introduced by the phase detector (see Chapter 10). Further delays may need to be introduced in the loop for implementation reasons, particularly at high operating speeds. Any loop delay will affect loop properties and in particular the range of loop gains K_t for which the PLL remains stable. In Appendix 11B it is shown that a loop delay of M symbol intervals reduces the stability range to

$$0 \leq K_t \leq 2 \sin \frac{\pi}{2(2M+1)}. \tag{11.5}$$

The delay ranges according to (11.5) are depicted as vertical bars in Fig. 11.8.

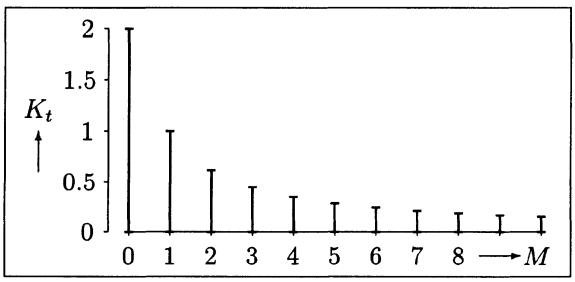


Figure 11.8: Stability range of first-order discrete-time PLL as a function of the loop delay M.

For $M = 0$ we have the configuration of Fig. 11.6. For $M = 1$ the stability range is already halved with respect to $M = 0$. For $M \gg 1$ the region extends until $K_t \simeq$

$\pi/(2M+1)$. For example, for $M = 10$ we must take $K_t \leq \pi/21 \simeq 0.15$. Problem 11.2 reveals that PLL properties closely approach those of a delayless loop when K_t is well within the stability range. This is so, coarsely speaking, when the *time constant* $\tau \triangleq 1/\omega_c = T/K_t$ of the loop is much larger than the loop delay MT. A well-engineered loop will satisfy this rule of thumb.

11.3.3 Dynamic behavior

Response to a phase step Let the PD input ϕ_k be a unit step

$$\phi_k = \begin{cases} 0 & \text{if } k < 0, \text{ and} \\ 1 & \text{if } k \geq 0. \end{cases} \tag{11.6}$$

Then the phase error Δ_k is given by

$$\Delta_k = \begin{cases} 0 & \text{if } k \leq 0, \text{ and} \\ (1 - K_t)^k & \text{if } k \geq 1. \end{cases} \tag{11.7}$$

The error Δ_k decays by $-20 \log_{10} |1 - K_t| \simeq 20K_t/\ln 10$ dB/iteration, where the approximation holds for $K_t \ll 1$. It is less than 5% of its initial value after approximately 3 time constants, i.e. after $3\tau/T = 3/K_t$ iterations. Oscillatory behavior occurs for $K_t \in (1, 2)$.

Response to a frequency step We next consider a phase ramp (frequency step)

$$\phi_k = \begin{cases} 0 & \text{if } k < 0, \text{ and} \\ \lambda k & \text{if } k \geq 0, \end{cases} \tag{11.8}$$

where λ is the relative frequency jump (i.e. $\lambda = 0.01$ means that frequency increases by 1%). The response of the PLL to this PD input is analyzed in Problem 11.4. Unlike the phase step, Δ_k does not tend to zero as $k \to \infty$. To determine the limiting error Δ_∞ we note that this error depends only on the difference of ϕ and ψ. The absolute frequencies of both PD input signals are irrelevant; only their offset λ counts. In Fig. 11.9, this offset is thought of as being introduced by the VCO. The integrator output has settled down at some steady-state phase ψ. This is only possible if its input is zero, i.e. if $K_t\Delta_\infty - \lambda = 0$. Hence

$$\Delta_\infty = \frac{\lambda}{K_t} = \lambda\frac{\tau}{T}. \tag{11.9}$$

This residual error is entirely due to the frequency offset and is sometimes called *loop stress*. Loop stress can be altogether avoided with the aid of a high-gain second-order PLL (see Exercise 11.2).

Example 11.1 In systems with little excess bandwidth, timing errors Δ larger than a few percents of a symbol interval T may degrade bit-error rates significantly. If we require that the loop

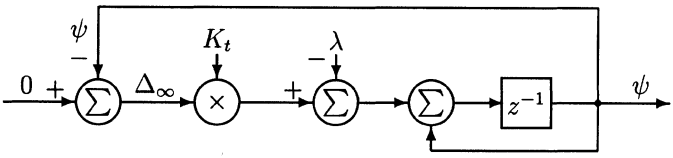


Figure 11.9: Simplified model of first-order PLL to assess the effect of offset of free-running VCO-frequency and input frequency.

stress Δ_∞ be smaller than 0.02 and have a frequency offset λ of 2%, then K_t must be at least $0.02/0.02 = 1$. This large gain is incompatible with values $K_t \ll 1$ that are desirable for noise suppression (see Section 11.3.4) and for stability when delay is present in the loop. For a more practical value $K_t \simeq 0.01$, frequency offsets may be at most 2.10^{-4}, i.e. 200 ppm. In practice this small tolerance is often unattainable (see Examples 9.11 and 9.12) and this is the principal reason why a first-order PLL is not often used.

11.3.4 Noise suppression and dynamic tracking errors

After transients due to deterministic changes of ϕ_k have died out, the PLL is desired to track random fluctuations of ϕ_k while suppressing input-referred phase noise θ_k. The phase detector responds to the *sum* of ϕ_k and θ_k (see Fig. 11.3). For this reason ϕ_k and θ_k can only be discriminated from each other in as far as they have no spectral overlap. Typically ϕ_k is a low-pass process (see Section 9.2.1), while θ_k is more or less white (Fig. 11.10). In that case PLL bandwidth should be only slightly larger than that of ϕ_k, so that all variations of ϕ_k will be tracked while θ_k will be suppressed as well as possible. It is, of course, intrinsically impossible to suppress the low-pass portion of θ_k while tracking that of ϕ_k.

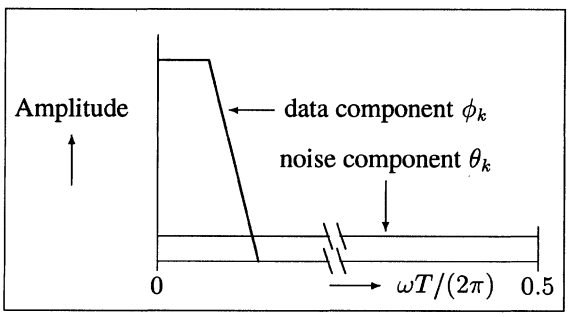


Figure 11.10: Typical power spectral densities of data and noise components of PD input.

The noise component of ψ_k has power

$$\sigma_\psi^2 = \frac{T}{2\pi} \int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} P_\theta(e^{j\omega T}) |G_\psi(e^{j\omega T})|^2 d\omega. \tag{11.10}$$

Except for a minus sign, this same noise component is present in the sampling phase error Δ_k (see Fig. 11.10), and its power σ_Δ^2 equals σ_ψ^2. For white noise θ_k with variance σ_θ^2 we obtain

$$\sigma_\Delta^2 = \sigma_\theta^2 B_l \tag{11.11}$$

where the *normalized equivalent noise bandwidth* B_l of the loop is given by

$$B_l \triangleq \frac{T}{2\pi}\int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} |G_\psi(e^{j\omega T})|^2 d\omega. \tag{11.12}$$

This bandwidth is normalized with respect to the symbol rate and thereby dimensionless. In the absence of any filtering (i.e. when $|G_\psi| = 1$ for all ω) we have $B_l = 1$. The integral of (11.12) may be evaluated analytically for the function $G_\psi(e^{j\omega T})$ at hand. The result is

$$B_l = \frac{K_t}{2 - K_t}. \tag{11.13}$$

If noise θ_k is not white then we may exploit the fact that its power spectral density $P_\theta(e^{j\omega T})$ is normally a smooth function of frequency. In the usual event that the loop bandwidth is much smaller than the signaling rate ($K_t \ll 1$), σ_Δ^2 will be governed by the power spectral density of θ_k at DC ($e^{j\omega T} = 1$) according to

$$\sigma_\Delta^2 \simeq P_\theta(1) B_l \simeq P_\theta(1)\frac{K_t}{2}. \tag{11.14}$$

For optimum noise suppression, B_l (or, equivalently, K_t) should be as small as possible.

Example 11.2 Let us assume that ϕ_k has significant spectral content up to about 0.1% of the signaling rate $1/T$, i.e. for normalized frequencies $\Omega = \omega T/(2\pi)$ up to 10^{-3}. To ensure adequate tracking, the normalized PLL cut-off frequency $\Omega_c = \omega_c T/(2\pi) = K_t/(2\pi)$ must be at least 10^{-3}, i.e. K_t must be in the order of 10^{-2}. Let us further assume that θ_k is uncorrelated and has an RMS magnitude of 0.5, so that $\sigma_\theta^2 = P_\theta(1) = 0.25$.[1] Then $\sigma_\Delta = \sigma_\theta\sqrt{B_l} \simeq 0.035$. Thus the sampling phase ψ will vary randomly about its desired value with an RMS magnitude of about 3.5% of a symbol interval T.

11.3.5 Relation to continuous-time PLL

The discrete-time PLL may be related to its continuous-time counterpart via the method of replacing differentials by forward differences [7, Sec. 8.3.2]. This method is based on the fact that $z = e^{j\omega T} \simeq 1 + j\omega T$ for small $|\omega T|$, whence $z - 1 \simeq j\omega T$. This relation allows the PLL transfer functions (11.1) and (11.2) to be approximated as

$$G_\psi(\omega T) \simeq \frac{K_t}{K_t + j\omega T} = \frac{\omega_c}{\omega_c + j\omega} \tag{11.15}$$

and

$$G_\Delta(\omega T) \simeq \frac{j\omega T}{K_t + j\omega T} = \frac{j\omega}{\omega_c + j\omega}. \tag{11.16}$$

[1] We recall that the phases ϕ_k and θ_k are normalized in symbol intervals T.

The right-hand sides may be recognized as the transfer functions of the first-order continuous-time PLL shown in Fig. 11.11 (the continuous-time integrator in Fig. 11.11 has transfer function $1/(j2\pi\Omega) = 1/(j\omega T)$). The implication is that the discrete-time PLL mimics its continuous-time counterpart for small $|\omega T|$. Beyond that range we are in the PLL stopband in the usual event that $K_t \ll 1$. Here $G_\psi \simeq 0$ and $G_\Delta \simeq 1$ for both PLL types. Consequently, both PLL types behave essentially identically at all frequencies of interest when $K_t \ll 1$ (see also Fig. 11.12).

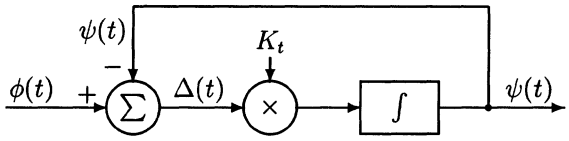


Figure 11.11: Phase-domain model of first-order continuous-time PLL.

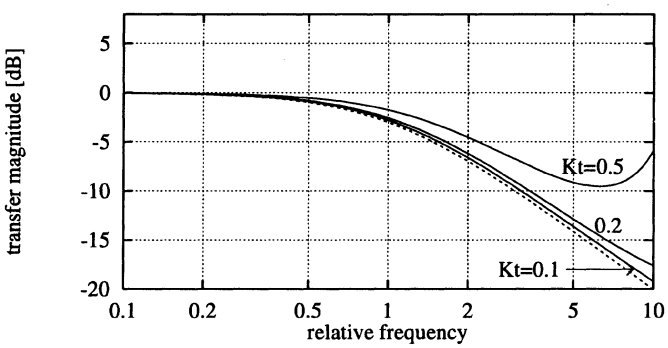


Figure 11.12: Transfer magnitude $|G_\psi|$ versus relative frequency ω/ω_c for first-order discrete-time PLL (solid) and continuous-time PLL (dashed) for various open-loop gains $K_t = \omega_c T$. The behavior of the discrete-time PLL approaches that of the continuous-time PLL as $K_t \ll 1$.

The transfer functions of the continuous-time PLL depend only on the ratio of ωT and $\omega_c T = K_t$, i.e. their shape is independent of K_t. For the discrete-time PLL this is only true when $K_t \ll 1$. Similar remarks apply to the time-domain responses of both PLLs.

11.3.6 Effect of VCO Phase Noise

We now consider the phase noise ν_k that is generated by the VCO (see Fig. 11.4). In a continuous-time VCO there are several phase-noise components which all have power spectral densities of the form $|\omega|^\alpha$ where α is some real-valued number. These components combine spectrally to produce the total power spectral density $P_\nu(e^{j\omega T})$ of the phase-noise process (Fig. 11.13). For more background on these noise spectra

and their measurement we refer to [18, sec. 3.5]. In data transmission the VCO serves to demarcate decision instants t_k with a nominal spacing of T seconds. This gives the VCO a discrete-time nature, and the phase noise spectrum becomes periodic with a fundamental interval that extends from $\omega T = -\pi$ to $\omega T = \pi$.

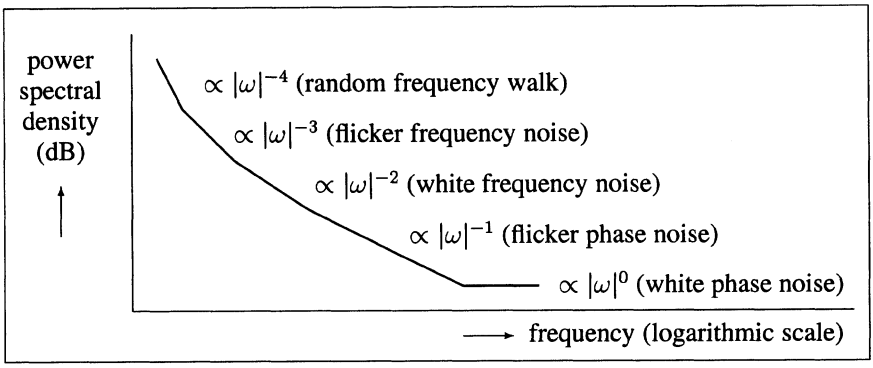


Figure 11.13: General form of the phase noise spectrum of a continuous-time VCO.

Exercise 11.1 The spectrum of 11.13 exhibits a singularity at DC that may either be an anomaly of the model or may actually manifest itself in practice. Argue that this singularity need not impede the proper operation of the PLL.

By way of illustration we will determine the effect of white frequency noise on loop performance. For this type of noise, the instantaneous VCO frequency fluctuates randomly about the average frequency. The fluctuations are white and have an RMS magnitude $\overline{\lambda}$ relative to an average VCO frequency of 1, i.e. $\overline{\lambda} = 0.01$ corresponds to an RMS fluctuation of 1 percent of the symbol rate. Problem 11.12 reveals that this type of phase noise has power spectral density

$$P_\nu(e^{j\omega T}) = \left(\frac{\overline{\lambda}}{2\sin\frac{\omega T}{2}}\right)^2. \tag{11.17}$$

The phase error Δ_k due to ν_k has variance

$$\sigma_\Delta^2 = \frac{T}{2\pi}\int_{-\frac{\pi}{T}}^{\frac{\pi}{T}} P_\nu(e^{j\omega T})|G_\Delta(e^{j\omega T})|^2 d\omega. \tag{11.18}$$

This integral may be evaluated analytically for the functions P_ν and G_Δ at hand. The result is $\sigma_\Delta^2 = \overline{\lambda}^2/[K_t(2-K_t)]$. It is worth observing that σ_Δ^2 grows without bound as K_t is decreased to zero. Phase noise may, therefore, become particularly cumbersome when loop gains are small. The same conclusion applies to second- and higher-order loops.

Example 11.3 Let us assume that the RMS frequency fluctuation amounts to 0.1% of the signaling rate ($\overline{\lambda} = 10^{-3}$) and that $K_t = 5.10^{-3}$. Then $\sigma_\Delta \simeq 0.01$, i.e. VCO phase noise induces an RMS sampling-phase error of 1% of a symbol interval. Reduction of K_t by a factor of 100 would increase this error to 10% of a symbol interval. This is unacceptably large for virtually all applications.

11.4 LINEAR ANALYSIS OF HIGH-GAIN SECOND-ORDER PLL

The loop filter of this PLL is characterized by the z transform $Q(z) = K_p + K_f/(z-1)$ for some $K_p, K_f \in [0, \infty)$ (compare Fig. 11.5.b). In Appendix 11A it is shown for an active first-order analog loop filter how the continuous-time filter parameters determine K_p and K_f. In the forthcoming analysis we assume again that the PLL is in lock, so that the linear PD model of Fig. 11.3 applies. Also, we neglect VCO phase noise and frequency offset, and we define the total open-loop phase- and frequency- gains $K_p^t \triangleq K_d K_p K_o$ and $K_f^t \triangleq K_d K_f K_o$. This leads to the discrete-time PLL model of Fig. 11.14. From left to right, this model contains an ideal phase detector with gain 1, a loop filter with modified gain factors K_p^t and K_f^t, and an ideal phase-domain VCO with gain 1.

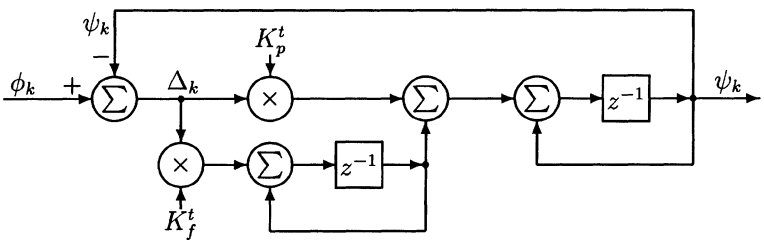


Figure 11.14: Phase-domain model of high-gain second-order PLL.

11.4.1 Phase and error transfer functions

For the topology of Fig. 11.14 one finds that

$$G_\psi(z) = \frac{K_o K_d Q(z)}{(z-1) + K_0 K_d Q(z)} = \frac{K_f^t + K_p^t(z-1)}{K_f^t + K_p^t(z-1) + (z-1)^2} \tag{11.19}$$

and

$$G_\Delta(z) = 1 - G_\psi(z) = \frac{(z-1)^2}{K_f^t + K_p^t(z-1) + (z-1)^2}. \tag{11.20}$$

Corresponding frequency responses are obtained by putting $z = e^{j\omega T}$, and may be compared to those of the continuous-time PLL. The latter functions are given by

$$G_\psi^c(\frac{\omega}{\omega_n}) = \frac{1 + 2j\zeta\frac{\omega}{\omega_n}}{1 + 2j\zeta\frac{\omega}{\omega_n} - (\frac{\omega}{\omega_n})^2} \tag{11.21}$$

and

$$G_\Delta^c(\frac{\omega}{\omega_n}) = \frac{(\frac{\omega}{\omega_n})^2}{1 + 2j\zeta\frac{\omega}{\omega_n} - (\frac{\omega}{\omega_n})^2}. \tag{11.22}$$

Here ω_n is an angular frequency that is referred to as the *undamped natural frequency* or just *natural frequency* of the PLL. Furthermore, ζ is called the damping factor and is typically in the order of 0.5 to 2. For a given damping factor, G_ψ^c and G_Δ^c are fully determined by the *relative frequency* ω/ω_n (Figs. 11.15 and 11.16). The damping factor ζ determines the amount of *peaking* in both functions, and also affects the bandwidth of G_ψ^c to a certain extent. The error transfer magnitude increases by 12 dB/octave for low frequencies, as opposed to 6 dB/octave for the first-order loop.

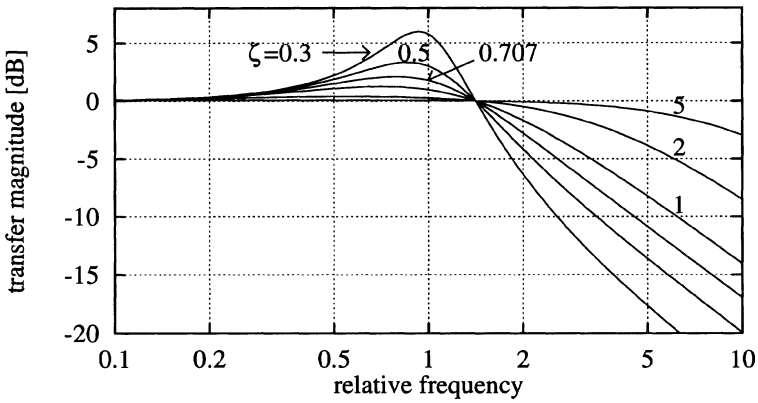


Figure 11.15: Phase transfer magnitude $|G_\psi^c|$ versus relative frequency ω/ω_n for continuous-time high-gain second-order PLL. The damping factor ζ is used as a parameter.

Just as for the first-order PLL, we may relate the transfer functions of the continuous-time PLL to those of their discrete-time counterpart via the approximation of continuous-time differentiations by forward differences [7, Sec. 8.3.2]. To this end we replace $j\omega T$ by $z - 1$. The resulting discrete-time transfer functions describe a discrete-time PLL with $K_f^t = (\omega_n T)^2$ and $K_p^t = 2\zeta\omega_n T$. This suggests that we assign to the discrete-time PLL of Fig. 11.14 a natural frequency ω_n^d and damping factor ζ^d according to the relations

$$\omega_n^d T = \sqrt{K_f^t} \qquad \zeta^d = \frac{K_p^t}{2\sqrt{K_f^t}}$$

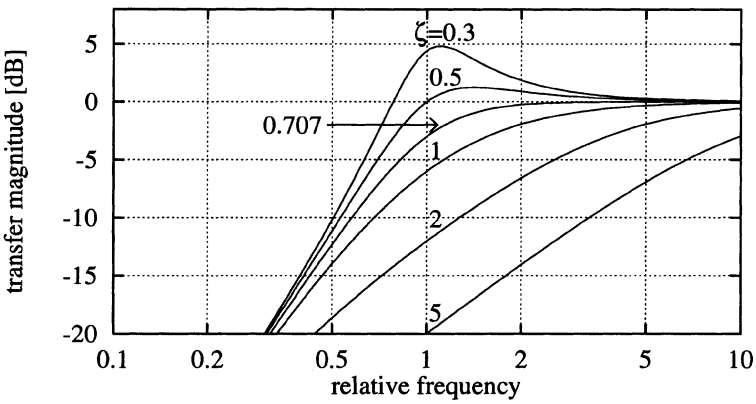


Figure 11.16: Error transfer magnitude $|G_\Delta^c|$ versus relative frequency ω/ω_n for continuous-time high-gain second-order PLL. The damping factor ζ is used as a parameter.

$$K_p^t = 2\zeta^d\omega_n^d T \qquad K_f^t = (\omega_n^d T)^2. \tag{11.23}$$

In terms of these quantities we may write the transfer functions of the discrete-time PLL as

$$G_\psi(z) = \frac{(\omega_n^d T)^2 + 2\zeta^d\omega_n^d T(z-1)}{(\omega_n^d T)^2 + 2\zeta^d\omega_n^d T(z-1) + (z-1)^2} \tag{11.24}$$

and

$$G_\Delta(z) = \frac{(z-1)^2}{(\omega_n^d T)^2 + 2\zeta^d\omega_n^d T(z-1) + (z-1)^2}. \tag{11.25}$$

For $0 \le \zeta^d < 1$ both functions have a complex conjugate pole pair $z_{1,2} = 1 - \zeta^d\omega_n^d T \pm j\omega_d^d T$, where $\omega_d^d \triangleq \omega_n^d\sqrt{1-(\zeta^d)^2}$ is referred to as the *damped natural frequency*. For $\zeta^d > 1$ there are two real-valued poles $z_{1,2} = 1 - \omega_n^d T(\zeta^d \pm \sqrt{(\zeta^d)^2 - 1})$.

In the usual event that $\omega_n^d T \ll 1$, the discrete-time PLL of Fig. 11.14 essentially mimics its continuous-time counterpart. Figs. 11.15 and 11.16 are then applicable with ω_n and ζ replaced by ω_n^d and ζ^d. To illustrate the accuracy of the approximation, Fig. 11.17 shows PLL transfer magnitudes $|G_\psi|^2$ corresponding to (11.24) for various values of $\omega_n^d T$ for a typical damping factor $\zeta^d = 1/\sqrt{2} \simeq 0.707$. Also depicted is the corresponding continuous-time function according to (11.21). The discrete-time characteristic is virtually indistinguishable from the continuous-time characteristic for $\omega_n^d T < 0.1$. This covers most cases of practical interest.

11.4.2 Stability region

In Appendix 11B it is shown that this region is the intersection of the regions $0 \le \omega_n^d T \le 2$ and $\omega_n^d T/2 \le \zeta \le \omega_n^d T/4 + 1/(\omega_n^d T)$. The appendix also covers the effect of an additional loop delay of M symbol intervals. It is found that this delay reduces

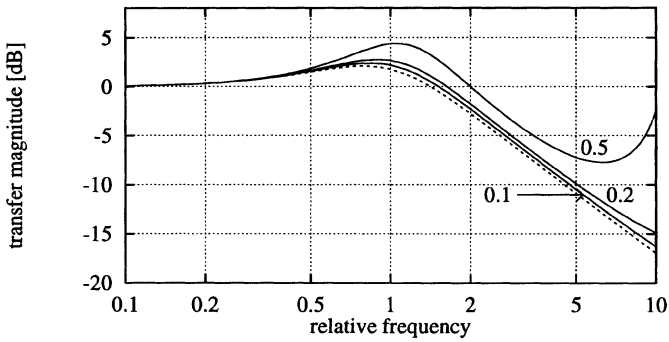


Figure 11.17: Phase transfer magnitude $|G_\psi|$ versus relative frequency ω/ω_n^d for discrete-time high-gain second-order PLL with damping factor $\zeta = 1/\sqrt{2} \simeq 0.707$ and $\omega_n^d T = 0.1, 0.2$ and 0.5. Also depicted is the corresponding continuous-time characteristic (dashed).

the stability range by roughly a factor $3M + 1$ in the ω_n^d-direction (Fig. 11.18). For $\zeta^d \ll 1$ the outer edge of the stability region is the line $2\zeta^d = (M + 1)\omega_n^d T$. Again, PLL behavior for $M \geq 1$ closely approaches that of a delayless PLL whenever the PLL is well within the stability region.

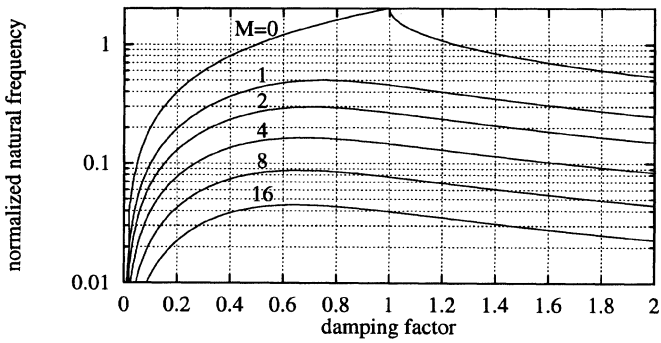


Figure 11.18: Outer edges of stability region of second-order discrete-time PLL for various loop delays M as a function of normalized natural frequency $\omega_n^d T$ and damping factor ζ^d.

Example 11.4 For a loop delay M of 5 symbol intervals and a damping factor $\zeta^d = 1/\sqrt{2} \simeq 0.707$, Fig. 11.19 depicts transfer magnitudes $|G_\psi|$ for various normalized natural frequencies $\omega_n^d T$. As conjectured, the curves closely approach those of Fig. 11.17 for $\omega_n^d T \ll 2/(3M+1) = 0.125$.

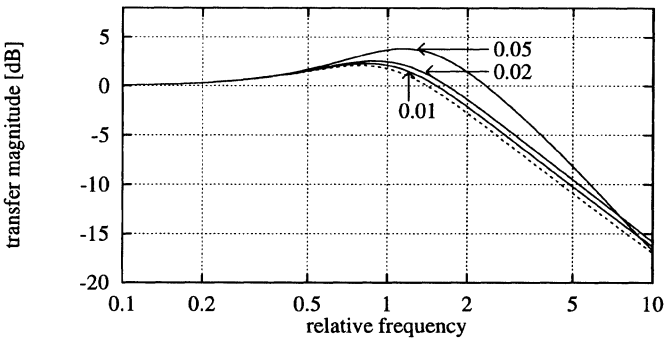


Figure 11.19: Transfer magnitude $|G_\psi|$ versus relative frequency ω/ω_n^d for high-gain second-order discrete-time PLL with damping factor $\zeta^d = 1/\sqrt{2} \simeq 0.707$ and a loop delay of 5 symbol intervals. The normalized natural frequency $\omega_n^d T$ is used as a parameter.

11.4.3 Dynamic behavior

Responses of the second-order PLL to a phase- and frequency step at moment $k = 0$ may be determined in a straightforward though somewhat laborious fashion. The results are summarized in Table 11.1. The variables β, ω_d^d, θ and ξ in the table are defined as follows:

$$\beta \triangleq \sqrt{1 - 2\zeta^d\omega_n^d T + (\omega_n^d T)^2}; \qquad \omega_d^d \triangleq \omega_n^d\sqrt{1-(\zeta^d)^2};$$

$$\theta \triangleq \tan^{-1}\frac{\omega_d^d T}{1-\zeta^d\omega_n^d T}; \qquad \xi \triangleq \frac{1}{2}\ln\frac{1-\zeta^d\omega_n^d T + \omega_n^d T\sqrt{(\zeta^d)^2-1}}{1-\zeta^d\omega_n^d T - \omega_n^d T\sqrt{(\zeta^d)^2-1}}.$$

	Phase step	Frequency step
$\zeta^d < 1$	$\beta^k(\cos k\theta - \frac{\zeta^d}{\sqrt{1-(\zeta^d)^2}}\sin k\theta)$	$\frac{1}{\omega_d^d}\beta^k \sin k\theta$
$\zeta^d = 1$	$(1-\omega_n^d T)^k(1 - k\frac{\omega_n^d T}{1-\omega_n^d T})$	$k(1-\omega_n^d T)^{k-1}$
$\zeta^d > 1$	$\beta^k(\cosh k\xi - \frac{\zeta^d}{\sqrt{(\zeta^d)^2-1}}\sinh k\xi)$	$\frac{1}{\omega_n^d T\sqrt{(\zeta^d)^2-1}}\beta^k \sinh k\xi$

Table 11.1: Transient responses Δ_k of discrete-time high-gain second-order phase-locked loop.

Unlike the first-order PLL, the error due to a frequency step vanishes as $k \to \infty$ irrespective of $\omega_n^d T$ and ζ^d. For practical values of ζ^d (say $\zeta^d \in [0.5, 2]$), the shape of the responses becomes independent of $\omega_n^d T$ when $\omega_n^d T \ll 1$. In that case

$$\sqrt{1 - 2\zeta^d\omega_n^d T - (\omega_n^d T)^2}^{\,k} \simeq e^{-k\zeta^d\omega_n^d T},$$

	Phase step	Frequency step
$\zeta < 1$	$e^{-\zeta\tau}(\cos\sqrt{1-\zeta^2}\tau - \frac{\zeta}{\sqrt{1-\zeta^2}}\sin\sqrt{1-\zeta^2}\tau)$	$\frac{1}{\omega_n\sqrt{1-\zeta^2}}e^{-\zeta\tau}\sin\sqrt{1-\zeta^2}\tau$
$\zeta = 1$	$e^{-\tau}(1-\tau)$	$\frac{1}{\omega_n}\tau e^{-\tau}$
$\zeta > 1$	$e^{-\zeta\tau}(\cosh\sqrt{\zeta^2-1}\tau - \frac{\zeta}{\sqrt{\zeta^2-1}}\sinh\sqrt{\zeta^2-1}\tau)$	$\frac{1}{\omega_n\sqrt{\zeta^2-1}}e^{-\zeta\tau}\sinh\sqrt{\zeta^2-1}\tau$

Table 11.2: Transient responses $\Delta(t)$ of continuous-time high-gain second-order PLL.

$$\theta \simeq \omega_d^d T, \text{ and } \qquad \xi \simeq \omega_n^d T\sqrt{(\zeta^d)^2 - 1}.$$

Here the loop mimics its continuous-time counterpart, whose transient responses are summarized in Table 11.2 and shown in Figs. 11.20 and 11.21 as a function of the relative time $\tau \triangleq \omega_n t/T$. The parameter ω_d in the table is defined as $\omega_d \triangleq \omega_n\sqrt{1-\zeta^2}$.

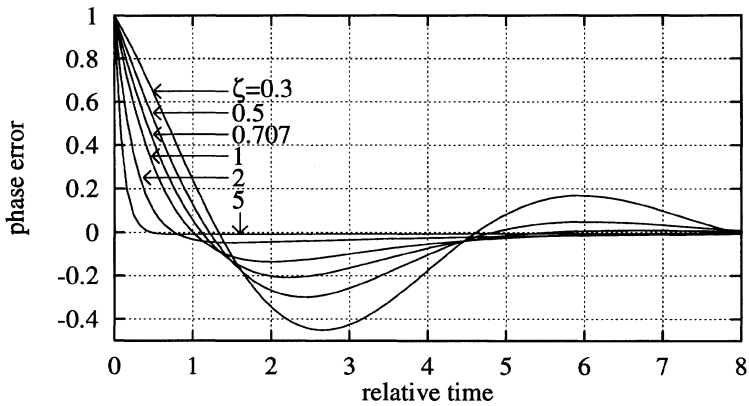


Figure 11.20: Phase error $\Delta(t)$ due to a unit phase step $\phi(t)$ for high-gain second-order continuous-time PLL. Time is normalized in units ω_n. The damping factor ζ is used as a parameter.

The error due to a phase step decays initially by

$$\begin{aligned} -20\log_{10} e^{-\zeta\omega_n}(1-\zeta\omega_n) &= 20(\zeta\omega_n - \ln(1-\zeta\omega_n))/\ln 10 \\ &\simeq 40\zeta\omega_n/\ln 10 \text{ dB per symbol interval } T, \end{aligned}$$

where the approximation holds for $\zeta\omega_n \ll 1$. It is less than 5% of its initial value after approximately $3/(\zeta\omega_n)$ symbol intervals. The error due to a frequency step decays most rapidly for $\zeta \simeq 1/\sqrt{2} \simeq 0.707$. Larger values of ζ yield a slower response to frequency steps but a faster response to phase steps. Values of ζ significantly smaller than 0.707 are not often encountered.

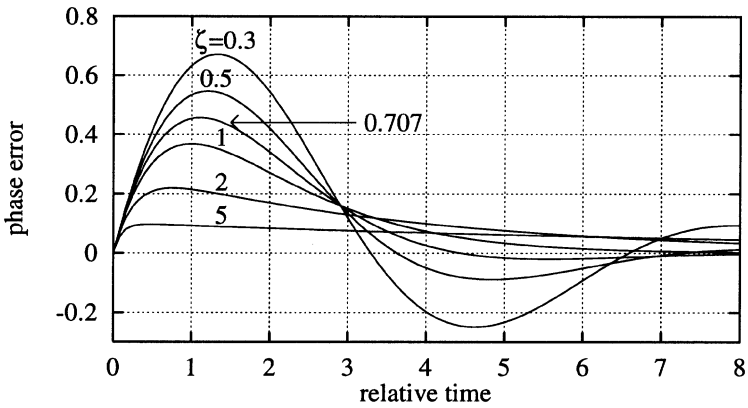


Figure 11.21: Phase error $\Delta(t)$ due to a unit phase ramp (frequency step) $\phi(t)$ for high-gain second-order continuous-time PLL. Time is normalized in units ω_n. The damping factor ζ is used as a parameter.

For small ω_n, the transient responses of the discrete-time PLL are essentially sampled versions of those of Figs. 11.20 and 11.21. To illustrate this, Fig. 11.22 shows PLL responses to a unit phase step according to Table 11.1 for a typical damping factor $\zeta^d = 1/\sqrt{2} \simeq 0.707$ and various values of ω_n^d. Also depicted is the corresponding continuous-time response according to Table 11.2. The discrete-time response is virtually indistinguishable from the continous-time one for $\omega_n^d T < 0.1$. This covers most cases of practical interest.

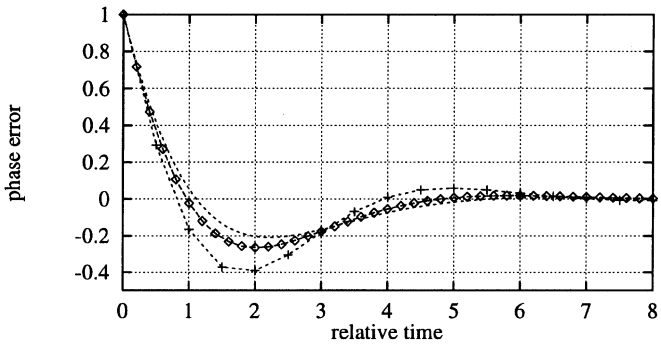


Figure 11.22: Phase error Δ due to a unit phase step ϕ for discrete-time high-gain second-order PLL with damping factor $\zeta^d = 1/\sqrt{2} \simeq 0.707$ and natural frequencies $\omega_n^d T = 0.5$ (crosses) and $\omega_n^d T = 0.2$ (squares). Also depicted is the corresponding continuous-time response (dashed). Time is normalized in units ω_n.

Exercise 11.2 Use a reasoning akin to that of Section 11.3.3 to show for all ζ^d and $\omega_n^d T$ that a frequency offset λ causes no limiting error Δ_∞.

11.4.4 Noise suppression

The high-gain second-order PLL has a normalized equivalent noise bandwidth B_l that differs, in general, from that of the first-order PLL. In all other respects the results of Section 11.3.4 remain applicable. From (11.12), we observe that B_l may be computed as a contour integral

$$B_l = \frac{1}{2\pi j} \oint G_\psi(z) G_\psi(\frac{1}{z}) \frac{dz}{z}. \tag{11.26}$$

The region of convergence includes the unit circle. Within this region, only $G_\psi(z)$ has poles. Upon evaluating their residues one finds after some elementary rearrangements that

$$B_l = \omega_n^d T \frac{2 + 8(\zeta^d)^2 - 6\omega_n^d T \zeta^d + (\omega_n^d T)^2}{(2\zeta^d - \omega_n^d T)(4 - 4\omega_n^d T \zeta^d + (\omega_n^d T)^2)}. \tag{11.27}$$

Just as for the first-order loop, B_l is normalized with respect to the symbol rate and thereby dimensionless. In the absence of any filtering (i.e. when $|G_\psi| = 1$ for all ω) we have $B_l = 1$. For all $\omega_n^d T$ and ζ^d within the PLL stability range it may be shown that

$$B_l \geq \tilde{B}_l \triangleq \omega_n^d T(\zeta^d + \frac{1}{4\zeta^d}). \tag{11.28}$$

The right-hand side is the normalized equivalent noise bandwidth of the continuous-time high-gain second-order PLL [8]. Consequently, the discrete-time PLL suppresses noise less effectively than its continuous-time counterpart for the same loop bandwidth $\omega_n T$ (i.e. for roughly the same tracking properties). The difference vanishes (i.e. $B_l \simeq \tilde{B}_l$) as $\omega_n^d T \ll 1$. This would seem to imply that a discrete-time PLL must be designed to approximate an analog PLL in order to function best.

Exercise 11.3 Show that $B_l \geq \omega_n^d T$. Which damping factors ζ^d are desirable for noise suppression?

11.5 LINEAR ANALYSIS OF GENERAL SECOND-ORDER PLL

We finally allow the loop filter to contain a leaky integrator as in Fig. 11.5.c. Because this case is of less interest than the high-gain loop (which has $\mu = 1$) our discussion will be brief. The LF has transfer function

$$Q(z) = K_p + \frac{K_f}{z - \mu} \tag{11.29}$$

for some $K_p, K_f \in [0, \infty)$ and $\mu \in [0, 1]$. In conjunction with a linearized phase detector we arrive at a loop transfer function

$$G_\psi(z) = 1 - G_\Delta(z) = \frac{K_f^t + K_p^t(z-\mu)}{K_f^t + K_p^t(z-\mu) + (z-1)(z-\mu)} \tag{11.30}$$

where $K_p^t = K_d K_p K_o$ and $K_f^t = K_d K_f K_o$. This can be recast in the standard form

$$G_\psi(z) = \frac{(\omega_n^d T)^2 + (2\zeta^d \omega_n^d T - (1-\mu))(z-1)}{(\omega_n^d T)^2 + 2\zeta^d \omega_n^d T(z-1) + (z-1)^2}, \tag{11.31}$$

where $\omega_n^d T \triangleq \sqrt{K_f^t + K_p^t(1-\mu)}$ and $\zeta^d \triangleq [K_p^t + 1 - \mu]/[2\sqrt{K_f^t + K_p^t(1-\mu)}]$. Conversely, $K_p^t = 2\zeta^d \omega_n^d T - (1-\mu)$ and $K_f^t = (\omega_n^d T)^2 - 2\zeta^d \omega_n^d T(1-\mu) + (1-\mu)^2$. In Appendix 11A it is shown for a passive analog loop filter how the continuous-time filter parameters determine K_p, K_f and μ. Because the numerator of (11.31) coincides with that of the high-gain loop, the stability results of Section 11.4.2 continue to apply.

For small $|\omega T|$ we have $z = e^{j\omega T} \simeq 1 + j\omega T$, so that $(z-1) \simeq j\omega T$. Then

$$G_\psi \simeq \frac{1 + 2j\zeta^d(1 - \frac{1-\mu}{2\zeta^d \omega_n^d T})\frac{\omega}{\omega_n^d}}{1 + 2j\zeta^d \frac{\omega}{\omega_n^d} - (\frac{\omega}{\omega_n^d})^2}. \tag{11.32}$$

The right hand side describes the transfer function of the second-order continuous-time PLL with damping factor $\zeta = \zeta^d$ and natural frequency $\omega_n = \omega_n^d$. With (11.32) it can be shown that this PLL has normalized equivalent noise bandwidth

$$B_l = \omega_n^d[\zeta^d(1 - \frac{(1-\mu)\zeta^d}{2\omega_n^d T})^2 + \frac{1}{4\zeta^d}]. \tag{11.33}$$

11.6 NONLINEAR PLL BEHAVIOR

The linear PD approximation used in Sections 11.3 and 11.4 always has the PLL in lock. Loss of lock and (re)acquisition are governed by nonlinear PD behavior. Fig. 11.23 shows some of the PD transfer characteristics that were encountered in Chapter 9. They are normalized to have the same slope K_d in the origin, i.e. their linear behavior is identical. The sinusoidal characteristic has the smallest linear operating range. The corresponding PLL will loose lock more easily and (re-)acquire lock less easily than for the other characteristics.

11.6.1 Unlock behavior

In a locked loop there is no average frequency error. For each cycle of the input there must be a corresponding cycle of the output. If there are *cycle slips*, the loop is considered to have lost lock, even if only momentarily.

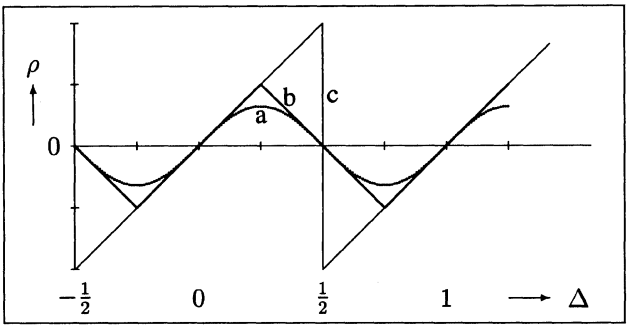


Figure 11.23: Typical phase detector transfer characteristics. a. sinusoidal; b. triangular; c. sawtooth.

The PLL does not necessarily loose lock when phase errors Δ_k grow beyond the linear PD operating range. PD nonlinearity causes harmonics of the phase error. These have a higher frequency and are hence suppressed more effectively than the 'baseband' error because of the low-pass nature of the loop. Once phase errors become too large, however, the PLL will loose lock. Possible causes are phase and frequency changes of the PLL input as well as noise. Phase changes must be sudden and large to cause unlock and will typically quickly cause re-acquisition. The other two causes deserve a more extensive discussion.

Unlock due to frequency changes of the PLL input: Here two notions are of particular interest:

The *Pull-out range* is the minimum frequency step which causes the loop to loose lock. It is roughly equal to 2-5 times the loop bandwidth, depending on the applied phase detector [6].

The *Hold range* (*hold-in range, static operating range*) of the PLL is the range of frequencies over which the loop will remain locked when the input frequency changes very slowly. For the high-gain second-order PLL, this range is mainly determined by the control range of the VCO.

Unlock due to noise: When noise power is significant, noise peaks may exceed the PD limits and cause cycle slips or complete loss of lock. This phenomenon depends strongly on the *loop signal-to-noise ratio*

$$\rho_L \triangleq \frac{1}{2\sigma_\theta^2 B_L}, \tag{11.34}$$

where σ_θ^2 is the variance of the input-referred phase noise θ_k and B_l is the equivalent noise bandwidth of the loop (see Sections 11.3.4 and 11.4.4).[2] Whenever the PD is in

[2]We adhere to common practice in denoting the loop SNR by ρ_L. This symbol should not be confused with the symbol ρ that we have reserved for the PD transfer function.

the linear operating range, we find with (11.11) that $\rho_L = 1/(2\sigma_\psi^2)$. In that case the loop SNR is fully determined by the residual jitter variance σ_ψ^2.

According to Gardner [8, Chapter 3], lock cannot be held below 0 dB loop SNR. At this SNR, the loop is in lock only part of the time, and any additional disturbance will tend to cause complete loss of lock. The average elapsed time τ_{av} between cycle slips appears to grow exponentially with the loop SNR. For a second-order high-gain continuous-time loop with damping factor $\zeta = 0.707$, it was found for low loop SNRs that $\tau_{av}/T \simeq \frac{2}{\omega_n T}\exp\{\pi\rho_L\}$. For example, if $\omega_n T = 0.01$, then cycle slips occur roughly every $4000T$ when $\rho_L = 0$ dB. This is almost certainly unacceptable. Conversely, $\tau_{av} \simeq 9.10^{15}T$ for $\rho_L = 10$ dB, i.e. cycle slips are virtually absent. A reasonable requirement on the cycle slip rate is that it be much smaller than the rate at which bit or symbol errors occur.

11.6.2 Acquisition

Acquisition was discussed at some length in Section 9.4. Here we mention only a few additional aspects.

Effect of frequency offsets In Section 9.4 we have distinguished *frequency pull-in* and *phase lock-in* as the two principal acquisition stages. The *pull-in range*, i.e. the frequency range over which the loop can acquire lock, is usually proportional to $\sqrt{\zeta\omega_n}$ for a second-order continuous-time PLL [6]. The *pull-in time* t_p increases rapidly with the frequency offset λ and is often much larger than the lock time. In [6, p. 62] it is argued that $t_p/T \simeq c\lambda^2/[2\zeta(\omega_n T)^3]$ for a certain category of phase detectors, where the constant c depends on the PD and ranges roughly between 6 and 40. The *lock time* is on the order of $1/(\zeta\omega_n T)$ symbol intervals T.

The pull-in range is typically smaller than the hold-in range. The *false-lock range* is the part of the hold-in range that is outside the pull-in range. In this range the loop can lock temporarily or permanently at an erroneous frequency, usually a harmonic or subharmonic of the desired frequency (*harmonic* or *subharmonic lock*).

Effect of noise According to [8], it is nearly impossible to acquire lock if the loop has 0 dB SNR. If the frequency of the input signal is well known then acquisition at 3 dB appears practicable.

11.7 DESIGN EXAMPLE

We wrap our treatment of timing recovery up with a brief design example that draws freely from the material in Chapters 9-11. The example pertains to the Class IV partial-response system without excess bandwidth that we have encountered earlier in Examples 10.13, 10.14, 9.13 and 9.14. Data a_k is uncorrelated and noise $n(t)$ is white with power spectral density N_0 (Fig. 11.24).

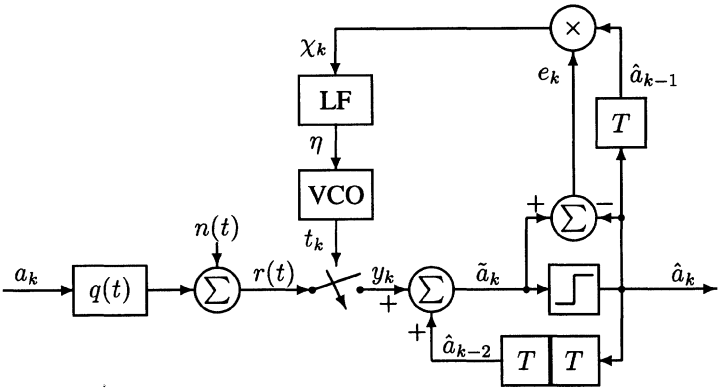


Figure 11.24: Class IV partial-response data-transmission system with data-aided timing-recovery loop.

The receiver comprises the bit-by-bit feedback detector of Section 6.2 (Fig. 6.4). Old decisions $\hat{a}_{k-2}$ are added to r_k. When these are correct, a decision variable $\tilde{a}_k \simeq a_k - 2\Delta a_{k-1} + v_k$ emerges, where v_k denotes sampled noise. Timing recovery is based on the zero-forcing scheme of Example 10.13 (Fig. 10.50). For the system at hand this scheme has efficiency $\gamma = 2$ (see Example 10.14). According to (9.6), this will yield a mean-square jitter variance

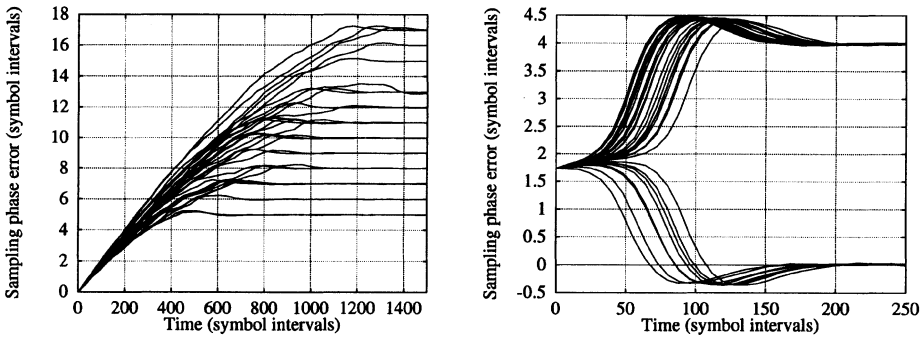
$$\sigma_\psi^2 = \frac{K_t}{2\gamma \text{SNR}} = \frac{K_t}{4\text{SNR}} \tag{11.35}$$

for a first-order PLL with total open-loop gain K_t. To restrict RMS jitter σ_Δ to 1% of a symbol interval T at SNR = 20 dB we must choose $K_t = 4.10^{-2}$. This is the choice that underlies the decision-directed acquisition characteristics of Fig. 9.15. Acquisition can be improved by basing it on a known preamble so as to avoid decision errors. Of particular interest for Class IV is the $4T$ preamble '$\ldots + + - - + + - - \ldots$'. An advantage of this pattern is that γ increases to 4 (see Problems 10.45 and 10.46), i.e. that K_t can be twice as large for the same steady-state RMS jitter σ_Δ. Resulting phase acquisition characteristics were shown in Fig. 9.16.

In practice the system of Fig. 11.24 will normally be dimensioned as a high-gain second-order PLL. This requires a proportional-plus-integral loop filter (see Fig. 11.5.b) whose output η_k drives the VCO. This VCO has a *negative* sensitivity K_o so as to compensate for the negative PD gain $K_d = -2$ (see Fig. 10.51.a). After lumping all gain factors into total open-loop gains K_p^t and K_f^t we arrive at the phase-domain PLL model of Fig. 11.14. It remains to determine K_p^t and K_f^t or, equivalently, the damping factor ζ^d and the normalized natural frequency $\omega_n^d T$ (see (11.23)). To this end we combine (9.3), (11.14) and (11.28) to find that we now have steady-state jitter with variance

$$\sigma_\psi^2 \simeq P_\theta(1) B_l = \frac{B_l}{\gamma \text{SNR}} \simeq \frac{1}{\gamma \text{SNR}} \omega_n^d T (\zeta^d + \frac{1}{4\zeta^d}). \tag{11.36}$$

We fix the damping factor at the common value $\zeta^d = 1/\sqrt{2}$. In order to achieve $\sigma_\Delta = 10^{-2}$ at SNR = 20 dB we must use a normalized natural frequency $\omega_n^d T \simeq 0.02$ for $\gamma = 2$ (uncorrelated binary data), and $\omega_n^d T \simeq 0.04$ for $\gamma = 4$ (binary $4T$ pattern). Corresponding acquisition characteristics are depicted in Fig. 11.25.



Uncorrelated data, decision-directed

Binary $4T$ pattern, actual data symbols used instead of decisions

Figure 11.25: Phase acquisition curves for the timing recovery scheme of Fig. 11.24 with an initial frequency error of 2% and a worst-case initial phase. The channel has SNR = 20 dB.

For the decision-directed scheme, the direct-lock range is only about $\pm 0.5\%$ and for this reason many cycle slips occur in Fig. 11.25 before the loop acquires lock. For the $4T$ pattern, on the other hand, the loop locks up directly because the direct-lock range of about $\pm 10\%$ easily covers the initial offset of 2%. Due to both cycle slips and lower efficiency, acquisition takes some 3 to 7 times longer for uncorrelated data than for the $4T$ pattern. Even for the $4T$ pattern acquisition is clearly hampered by hangup. This problem can be solved for the timing-recovery scheme at hand by the technique of [17].

11.8 FURTHER READING

The history of the phase-lock principle is long and interesting. According to Lindsey [14], the phenomenon of phase-lock was first observed in 1665 by Christiaan Huygens, who noted that two adjacent pendulum clocks tended to synchronize as a result of coupling via a wall [11]. Mathematical foundations were laid in the twenties and thirties by Van der Pol [19] and by a number of Russian scientists (see [14, sec. 3-12]). The first electronic implementation of a PLL predates 1923 [1].

Because of the introductory scope of this chapter, various topics have been underexposed or have not been touched upon at all. The instrumentation of PLL components, for example, has not been addressed, even though this is an important practical issue. Similarly, our coverage of nonlinear PLL behavior [20] is probably not at par with its

practical significance. Further omissions include loop threshold effects [18, Chapter 6], the effect of limiters within the loop [8, Chapter 5], and analysis tools such as phase-plane techniques [20] and Fokker-Planck operators [18]. The literature on analog and digital PLLs amply covers these and other subjects. As points of entry to this wealth of information we mention [3], [4], [8], [10], [14], [15], [20], [16], [6] and [18].

APPENDIX 11A

Equivalent discrete-time model of cascade of analog loop filter and VCO

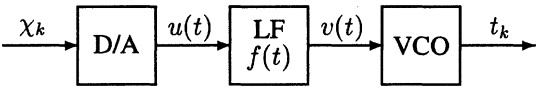


Figure 11A.1: Analog LF/VCO cascade.

In Fig. 11A.1, the PD output χ_k is converted into an analog signal $u(t)$ that drives a continuous-time LF/VCO cascade. The VCO sampling instants $t_k = (k + \psi_k)T$ have a phase ψ_k that depends via the LF on χ_k. In this appendix we show for 3 popular LF types that this dependence can be modeled as in Fig. 11.4. We first consider the D/A-converter and VCO.

D/A-converter Since PD outputs χ_k are produced after every sampling instant t_k, D/A-conversion must occur in synchronism with the VCO clock. Let $T_k \triangleq t_k - t_{k-1}$ be the duration of the kth clock interval, and denote by $c_k(t)$ the hold function of length T_k, i.e. $c_k(t) = 1$ for $t \in [0, T_k]$ and 0 else. Then the D/A output $u(t)$ may be described as

$$u(t) = \sum_{k=-\infty}^{\infty} \chi_k c_k(t - t_k). \tag{11A.1}$$

When the PLL is in lock, T_k will approach the actual symbol interval T. In that case the D/A can be considered to have a time-independent hold function $c(t) \triangleq 1$ for $t \in [0, T]$ and 0 else.

VCO: Denote by λ the normalized offset between the free-running VCO frequency and the symbol rate $1/T$. The actual VCO frequency $1/T_k$ is shifted with respect to the free-running frequency $(1 - \lambda)/T$ in proportion to the VCO input $v(t)$. For simplicity it is assumed that the VCO responds instantaneously, such that the duration of the kth sampling interval depends on the VCO control voltage at the beginning of that interval.[3]

[3]The results do not change significantly when the VCO responds more diffusely.

Then

$$\frac{1}{T_{k+1}} = \frac{1-\lambda}{T} - K_o^a v(t_k), \tag{11A.2}$$

where K_o^a represents the sensitivity of the VCO (with dimension Hz/V). Now $T_{k+1} = t_{k+1} - t_k = (1+\psi_{k+1}-\psi_k)T$ for all k. Typically $|\psi_{k+1}-\psi_k| \ll 1$. Then $T/T_{k+1} \simeq 1 - \psi_{k+1}+\psi_k$, whence $\psi_{k+1} \simeq \psi_k+\lambda+K_o v_k$, where $K_o \triangleq TK_o^a$, while $v_k \triangleq v(t_k)$. This is the phase-domain relation that governs VCO behavior (compare Fig. 11.4). In practice the VCO may exhibit phase noise that perturbs the sampling phases ψ_k randomly. This noise is accounted for in Fig. 11.4, but is of no concern to the rest of this appendix.

From this point on we must specify the loop filter. We consider three cases.

11A.1 Degenerate filter

Here the analog filter degenerates into a constant gain factor K_p. With (11A.1) this means that $v_k = v(t_k) = K_p\chi_k$ (compare Fig. 11.5.a).

11A.2 Active filter

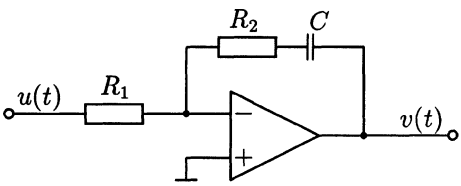


Figure 11A.2: Active first-order analog loop filter.

In Fig. 11A.2 we show a first-order analog loop filter with two resistors R_1 and R_2 and a capacitor C. The filter has transfer function $-(s\tau_2+1)/s\tau_1$, where s is the Laplace operator, $\tau_1 \triangleq R_1C$, and $\tau_2 \triangleq R_2C$. In response to the unit step function $u_s(t)$, it produces an output $v_s(t) = -[\frac{\tau_2}{\tau_1} + \frac{t}{\tau_1}]u_s(t)$. The response $v_h(t)$ to the hold function $u_h(t) = c(t) = u_s(t) - u_s(t-T)$ is

$$v_h(t) = v_s(t) - v_s(t-T) = -\frac{\tau_2}{\tau_1}c(t) - \frac{t}{\tau_1}u_s(t) + \frac{t-T}{\tau_1}u_s(t-T). \tag{11A.3}$$

The sampled version $q_k \triangleq v_h(kT+\varepsilon)$ (with $0 < \varepsilon \ll T$) of $v_h(t)$ is the impulse response of the equivalent discrete-time loop filter. It satisfies $q_k = 0$ for $k < 0$, $q_0 = -\tau_2/\tau_1$, and $q_k = -T/\tau_1$ for all $k \geq 1$. To compensate for the minus signs we can use a VCO with a *negative* VCO sensitivity $-K_o$. Then $q_k = K_p\delta_k + K_f u_{k-1}^s$ where $K_p \triangleq \tau_2/\tau_1$, $K_f \triangleq T/\tau_1$, and u_k^s is the discrete-time unit step function. This corresponds to the loop filter of Fig. 11.5.b. From the definitions of K_p and K_f we find that $\tau_2/T =$

K_p/K_f and $\tau_1/T = 1/K_f$. Hence τ_1 should be selected to achieve a desired value K_f. Then τ_2 fully determines K_p.

It should be noted that the filter of Fig. 11A.2 has infinite bandwidth. Problem 11.15 assesses the effect of a bandwidth limitation.

11A.3 Passive filter

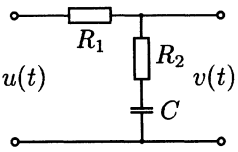


Figure 11A.3: Passive first-order analog loop filter.

Fig. 11A.3 depicts a first-order analog loop filter with two resistors R_1 and R_2 and a capacitor C. The filter has transfer function $(s\tau_2+1)/(s\tau+1)$ where s is the Laplace operator, $\tau_2 \triangleq R_2C$ and $\tau \triangleq (R_1+R_2)C$. In response to the unit step function $u_s(t)$, it produces an output $v_s(t) = [1-\rho\exp(-t/\tau)]u_s(t)$ where $\rho \triangleq R_1/(R_1+R_2) \in [0,1]$. The response $v_h(t)$ to the hold function $c(t) = u_s(t) - u_s(t-T)$ is

$$v_h(t) = v_s(t) - v_s(t-T) = c(t) + \rho[e^{-(t-T)/\tau}u_s(t-T) - e^{-t/\tau}u_s(t)]. \quad (11A.4)$$

The sampled version $q_k \triangleq v_h(kT+\varepsilon)$ (with $0 < \varepsilon \ll T$) satisfies $q_k = 0$ for $k < 0$, $q_0 = 1-\rho$, and $q_k = \rho(1-\mu)\mu^{k-1}$ for $k \geq 1$, where $\mu \triangleq e^{-T/\tau} \in [0,1]$. Hence $q_k = K_p\delta_k + K_f\mu^{k-1}u_{k-1}$ where $K_p \triangleq 1-\rho$, $K_f \triangleq \rho(1-\mu)$, and u_k^s is the discrete-time unit step function. This is just the impulse response of the discrete-time LF of Fig. 11.5.c.

Exercise 11.4 Let $K_p^t \triangleq K_dK_pK_o$ and $K_f^t \triangleq K_dK_fK_o$, where K_d and K_o represent the PD and VCO gain, respectively. Show that K_o and ρ should be selected to achieve a specified pair of total loop gians (K_p^t, K_f^t) according to $K_o = (K_p^t/K_d) + K_f^t/[K_d(1-\mu)]$ and $\rho = K_f^t/[K_f^t + K_p^t(1-\mu)]$.

Example 11.5 In Section 11.5 we have seen that K_p^t and K_f^t depend on ω_n^d, ζ^d and μ as $K_p = 2\zeta^d\omega_n^dT - (1-\mu)$ and $K_f^t = (\omega_n^dT)^2 + 2\zeta^d\omega_n^dT(1-\mu) + (1-\mu)^2$. It is of interest to know what values of ω_n^dT and ζ^d are attainable for a given value of μ. Since K_p and K_f are both nonnegative, K_p^t and K_f^t must have the same sign. One easily verifies that K_f^t is positive definite when $\zeta^d < 1$. In that case $K_p = 2\zeta^d\omega_n^dT - (1-\mu)$ must also be positive, i.e. we must take $\omega_n^dT \leq (1-\mu)/(2\zeta^d)$ for $\zeta^d \leq 1$. For $\zeta^d > 1$, $K_f^t < 0$ for $1-\mu \in [\omega_n^dT(\zeta^d - \sqrt{(\zeta^d)^2-1}), \omega_n^dT(\zeta^d + \sqrt{(\zeta^d)^2-1})]$, while $K_p^t < 0$ for $1-\mu > 2\zeta^d\omega_n^dT$ (Fig. 11A.4).

There are two subregions where K_p^t and K_f^t have the same sign. Correspondingly, attainable values of ω_n^d for $\zeta^d > 1$ are bounded by

$$\omega_n^dT \in [\frac{1-\mu}{2\zeta^d}, \frac{1-\mu}{\zeta^d + \sqrt{(\zeta^d)^2-1}}] \text{ or } \omega_n^dT > \frac{1-\mu}{\zeta^d - \sqrt{(\zeta^d)^2-1}}. \quad (11A.5)$$

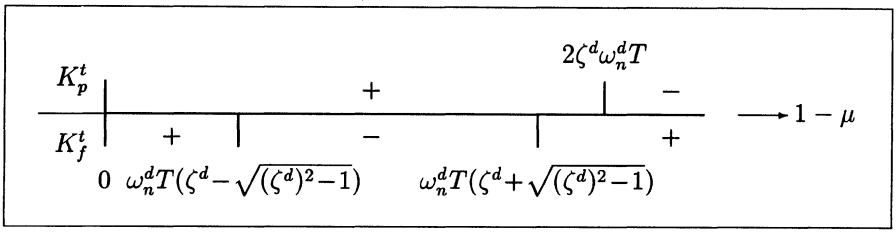


Figure 11A.4: Signs of K_p^t and K_f^t for $\zeta^d > 1$ as a function of $1-\mu$.

Exercise 11.5 Let $\mu = 0.5$. Which part of the PLL stability region of Fig. 11.18 is attainable?

Exercise 11.6 Let $\omega_n^dT = 0.01$ and $\zeta^d = 1$. Which values of μ are compatible with this choice?

APPENDIX 11B

Effect of loop delay on stability

First-order PLL The values of K_t for which the numerator $F(z) = K_t + z^M(z-1)$ has roots on the unit circle may be determined by setting $z = e^{j\theta}$. The real and imaginary parts $\cos(M+1)\theta - \cos M\theta + K_t$ and $\sin(M+1)\theta - \sin M\theta$ of $F(e^{j\theta})$ are both required to be zero. For the imaginary part to be zero we must have $\theta = 0$ or $\theta = (2m+1)\pi/(2M+1)$ with $m \in \{-M-1, \ldots, M\}$. For these values of θ the real part of $F(e^{j\theta})$ is zero when $K_t = 0$ or

$$K_t = 2(-1)^m \sin\frac{(2m+1)\pi}{2(2M+1)}. \tag{11B.1}$$

For $K_t = 0$, M roots of $F(z)$ lie at the origin ($z = 0$) and one root lies at $z = 1$. As K_t is increased, the root at $z = 1$ and one of the other roots move toward each other on the real axis to meet at $z = M/(M+1)$. The other roots move radially from the origin. For larger values of K_t, the two real axis roots split to form a complex conjugate pair which crosses the unit circle at $\theta = \pm\pi/(2M+1)$. This corresponds to $m = 0$ and $m = -1$ in (11B.1). Thus, for all roots to lie inside the unit circle, K_t must satisfy (11.5).

Second-order PLL Here stability is governed by the roots of the function

$$F(z) = (\omega_n^dT)^2 + 2\zeta^d\omega_n^dT(z-1) + z^M(z-1)^2 = K_f^t + K_p^t(z-1) + z^M(z-1)^2. \tag{11B.2}$$

We consider the cases $M = 0$ and $M \geq 1$ separately.

Zero loop delay ($M = 0$): Here $F(e^{j\theta}) = K_f^t - K_p^t + 1 + (K_p^t - 2)e^{j\theta} + e^{j2\theta}$. The real and imaginary parts of F are zero when

$$(K_f^t - K_p^t + 1) + (K_p^t - 2)\cos\theta + \cos 2\theta = 0 \tag{11B.3}$$

and $(K_p^t - 2)\sin\theta + \sin 2\theta = 0$, respectively. The latter equation holds in three different cases:

1. $\theta = 2n\pi$: Here $\cos\theta = \cos 2\theta = 1$. With (11B.3) this means that $K_f^t = 0$.

2. $\theta = (2n+1)\pi$: Here $\cos\theta = -1$ and $\cos 2\theta = 1$. With (11B.3) this means that $K_f^t + 4 = 2K_p^t$.

3. $2 - K_p^t = \frac{\sin 2\theta}{\sin\theta} = 2\cos\theta$: Here $\cos\theta = 1 - K_p^t/2$ and $\cos 2\theta = 2\cos^2\theta - 1 = 2(1 - K_p^t/2)^2 - 1$. With (11B.3) this means that $K_f^t = K_p^t$.

The region enclosed by the lines $K_f^t = 0$, $K_f^t + 4 = 2K_p^t$ and $K_f^t = K_p^t$ is the stability region. In terms of ω_n and ζ the stability region is demarcated by $\omega_n^d T \in [0,2] \cap \zeta^d \in [\frac{\omega_n^d T}{2}, \frac{\omega_n^d T}{4} + \frac{1}{\omega_n^d T}]$.

Nonzero loop delay ($M \geq 1$): Here $F(e^{j\theta}) = K_f^t + 2jK_p^t e^{j\theta/2}\sin\frac{\theta}{2} - 4e^{j(M+1)\theta}\sin^2\frac{\theta}{2}$. The real and imaginary parts of $F(e^{j\theta})$ are zero when

$$K_f^t - 2K_p^t \sin^2\frac{\theta}{2} - 4\cos(M+1)\theta\sin^2\frac{\theta}{2} = 0 \tag{11B.4}$$

and $2K_p^t\cos\frac{\theta}{2}\sin\frac{\theta}{2} - 4\sin(M+1)\theta\sin^2\frac{\theta}{2} = 0$, respectively. The latter equation holds in two different cases:

1. $\theta = 2n\pi$: Here $\sin\frac{\theta}{2} = 0$. With (11B.4) this means that $K_f^t = 0$.

2. $2K_p^t\cos\frac{\theta}{2} = 4\sin(M+1)\theta\sin\frac{\theta}{2}$. Here

$$K_p^t(\theta) = 2\sin(M+1)\theta\tan\frac{\theta}{2}, \tag{11B.5}$$

and upon using this fact in (11B.4) we find after some elementary manipulations that

$$K_f^t(\theta) = 4\sin\frac{\theta}{2}\tan\frac{\theta}{2}\cos(M+\frac{1}{2})\theta. \tag{11B.6}$$

It can be noted that $K_f^t = 0$ when $\theta = 2n\pi$ or $(M+0.5)\theta = (n+0.5)\pi$ [i.e. $\theta = (2n+1)\pi/(2M+1)$]. Throughout the interval $\theta \in [0, \pi/(2M+1)]$, $K_f^t(\theta)$ is nonnegative. Here $K_p^t(\theta)$ increases monotonically from 0 for $\theta = 0$ to $2\sin(\pi/(4M+2))$ for $\theta = \pi/(2M+1)$.

For $M \geq 1$ the stability region is enclosed by the line $K_f^t = 0$ and the curve $(K_f^t(\theta), K_p^t(\theta))$ for $\theta \in [0, \pi/(2M+1)]$. In terms of $\omega_n^d T$ and ζ^d it is enclosed by the line $\omega_n^d T = 0$ and the curve

$$(\omega_n^d T, \zeta^d) = (2\sin\frac{\theta}{2}\sqrt{\frac{\cos(M+\frac{1}{2})\theta}{\cos\frac{\theta}{2}}}, \frac{1}{\sqrt{2}}\frac{\sin(M+1)\theta}{\sqrt{\cos(M+1)\theta + \cos M\theta}}). \tag{11B.7}$$

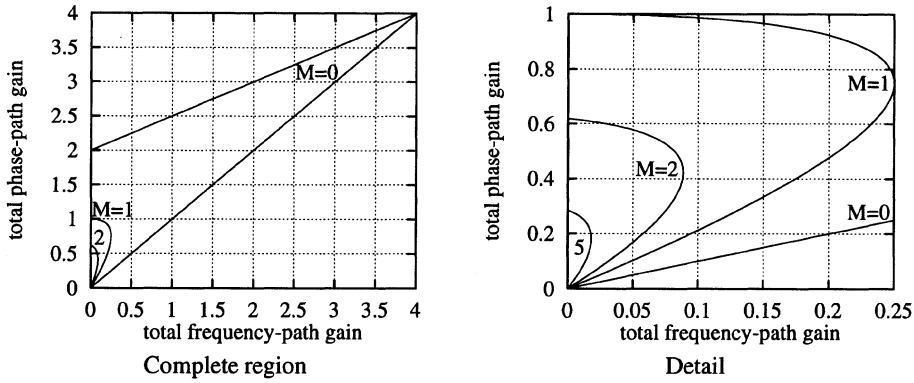


Figure 11B.1: Outer edges of stability range of second-order discrete-time PLL for various loop delays M as a function of total phase-path gain K_p^t and frequency-path gain K_f^t.

Stability regions are depicted in Fig. 11B.1 as a function of K_p^t and K_f^t, and in Fig. 11.18 as a function of $\omega_n^d T$ and ζ^d. In [2] these regions are compared with those of the continuous-time high-gain second-order PLL, for which an analytic expression was derived by Lindsey [14, sec. 4-3.3].

PROBLEMS

11.1 Consider a loop filter with order $N \geq 0$. Show that a PLL with this filter has order $N+1$.

11.2 Consider the first-order discrete-time PLL of Section 11.3. Assume that a delay of M symbol intervals is introduced into the loop. Show that $G_\psi(e^{j2\pi\Omega}) = K_t/[A(e^{j\omega T})\exp\{j\alpha(e^{j\omega T})\}+K_t]$ and $|G_\psi(e^{j\omega T})|^2 = K_t^2/[K_t^2+2K_tA(e^{j\omega T})\cos\alpha(e^{j\omega T})+A^2(e^{j\omega T})]$ where $A(e^{j\omega T}) \triangleq 2\sin\omega T/2$ and $\alpha(e^{j\omega T}) \triangleq (\frac{1}{2}+(2M+1)\omega T/2)$. Argue that these functions closely approach their counterparts for $M = 0$ whenever the PLL is well within the stability range as given by (11.5).

11.3 Prove (11.7).

11.4 Consider the first-order discrete-time PLL of Section 11.3 (Fig. 11.6). Let φ_k be given by (11.8). Show that the corresponding error response is

$$\Delta_k = \begin{cases} 0 & \text{if } k \leq 0, \text{ and} \\ \frac{\lambda}{K_t}(1-(1-K_t)^k) & \text{if } k \geq 1. \end{cases} \tag{11P.1}$$

Sketch this response for $K_t = 0.1$ and $K_t \to 1$. Verify that the limiting error coincides with (11.9).

11.5 Consider a discrete-time first-order PLL with open-loop gain K_t^d and a continuous-time one with gain K_t^c, and assume that both are excited by a unit phase step at moment 0. How should K_t^d be chosen as a function of K_t^c for the error responses Δ_k and $\Delta(kT)$ of both loops to decay equally fast? Which loop has the largest noise bandwidth B_l in this case?

11.6 Transmitters and receivers are sometimes operated back-to-back in a chain of transmission units or repeaters. In such a chain, jitter can accumulate to unacceptable proportions if the PLL transfer function $\tilde{G}_\psi(\omega)$ for a single transmission unit exhibits significant *peaking*, i.e. if the largest transfer magnitude $|\tilde{G}_\psi|$ exceeds 1 significantly [5, 21]. For the continuous-time second-order high-gain PLL, $|\tilde{G}_\psi|$ assumes its maximum when ω is in the vicinity of the natural frequency ω_n. Determine $|G_\psi|$ for $\omega = \omega_n$ and comment on the severeness of peaking. What damping factors ζ are preferable?

11.7 Determine the transfer magnitude $|G_\psi(e^{j\omega T})|$ of the second-order high-gain discrete-time PLL at the natural frequency $\omega = \omega_n^d$. Compare the result with that of Problem 11.6. For which PLL is peaking smallest?

11.8 Consider a system in which the phase ϕ_k of the received signal has significant spectral content up to 1% of the signaling rate $1/T$. For a second-order high-gain PLL with damping factor $\zeta^d = 1$, what normalized natural frequency $\omega_n^d T$ must be chosen to ensure adequate tracking? Determine the largest PD input noise variance σ_θ^2 that keeps the RMS magnitude of noise-induced jitter below 5% of a symbol interval T.

11.9 For some timing-recovery techniques (such as those of Sections 10.7 and 10.6), the PD gain K_d depends linearly on the gain of the channel. In digital recording systems, this gain may fluctuate severely as a result of dropouts. Let K_p^t and K_f^t be such that $\omega_n^d T = 0.01$ and $\zeta^d = 1$ in the absence of dropouts. How do $\omega_n^d T$ and ζ^d change if a dropout reduces the channel gain to 10% of its nominal value? How does this affect the properties of the loop? Assume that the loop delay M is such that the loop is stable for the nominal gain of the channel. Can the loop ever become unstable as a result of dropouts?

11.10 Let ϕ_k have power spectral density

$$P_\phi(e^{j\omega T}) = \begin{cases} \frac{1}{\omega_0}\sigma_\phi^2 & \text{if } |\omega| < \omega_0, \text{ and} \\ 0 & \text{if } \omega \geq \omega_0. \end{cases} \tag{11P.2}$$

This ideal low-pass function has one-sided bandwidth ω_0. Furthermore, ϕ_k has variance

$$\frac{T}{2\pi}\int_{-\pi/T}^{\pi/T} P_\phi(e^{j\omega T})d\omega = \sigma_\phi^2. \tag{11P.3}$$

Verify that the resulting residual jitter Δ_k for a first-order discrete-time PLL has variance

$$\sigma_\Delta^2 = \frac{\sigma_\phi^2}{1-K_t}[1 - \frac{2K_t}{2\pi\Omega_0(2-K_t)}\tan^{-1}(\frac{2-K_t}{K_t}\tan\frac{\omega_0 T}{2})]. \tag{11P.4}$$

How does this expression simplify for typical values $\omega_0 T \ll \pi$ and $K_t \ll 1$? Approximate the result for $\omega_0 T \gg K_t$, $\omega_0 T \simeq K_t$, and $\omega_0 T \ll K_t$.

11.11 In this problem we compare noise and tracking performance of the continuous-time first-order and high-gain second-order PLLs.

1. Assume that there is an initial phase error and no initial frequency error. How must the loop gain K_t of the first-order PLL be chosen in relation to $\omega_n^d T$ and ζ^d in order to achieve the same initial rate of convergence as for the second-order PLL? How much longer does it take for the second-order PLL for the phase transient to settle down?
2. Compare the equivalent noise bandwidths of both PLLs for the same initial rate of convergence. Which PLL is favorable?
3. Answer the same question for the situation that the both PLLs are dimensioned for phase transients to settle down equally fast.

11.12 Assume that the sampling instants t_k that are demarcated by the VCO exhibit frequency noise. For this type of noise we have sampling periods $T_k = (1 + d_k)T$ where T denotes the nominal period and d_k is an uncorrelated random sequence with zero mean and variance λ^2. The sampling instants t_k are determined recursively according to $t_k = t_{k-1} + T_k$ for all k. In non-recursive terms we may write $t_k = (k + \nu_k)T$ where ν_k is the VCO phase noise in symbol intervals T.

1. Show that ν_k is related to d_k via a linear filtering operation. What is the transfer function of this filter?
2. Show that the power spectral density $P_\nu(e^{j\omega T})$ of ν_k is given by (11.17).

11.13 Consider the general second-order PLL of Section 11.5. Which value of μ minimizes the normalized equivalent noise-bandwidth B_l of (11.33)? Is this value always attainable with a passive analog loop filter?

11.14 Consider the general second-order PLL of Section 11.5. Assume a static frequency offset λ between the free-running frequency of the VCO and the PD input ϕ. Use a reasoning akin to that of Section 11.3.3 to show that the loop stress Δ_∞ amounts to

$$\Delta_\infty = \lambda \frac{1-\mu}{K_f^t + K_p^t(1-\mu)} = \lambda \frac{1-\mu}{(\omega_n^d T)^2}. \tag{11P.5}$$

What value of μ is to be preferred? What type of PLL does this correspond to?

11.15 The active analog loop filter of Appendix 11A.2 is impractical in that it has infinite bandwidth. A more practical filter transfer function is $W(p) = 1/[p(1+ap)(1+bp)]$ where $p \triangleq sT$ and $a \geq b \geq 0$ remain to be determined. This filter has two cutoff frequencies $\omega_a \triangleq 1/(aT)$ and $\omega_b \triangleq 1/(bT)$. Below ω_a it behaves as an integrator. Between ω_a and ω_b, it has a more or less constant transfer magnitude $|W| \simeq a$, and above ω_b it exhibits first-order rol-off. The active analog loop filter of Appendix 11A.2 emerges for $b = 0$, i.e. for $\omega_b = \infty$.

1. Determine the transfer function $\tilde{G}_\psi(p)$ of a continuous-time PLL with this loop filter. What is the order of the loop?
2. The effect of b being nonzero is that two poles of the second-order PLL transfer function (which determine natural frequency and damping factor) are shifted somewhat, and that a spurious third pole arises. To make this effect explicit, we write the denominator $D(p)$ of $\tilde{G}_\psi(p)$ as $D = (\alpha + bp)((\omega_n')^2 + 2\zeta'\omega_n' p + p^2)$ where ω_n' and ζ' are the modified natural frequency and damping factor, and α and b determine the spurious pole. Express α, K and a in terms of b, ζ' and ω_n'. When is the loop stable? What condition must be imposed upon a/b in order for the PLL to exibit second-order high-gain behavior? What does this mean for the cutoff frequency f_b? Is a large damping factor desirable?

11.16 Consider the passive loop filter of Fig. 11A.3. Express the cut-off frequency $f_c = 1/(2\pi\tau_2)$ in terms of ω_n^d, T and ζ^d. How large is f_c for $\omega_n^d T = 0.05$, $\zeta^d = 1$ and $1/T = 20$ MBd?

11.17 Consider the parametric expressions (11B.5) and (11B.6) for the outer edge of the stability region of the second-order discrete-time PLL. Use these expressions to write K_f^t explicitly as a function of K_p^t for a loop delay of one symbol interval ($M = 1$).

REFERENCES

[1] E.V. Appleton, 'Automatic Synchronization of Triode Oscillators', Proc. Cambridge Phil. Soc., Vol. 21, p. 231, 1922-1923.

[2] J.W.M. Bergmans, 'Effect of Loop Delay on Stability of Discrete-Time PLL', *IEEE Trans. Circuits Syst. - I*, Vol. CAS-42, No. 4, pp. 229-231, Apr. 1995.

[3] R.E. Best, *Phase-Locked Loops*. New York: McGraw-Hill, 1984.

[4] A. Blanchard, *Phase-Locked Loops: Applications to Coherent Receiver Design*. New York: John Wiley and Sons, 1976.

[5] C.J. Byrne, B.J. Karafin, and D.B. Robinson, Jr., 'Systematic Jitter in a Chain of Digital Repeaters', *Bell Syst. Tech. J.*, Vol. 42, pp. 2679-2714, Nov. 1963.

[6] R.C. Den Dulk, *An Approach to Systematic Phase-Lock Loop Design*, Ph.D. Thesis, Delft University of Technology, The Netherlands, Nov. 1989.

[7] A.W.M. van den Enden and N.A.M. Verhoeckx, *Discrete-Time Signal Processing - An Introduction*. Hertfordshire, U.K.: Prentice-Hall (UK) Ltd, 1989.

[8] F.M. Gardner, *Phaselock Techniques*. New York: Wiley, 1966 (2nd ed. 1979).

[9] S.W. Golomb, J.R. Davey, I.S. Reed, H.L. Van Trees and J.J. Stiffler, 'Synchronization', *IEEE Trans. Communications Systems*, Vol. CS-11, pp. 481-491, Dec. 1963.

[10] S.C. Gupta, *Phase-Locked Loops*, *Proc. IEEE*, Vol. 63, No. 2, pp. 291-306, Feb. 1975.

[11] Christiaan Huygens, *Horologium Oscillatorum*, 1673.

[12] E.A. Lee and D.G. Messerschmitt, *Digital Communication*. Boston: Kluwer Academic Publishers, 1988.

[13] W.C. Lindsey and M.K. Simon (eds.), *Phase-Locked Loops and their Applications*. New York: IEEE Press, 1978.

[14] W.C. Lindsey, *Synchronization Systems in Communication and Control*. Englewood Cliffs: Prentice-Hall, 1972.

[15] W.C. Lindsey and M.K. Simon (Eds.), *Phase-Locked Loops and Their Applications*. New York: IEEE Press, 1978.

[16] W.C. Lindsey and C.M. Chie, 'A Survey of Digital Phase-Locked Loops', *Proc. IEEE*, Vol. 69, No. 4, pp. 410-431, Apr. 1981.

[17] J.-Y. Lin and C.-H. Wei, 'Fast Timing Recovery Scheme for Class IV Partial Response Channels', *Electron. Lett.*, Vol. 31, No. 3, pp. 159-161, Feb. 2, 1995.

[18] H. Meyr and G. Ascheid, *Synchronization in Digital Communications - Volume I*. New York: Wiley, 1990.

[19] B. van der Pol, 'Forced Oscillations in a Circuit with Nonlinear Resistance', *Phil.Mag.*, and *Ire. of Science*, 7th Series, Vol. 3, 1923.

[20] A.J. Viterbi, *Principles of Coherent Communications*. New York: McGraw-Hill, 1966.

[21] L.E. Zegers, 'The Reduction of Systematic Jitter in a Transmission Chain with Digital Regenerators', *IEEE Trans. Commun.*, Vol. COM-15, pp. 542-551, Aug. 1967.

Index

2B1Q, 108, 172
4B-3T, 207
4B/3T, 172, 207

(1,7), 181
(2,7), 180

AC-coupling, 158
Acquisition, 465
Adaptation schemes
    asynchronous, 398
    blind, 398
    data-aided, 376
    gear-shift techniques, 429
    leaky LMS, 396
    minimum ISI power (MMSI), 404
    minimum mean-square error (MMSE), 414
    non-data-aided, 376, 398
    open-loop, 399
    orthogonalized, 407, 415
    partial zero-forcing (PZF), 403
    self-orthogonalizing, 430
    self-recovering, 398
    sign algorithm, 419
    use of prior knowledge in, 373
    zero-forcing (ZF), 400, 404
Adaptive branch metric computation, 328
Adaptive DC-compensation, 425
Adaptive detection, 376, 421–427
Adaptive equalization, 375, 400–421
Adaptive filters
    structures of, 380
Adaptive receivers, 373–450
    adaptive detection, 421
    adaptive equalization, 400, 412
    automatic gain control, 382
    implementation aspects, 430
    structures of, 375
ADSL, 48, 54
AGC, *see* Automatic gain control
AGWN channel, 117
All-digital timing recovery, 457, 476
Alternate Mark Inversion (AMI), 198
Analog filters, 382
    adaptive, 410
Anti-whistle code, 171
Arithmetic average, 29
Asymmetry of recorded domains, 80
Asynchronous adaptation, 398
ATM, 49
Autocorrelation function, 17, 18
Automatic equalization, 373
Automatic gain control, 382–398, 411
    closed-loop, 383
    data-aided, 386
    effect of DC offset, 395
    effect of integrator leakage, 396
    effect of loop delay, 397
    exponential, 384
    minimum mean-square error, 390
    non-data-aided, 382
    open-loop, 382
    overload of, 397
    startup behavior, 397
    zero-forcing, 386
Azimuth, 72

Balanced modulation code, *see* DC-free modulation code
Balanced quadricorrelator, 486
Barkhausen noise, 74
Baseband data transmission, 47, 105
Baseline wander, 158, 211, 425
    effect of modulation code, 161
Baud rate, 107
Bayesian detection, 143
Bi-mode modulation codes, 204
Bi-phase, 107, 120, 197, 201
    timing recovery for, 581

Bias
effect on receiver performance, 257
of MMSE AGC, 391
of MMSE LE, 244
of MMSE PRE, 276
of open-loop AGC, 438
Bipolar, 198, 266
Bit synchronization, *see* Timing recovery
Blind adaptation, 398
Block synchronization, 451
Branch in trellis diagram, 305
Branch metric, 305

Calculus of variations, 242
Capacity
of Additive Gaussian White Noise (AGWN) channel, 117
of DC-free codes, 188
of ideal low-pass channel, 117
of noisy dispersive channels, 119
of run-length-limited codes, 176
Carrier modulation, 105
Carrier recovery, 105
Carrier serving area (CSA), 47
Carrier synchronization, 451
Causality, 4
Central equalizer tap
non-uniqueness of, 379, 414
Central limit theorem, 28
Channel
capacity, *see* Capacity
equivalent discrete-time model, 129
hybrid, 111
impulse response, 109
memory length, 335
pure delay, 335
symbol response, 111
Charge pump, 520
Charge-constrained modulation code, *see* DC-free modulation code
Class IV, 266, 315, 548
Clock preamble, *see* Clock run-in pattern
Clock recovery, *see* Timing recovery
Clock run-in pattern, 453, 469
Coefficient blow-up, 418
Coherent demodulation, 483
Compact disc (CD), 74, 163
Conditioned diphase, 201
Constant bit-rate recording, 58
Constrained optimization, 35
Convolution
associativity, 38, 40
commutativity, 38, 40
distributivity, 38, 40
of continuous-time signals, 38
of discrete-time sequences, 40
Cosine filter, 381, 427
Cramer-Rao bound, 506
Crank-Shaft, 197
Cross-talk
in digital subscriber lines, 52
in magnetic recording, 72
in optical recording, 86
Cycle slip, 581, 610
Cyclostationary random process, 19

$\mathcal{D}$-transform, 7
Damping factor, 603
DAT (Digital Audio Tape), 55, 56
Data separator, 519
Data transmission, 105
baseband, 47, 105
carrier-modulated, 47
DC offset
adaptive compensation, 425
effect on AGC, 395
effect on MMSE adaptation loop, 418
DC-free modulation codes, 185–210
linear, 187, 195
maxentropic, 189
nonlinear, 201
Viterbi detection for, 310
DC-free modulation codse
Viterbi detection for, 344
DC-wander, 158
DCC (Digital compact cassette), 55
DDFE, *see* Dual DFE
Decision feedback sequence estimator (DFSE), 324, 326, 329
Decision region, 340
Decision-feedback equalizer (DFE), 265–300
error propagation, 270

minimum mean-square error (MMSE), 281
zero-forcing (ZF), 277
Delay-tracking loop, 451
Derivative equalizer, 381, 400, 410
Desired response, *see* Target response
DFE, *see* Decision-feedback equalizer (DFE)
Dibit response, 69
Dicode, 197, 266
Difference path metric, 318, 360
Digital magnetic recording, 55–74
nonlinear bit-shifts, 90
Digital optical recording, 74–87
nonlinear bit-shifts, 95
Digital subscriber line (DSL), 47–54
Digital sum variation (DSV), 187
Digital timing recovery, 457
Digital versatile disc (DVD), 74, 209
Digital video recording, 55, 56, 58
Diphase, 197
Disparity, 203
Distance metric, 123
Domain bloom, 80, 84, 95
Down-conversion, 483
Dropouts, 73
DSL, *see* Digital subscriber line
DSV, *see* Digital sum variation
Dual DFE (DDFE), 342
Duobinary, 200, 266, 551
DVD, *see* Digital versatile disc
Dynamic programming, 304

Early-late timing recovery, 533
Echo cancellation, 49
EEPRIV, 267
Efficiency
of maximum-likelihood timing recovery, 506
of MMSE adaptation loop, 416
of MMSE AGC, 392
of MMSE timing recovery, 539
of modulation codes, 172
of non-data-aided AGC, 393
of orthogonalized ZF adaptation loop, 409
of timing-recovery schemes, 463
of ZF AGC, 389
of ZF timing recovery, 550
EFM, 80, 84, 209
EFMPlus, 209
Embedded servo, 58
Ensemble, 15
EPRIV, *see* Extended class IV
Equalizer
T-spaced, 231
decision-feedback, 277
fractionally-spaced, 231, 254, 292, 481
full-response, 270
linear, 231
partial-response, 266
Erasure, 342
Ergodicity, 26
Error event, 192, 345
Error propagation, 338
catastrophic, 174, 268
in decision-feedback equalizer, 270
in decision-feedback sequence estimator, 326
in dual DFE, 343
in modulation decoder, 114
quasi-catastrophic, 342, 356
Error sequence, 133
Ethernet, 49
Euclidian distance, 123, 132, 345
Euclidian weight, 133, 144
Excess bandwidth, 109
Excess mean-square error (EMSE) due to gradient noise, *see* gradient noise
Extended class IV, 266
Extended decision feedback equalizer (EDFE), 340
Eye pattern, 51, 157

False-lock detection, 511
FDDI, 49
FDTS, *see* Fixed-delay tree search
Feedback detector, 269
Ferguson detector, 318
Filter
all-pass, 6, 9, 32
analog, 382
finite impulse response (FIR), 8
group delay, 5, 7

impulse response, 4
infinite impulse response (IIR), 8
linear-phase, 7
memory-accumulator, 381
minimum-phase, 31
phase characteristic, 5, 7
poles and zeros, 7
power transfer function, 20
table look-up, 381
tapped delay line, *see* Filter, transversal
transfer function, 4
transversal, 8, 380
Fixed-delay tree search, 337
FLL, *see* Frequency-locked loop
Folded signal-to-noise ratio, 112
Forward dynamic programming, 304
Fourier transform
for continuous-time signals, 2, 37
for discrete-time sequences, 2, 40
Fractionally-spaced equalizer (FSE), 231, 254, 292, 481
Frame synchronization, 451
Free-running frequency, 593
Frequency detector, 470, 486, 488, 518
Frequency discriminator, *see* Frequency detector
Frequency doubling code, 201
Frequency modulation (FM) code, 201
Frequency synthesizer, 470
Frequency-locked loop (FLL), 470, 488
Full-response equalizer, 231, 270

Gaussian noise, 28, 143
Gear-shift techniques, 429
Geometric average, 29
Gradient noise
in MMSE adaptation, 416
in MMSE AGC, 392
in ZF adaptation, 408
in ZF AGC, 388
misadjustment due to, 389
Gradient search, 501
Group delay, 5, 7
Guided scrambling, 202

Hamming weight, 350
Hard disk drive (HDD), 56, 466
Harmonic average, 29
Harmonic lock, 612
Hartley-Shannon capacity formula, 118
HDSL, 48, 54
Hedeman codes, 197
Helical scanning mechanism, 56
Hilbert transform, 34, 42
Hybrid channel, 111
Hybrid convolution, 12

Impulse noise, 53
Independence assumption, 431
Inductive recording head, 67
Integrated Services Digital Network (ISDN), 47
Integrator leakage
effect on AGC, 396
effect on MMSE adaptation, 421
Interpolation, 457, 476
Intersymbol interference (ISI), 50, 156, 277
nonlinear, 65, 84, 95

Jitter, 454
accumulation in a chain of transmission units, 621
noise-induced, 463
pattern-dependent, 463
spectrum, 454
transition, 66, 80, 92
Joint random processes, 24

Kurtosis, 431, 438

Lagrange multiplier, 35
Latency, 314, 596
Lattice filter, 381
Leaky LMS, 396
Learning curve, 394
Least mean-square adaptation, *see* Minimum mean-square error adaptation
Lee weight, 366
Line code, 107
Linear equalizer, 231–264
for run-length-limited codes, 232, 246
full-response, 231, 270

minimum mean-square error (MMSE), 232, 240
partial-response, 270
zero-forcing (ZF), 232
Linear modulation codes, 195
Linear pulse modulator, 11
Linearity, 3
LMS, *see* Minimum mean-square error adaptation
Loop filter, 591
adaptive, 472
Lorentzian channel, 69
Low-disparity modulation codes, 204

Magnetic field modulation, 78
Magnetic recording, *see* Digital magnetic recording
Magnetic tape recording, 56
Magneto-optical (MO) recording, 75, 78
Magneto-resistive (MR) recording head, 67, 73
Manchester code, 201
Mastering, 78
Matched filter
for colored noise, 126
for white noise, 124
Matched spectral null (MSN) codes, 193
Matched-filter bound, 127
Maxentropic modulation code
DC-free, 189
run-length-limited, 177
Maximum a posteriori probability (MAP) detection, 143
Maximum-length shift-register sequence, 213, 214
Maximum-likelihood (ML) detection, 143
Maximum-likelihood (ML) timing recovery, 500
Maximum-likelihood sequence detection (MLSD), 123
Mean-square error (MSE), 241, 275
Memory length, 335
Memory-accumulator filter, 381
Memoryless inverse mapping (MIM), 268
MFM, *see* Modified frequency modulation
Miller code, *see* Modified frequency modulation
Milller-squared (M^2), 209
Minimum mean-square error (MMSE) adaptation, 412–421
coefficient blowup in, 418
efficiency factor of, 416
of transversal filter, 417
orthogonalized, 415
Minimum mean-square error (MMSE) AGC, 390
Minimum mean-square error (MMSE) equalizer
decision-feedback, 281
linear, 232, 240
partial-response, 275
Minimum mean-square error (MMSE) timing recovery, 535–544
Minimum-phase function, 31
Minimum-phase root, 277
Modified duobinary, 200, 266
Modified frequency modulation (MFM), 179, 209
Modulation, 11
Modulation code, 153–230
balanced, *see* Modulation code, DC-free
bi-mode, 204
charge-constrained, *see* Modulation code, DC-free
DC^2-constrained, 192
DC-free, 161, 185, 187, 310
effect on timing recovery, 167, 458
efficiency, 172
fixed-length, 172
information rate, 107
linear, 187, 195
low disparity, 204
maximum run-length (k), 313
minimum run-length (d), 309
polarity-bit, 183, 201
pseudoternary, 199
run-length-limited (RLL), 162, 168, 174, 309, 313
self-punctuating, 180
staggered, 197
systematic, 180
transparency, 154
unit-disparity, 205

variable-length, 172
zero-disparity, 202
Monic impulse response, 269
MS43, 208
MSE, *see* Mean-square error
Multicarrier modulation, 54, 138

Natural frequency, 603
Newton/LMS adaptation, 415
Noise
colored, 19
Gaussian, 28
white, 19
Noise enhancement, 239
Noise whitening, 125, 286
Nonlinear bit-shifts, 62, 64, 90
Nonlinear intersymbol interference
decision-feedback equalizer for, 282
in magnetic recording, 65
in optical recording, 84, 95
Viterbi detector for, 328
Nonlinear spectral line methods, 524
NRZ (non-return-to-zero), 108, 120
NRZI (non-return-to-zero-inverse), 176, 519
Nyquist frequency, 109
Nyquist rate, 109
Nyquist-1 function, 157
Nyquist-1 pulse, 11, 233

One-shot bound, *see* Matched-filter bound
Optical mastering, 78
Optical recording, 74–87
magneto-optical (MO), 75, 78
phase-change, 75
read-only, 74
rewritable, 75
servo, 77
write-once read-many (WORM), 74
Overwrite noise, 66

Packet synchronization, 451
PAM, *see* Pulse amplitude modulation
PAM process, 22
Parameter variations, 373
Parseval identity, 39, 41
Partial response Viterbi detector, 316
Partial-response equalizer (PRE), 265–300
minimum mean-square error (MMSE), 275
with Viterbi detector, 316
zero-forcing (ZF), 270
Partial-response techniques, 266
Path memory, *see* Viterbi detector, path memory
Path metric, 306
Pattern noise, *see* Pattern-dependent jitter
Pattern-dependent jitter, 463
Peak detector, 69, 519
Peak distortion, 403
Peak eye closure, 403
Peak shift, 519
PFD, *see* Phase-and-frequency detector
Phase characteristic, 5, 7
Phase detector (PD), 451, 591, 592
Phase noise, 454, 600
Phase unwrapping, 486
Phase-and-frequency detector (PFD), 518
Phase-change recording, 75
Phase-locked loop (PLL), 451, 591–624
acquisition, 465, 612
acquisition range, 466
aided acquisition, 469
capture range, 466
cycle slip, 466, 581, 610
direct lock range, 466
equivalent noise bandwidth, 599
false lock, 462, 505, 511
false lock range, 612
first-order, 593
frequency pull-in, 466, 612
hangup, 462, 485, 531
harmonic lock, 612
hold range, 611
lock detection, 581
lock indicator, 470
lock time, 466, 612
loop filter, 472, 591
loop SNR, 611
loop stress, 469, 597
phase lock-in, 466, 612
pull-in range, 466, 612
pull-in time, 466, 612
pull-out range, 611
second-order, 593, 603

seize range, 466
static operating range, 611
unlock behavior, 610
Pilot tones for track following, 58
Planar filtering, 483, 485, 512, 531
Poisson summation formula, 39
Polarity-bit codes, 183, 201
Post-cursive intersymbol interference (ISI), 277
Power spectral density (PSD)
of continuous-time random process, 18
of discrete-time random process, 16
Power transfer function, 20
Pre-cursive intersymbol interference, 277
Preamble, 458, 466, 470
Precoder, 176, 199, 222, 268
Prefix code, 180
PRIV, *see* Class IV
PRML, 182, 315
Probability density function (PDF), 27
Pseudorandom sequence, 170, 213
cycle-and-add property, 221
Pseudoternary code, 199
Pulse amplitude modulation (PAM), 11, 108

Q-function, 28
Quadra-Phase (QP), 197
Quadratic-Residue arrays, 381
Quadricorrelator, 488
Quality factor, 44
Quantized feedback, 426
Quiescent frequency, *see* Free-running frequency

Raised-cosine pulse, 233
zero crossings, 512
Random process, 15–28
RDS, *see* Running digital sum
Recording code, 107
Recording head
inductive, 67
magneto-resistive, 67, 73
Register-exchange path memory, 315
Resonance frequency, 44
Roll-off factor, 233
Root raised-cosine, 253
Rubbing noise, 74
Run-length trellis diagram, 328
Run-length-limited (RLL) codes, 174–185
anti-whistle, 171, 183
decision-feedback equalization for, 287
for PRML systems, 182
linear equalization for, 232, 246
maxentropic, 177
Viterbi detection of, 328
Running digital sum (RDS), 187, 310

S-curve, *see* Timing function
Sampled-derivative timing recovery, 541
Sampling
of deterministic signals, 10
of stochastic signals, 20
Sampling theorem, 10
Scrambler, 169, 458
frame-synchronized, 170
self-synchronized, 170, 212
Self-noise, *see* Pattern-dependent jitter
Self-orthogonalizing adaptation, 430
Self-recovering adaptation, 398
Self-timing, 451
Sequence detection, 123
Sequence feedback, 324
Servo
magnetic recording, 58
optical recording, 77
Set partitioning, 332
Sign algorithm, 419, 428
Signal correction, 376
Signal estimation, 376
Signal suppression factor
of MMSE AGC, 391
of non-data-aided AGC, 393
of open-loop AGC, 438
of ZF AGC, 393
Signaling rate, 107
Slant azimuth recording, 72
Sliding-block decoder, 174
Slope equalizer, 376, 382, 398, 427
Smearing filters, 53
Soft-decision decoding, 195
Squaring loop, 524
Stability, 4

Stationarity, 15
strict-sense, 17
wide-sense, 17
Stochastic process, *see* Random process
Subharmonic lock, 612
Subset trellis, 335
Sufficient statistic, 124, 537
Symbol response
of hybrid channel, 111
of linear pulse modulator, 12
Symbol-by-symbol detection, 123
Synchronization, 451
Systematic jitter, *see* Pattern-dependent jitter
Systematic modulation code, 180

Table look-up filter, 381
adaptive, 426
Tape recording, 56
Tapped delay line, *see* Transversal filter
Target response, 159
for decision-feedback equalizer, 277
for partial-response equalizer, 271
for Viterbi detector, 315
Thermal asperities, 74
Threshold adaptation, 84, 424
Time invariance, 3
Time-division multiplex, 106
Timing function, 462
Timing recovery, 451–590
acquisition, 453, 465, 469
all-digital, 457, 476
based on threshold crossings, 508
clock run-in pattern, 469
closed-loop, 456
data-aided (DA), 453, 500
decision-directed (DD), 453
deductive, 455
early-late, 533
efficiency, 463
inductive, 455
maximum-likelihood (ML), 500, 507
minimum mean-square error (MMSE), 535
non-data-aided (NDA), 453, 507
nonlinear spectral line methods, 524
open-loop, 456
preamble, 458
S-curve, *see* Timing recovery, timing function
sampled-derivative, 541
squaring loop, 508, 524
timing function, 462
tracking, 453
training sequence, 458
wave difference method, 535
zero-forcing (ZF), 544, 584
Timing-error detector (TED), 451, 591
Tomlinson/Harashima precoding, 283
Traceback path memory, 315
Track following, *see* Servo
Trailing intersymbol interference, *see* post-cursive intersymbol interference
Training sequence, 458, 470
Transition noise, 92
in magnetic recording, 66
in optical recording, 80
Transition response, 69
in digital magnetic recording, 69
in digital optical recording, 81
Transmit filter, 250
for decision-feedback equalizer, 282
for linear equalizer, 250
for partial-response equalizer, 276
Transparency of modulation code, 154
Transversal filter, 8, 380
MMSE adaptation of, 417
ZF adaptation of, 407
Trellis diagram, 187, 305, 328
Trellis precoding, 138
Trellis-coded modulation, 119, 138
Twinned binary, 197
Two-level AMI Class II, 201

Unbalance, 203
Unbalance trellis diagram, 187, 310
Ungerboeck partitioning, 334
Unit-disparity modulation codes, 205

Variance, 27
VCA, *see* Voltage-controlled amplifier
VCO, *see* Voltage-controlled oscillator
VDSL, 48, 54
Viterbi detector, 301–372

add-compare-select (ACS) unit, 314
branch metric, 305, 307
branch metric computation (BMC) unit, 314
desired response, *see* Viterbi detector, target response
detection delay, 312
difference metric, 314, 317, 318, 347, 360
effect of restricted detection delay, 312, 340, 350
Ferguson's algorithm, 318
for charge-constrained codes, 344, 346
for run-length-limited codes, 329, 360
path memory, 314, 320, 323, 362
path metric, 306, 311
path registers, *see* Viterbi detector, path memory
prefiltering, 315
preliminary decisions, 325
sequence feedback, 324, 326, 329
set partitioning, 331
state, 307
subset state, 335
surviving path, 306, 311
target response, 315
truncation depth, 312
Voltage-controlled amplifier (VCA), 384
Voltage-controlled oscillator (VCO), 453, 591
free-running frequency, 593
phase noise, 600
quiescent frequency, *see* Voltage-controlled oscillator, free-running frequency

Wal-1, 197
Wander, 454
baseline, 158, 211
time-base, 454
White noise, 19
Whitened matched filter (WMF), 131
canonical, 131, 278
Wiener-Kinchine relation, 17, 18
Word synchronization, 451
Write equalizer, 197
Write precompensation, 62, 65, 92

z-transform, 7
Zero-disparity modulation codes, 202
Zero-forcing (ZF) adaptation, 400–411
efficiency factor, 409
gradient noise, 408
of analog equalizer, 410
of transversal filter, 407
orthogonalized, 407
Zero-forcing (ZF) AGC, 387
Zero-forcing (ZF) equalizer
decision-feedback, 277
linear, 232
partial-response, 270
with finite number of adjustable parameters, 400
Zero-forcing (ZF) timing recovery, 544–562
for class IV partial-response system, 548, 584
for duobinary partial-response system, 551, 584
for full-response system, 555
Zero-Modulation, 209
Zero-phase restart, 471, 523
Zone-density recording, 58