INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.
CHANNEL EQUALIZATION TECHNIQUES
APPLIED TO
DIGITAL STORAGE AND TRANSMISSION SYSTEMS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Inkyu Lee
June 1995
I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

[Signature]
John M. Cioffi
(Principal Adviser)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

[Signature]
Bernard Widrow

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

[Signature]
Omer S. Inan

Approved for the University Committee on Graduate Studies:

[Signature]

iii
Abstract

The goal of communication systems is to transfer information reliably through various media. The content of information could be in any format including voice, image and computer data. Digital signal processing and communication theory have been applied to many digital transmission systems, and their theoretic analysis and performance evaluation on many transmission channels such as voice modems and High-speed Digital Subscriber Loops have been carried out for decades. In contrast to the advancement of signal processing techniques in digital transmission channels, the application of digital signal processing to data storage systems such as a magnetic recording disk drive has been relatively recent.

Both digital transmission and storage systems have the similar goal. Digital storage systems try to achieve the maximum data density, while digital transmission systems try to maximize the data rate. Therefore, the only difference in their systems is a matter of how they “transmit” or “store” data through their media. Digital storage channels can be thought of as conveying data temporally, while digital transmission systems transmit data spatially.

This dissertation explores channel equalization as a means to optimize the performance in digital transmission and storage systems. Within the signal processing framework, we treat both systems as a communication channel. We propose many signal processing techniques in this dissertation that also improve the performance in communication systems.

We introduce a fast computation algorithm for the Decision Feedback Equalizer (DFE). The proposed algorithm incorporates the Discrete Fourier Transform (DFT) to compute the DFE coefficients very efficiently. Performance of the fast algorithm is
verified on various channels including a magnetic recording channel and High-speed Subscriber Loops. Although the fast algorithm provides an approximate solution, simulation shows that the approximation is so close to the optimum solution that it is only a few tenths of a dB away from the optimum solution. This non-recursive algorithm is based on knowing the channel pulse response.

We also propose an Equalized Maximum Likelihood (EML) receiver for a magnetic recording system. The EML receiver is shown to achieve more than a 2 dB gain over the conventional Partial Response Maximum Likelihood (PRML) method at high densities. Also, we investigate several effects of nonlinearity in high density magnetic recording system.
Acknowledgments

I would like to express my deepest gratitude to my research advisor, Professor John Cioffi for his valuable support and guidance. His insights helped me approach my research efficiently and facilitated my completion of this thesis.

I want to thank Professor Bernard Widrow for serving as my associate advisor, Professor Butrus Khuri-Yakub for agreeing to be the chairman of my orals committee, Professor Madihally Narasimha for being a member of my orals committee, and Professor Umran Inan for being a member of my reading committee.

I enjoyed collaborating with my colleagues in the “storage” group: Bill Abbott, Phil Bednarz, Kevin Fisher, Susan Lin, Cory Modlin, Nick Sands, Paul Voois and Takashi Yamauchi. I also enjoyed the company and discussions with other officemates in Durand 112 — Naofal Al-Dhahir, Jacky Chow, Peter Chow, Ivo Dobbelaere, Minnie Ho, Krista Jacobsen, Hui-Ling Lou, Jose Tellado, Rick Wesel and Katie Wilson. Special thanks to Joice DeBolt for her endless assistance during these years.

My grandfather, a historian and an educator, was a great role model for me. Although he passed away when I was in middle school, his advice has been always with me. I believe he would be happy with what I’ve done.

Finally, my greatest thanks go to my family who have always encouraged me throughout my studies. Without their love and support, my work would have been much more difficult. It is to them I dedicate this dissertation.
Contents

Abstract ................................................................. iv
Acknowledge ............................................................. vi
List of Tables .......................................................... x
List of Figures .......................................................... xi

1 Introduction ......................................................... 1
  1.1 Digital Transmission and Storage Channel .................. 2
  1.2 Channel Equalization .......................................... 4
  1.3 Outline of Dissertation ....................................... 7
  1.4 Contributions ................................................ 8

2 Receivers in Digital Transmission/Storage System .......... 10
  2.1 Channel model ............................................... 10
  2.2 Design of Receivers ......................................... 12
    2.2.1 MMSE-DFE ............................................. 12
    2.2.2 MMSE-LE ............................................. 20
    2.2.3 PRML ................................................. 21
  2.3 Summary ..................................................... 23

3 A Fast Computation Algorithm for the Decision Feedback Equalizer 24
  3.1 Introduction ................................................ 24
  3.2 Fast Equalization Algorithm ................................ 27
  3.3 Channel Identification ...................................... 31
  3.4 Simulations ................................................ 33
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5</td>
<td>Conclusion</td>
<td>39</td>
</tr>
<tr>
<td>4</td>
<td>Performance of A Fast Algorithm in High-speed Subscriber Loop</td>
<td>40</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>40</td>
</tr>
<tr>
<td>4.2</td>
<td>Discrete Multitone (DMT) Modulation</td>
<td>43</td>
</tr>
<tr>
<td>4.3</td>
<td>Equalizer settings for the TEQ</td>
<td>47</td>
</tr>
<tr>
<td>4.4</td>
<td>A Fast Equalization Algorithm</td>
<td>48</td>
</tr>
<tr>
<td>4.5</td>
<td>Simulations</td>
<td>50</td>
</tr>
<tr>
<td>4.6</td>
<td>Conclusion</td>
<td>53</td>
</tr>
<tr>
<td>5</td>
<td>Equalized Maximum Likelihood Receiver in a Storage System</td>
<td>56</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>56</td>
</tr>
<tr>
<td>5.2</td>
<td>Structure of EML and channel model</td>
<td>59</td>
</tr>
<tr>
<td>5.3</td>
<td>EML with a unit tap constraint</td>
<td>61</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Equalizer settings</td>
<td>62</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Design of an unbiased receiver</td>
<td>64</td>
</tr>
<tr>
<td>5.3.3</td>
<td>Comparison with PRML</td>
<td>65</td>
</tr>
<tr>
<td>5.4</td>
<td>EML with a unit energy constraint</td>
<td>72</td>
</tr>
<tr>
<td>5.4.1</td>
<td>EML settings</td>
<td>73</td>
</tr>
<tr>
<td>5.4.2</td>
<td>Comparison with PRML</td>
<td>75</td>
</tr>
<tr>
<td>5.4.3</td>
<td>An adaptive algorithm for EML</td>
<td>78</td>
</tr>
<tr>
<td>5.5</td>
<td>The complexity of the Viterbi detector</td>
<td>79</td>
</tr>
<tr>
<td>5.6</td>
<td>Comparison between a unit tap and a unit energy constraint</td>
<td>81</td>
</tr>
<tr>
<td>5.7</td>
<td>Conclusion</td>
<td>82</td>
</tr>
<tr>
<td>6</td>
<td>A Nonlinear Channel Model for Digital Storage System</td>
<td>83</td>
</tr>
<tr>
<td>6.1</td>
<td>A Simple Partial Erasure Model</td>
<td>83</td>
</tr>
<tr>
<td>6.1.1</td>
<td>Nonlinear Channel Model</td>
<td>84</td>
</tr>
<tr>
<td>6.1.2</td>
<td>Minimum Mean Square Error Decision Feedback Equalizer (MMSE-DFE)</td>
<td>86</td>
</tr>
<tr>
<td>6.1.3</td>
<td>DFE with a tree search</td>
<td>87</td>
</tr>
<tr>
<td>6.1.4</td>
<td>Linear Partial Response Maximum Likelihood (PRML)</td>
<td>88</td>
</tr>
</tbody>
</table>
List of Tables

5.1 Target response b of EML ........................................ 68
5.2 Target response b of EML ........................................ 78
6.1 Comparison of SDR ................................................. 86
List of Figures

1.1 Digital transmission and storage channel .......................... 3
1.2 Comparison of a Zone-Bit Recording ............................... 6
2.1 Structure of DFE ................................................. 13
2.2 The effect of decision delay ...................................... 18
2.3 The optimum decision delay ....................................... 19
2.4 Structure of LE .................................................. 21
2.5 Structure of PRML ............................................... 22
3.1 Flow chart of the fast algorithm ................................. 32
3.2 Pulse responses of the Lorentzian model ......................... 34
3.3 Performance of the fast algorithm in Lorentzian channel with $N = 6$ . 35
3.4 Performance of the fast algorithm in ADSL channel with $N = 25$ . . 36
3.5 The effect of the imperfect channel estimation .................. 37
3.6 The effect of finite precision arithmetic ......................... 38
4.1 Learning curve of the LMS algorithm ........................... 41
4.2 Impulse response of 400m 26 gauge loop ....................... 42
4.3 Basic Structure of a DMT System ............................... 44
4.4 Block Diagram of a DMT Transmitter ........................... 44
4.5 Block Diagram of a DMT Receiver ............................... 45
4.6 Performance of the fast algorithm with different feedforward filter length 51
4.7 Performance of the fast algorithm with different feedforward filter length 52
4.8 Performance of the fast algorithm with different target response length 54
4.9 Performance of the fast algorithm with different target response length 54
4.10 Performance of the fast algorithm with different MFB ........... 55
4.11 Performance of the fast algorithm with different Knext

5.1 The structure of EML in infinite length case

5.2 The structure of EML in finite length case

5.3 Spectra of the partial responses

5.4 SNR gain of EML over PR4

5.5 SNR gain of EML over EPR4

5.6 Spectra comparison at pw50/T = 1.5

5.7 Spectra comparison at pw50/T = 3.5

5.8 Comparison of Bit Error Rate

5.9 Effective SNR of EML

5.10 SNR gain of EML over PR4

5.11 SNR gain of EML over EPR4

5.12 Bit Error Rate Comparison

5.13 Metric computation using a look-up table

6.1 Nonlinear channel model

6.2 Structure of DFE with tree search

6.3 Trellis diagram of a simple partial-erasure model

6.4 Error sequences corresponding to dmin

6.5 Comparison of SNR

6.6 Error events distributions for the nonlinear PRML

6.7 Error events distributions for the linear PRML

6.8 Probability of error

xii
Chapter 1

Introduction

The goal of communication systems is to transfer information reliably through various media. The content of information could be in any format including voice, image and computer data. Digital signal processing and communication theory have been applied to many digital transmission systems, and their theoretic analysis and performance evaluation on many transmission channels such as voice modems and High-speed Digital Subscriber Loops have been carried out for decades. In contrast to the advancement of signal processing techniques in digital transmission channels, the application of digital signal processing [1] to data storage systems such as a magnetic recording disk drive has been relatively recent.

Both digital transmission and storage systems have the similar goal. Digital storage systems try to achieve the maximum data density, while digital transmission systems try to maximize the data rate. Therefore, the only difference in their systems is a matter of how they “transmit” or “store” data through their media. Digital storage channels can be thought of as conveying data temporally, while digital transmission systems transmit data spatially.

This dissertation explores channel equalization as a means to optimize the performance in digital transmission and storage systems. Within the signal processing framework, we treat both systems as a communication channel. We propose many signal processing techniques in this dissertation that also improve the performance in communication systems.
Chapter 1. Introduction

We introduce a fast computation algorithm for the Decision Feedback Equalizer (DFE). The proposed algorithm incorporates the Discrete Fourier Transform (DFT) to compute the DFE coefficients very efficiently. Performance of the fast algorithm is verified on various channels including a magnetic recording channel and High-speed Subscriber Loops. Although the fast algorithm provides an approximate solution, simulation shows that the approximation is so close to the optimum solution that it is only a few tenths of a dB away from the optimum solution. This non-recursive algorithm is based on knowing the channel pulse response.

We also propose an Equalized Maximum Likelihood (EML) receiver for a magnetic recording system. The EML receiver is shown to achieve more than a 2 dB gain over the conventional Partial Response Maximum Likelihood (PRML) method at high densities. Also, we investigate several effects of nonlinearity in high density magnetic recording system.

To provide a background on the digital communication theory, we present a general description of a digital transmission and storage system in Section 1.1. Section 1.2 explains the channel equalization methods and other techniques used in practical systems. We then summarize the contents of each chapter and the contributions of this dissertation.

1.1 Digital Transmission and Storage Channel

One usually thinks of a communication system as transmitting information over space. However, in a digital storage system such as a hard-disk driver in a PC, the data transmission takes place over time. Therefore, there is no reason to distinguish the digital transmission system from the digital storage system. We will analyze both systems under the same framework.

Figure 1.1 shows a block diagram of a communication channel. The input is the data to be transmitted over a twisted copper pair, or stored on the magnetic disk. Though not shown in the figure, the input stream is usually assumed to be compressed using a source coding scheme and then converted using an Error Correcting Code (ECC). Channel coding occurs at a modulation encoder. Powerful codes such as
Chapter 1. Introduction

Trellis code are popular in many transmission channels. For a magnetic recording channel, a Matched Spectral Null (MSN) code exploiting the channel spectrum has been proposed for a channel coding scheme.

The modulation block converts the encoded data stream into a continuous time signal. Several signaling schemes can be found in communication systems. In a magnetic recording channel, the hysteresis effect usually restricts use to only a two-level Pulse Amplitude Modulation (PAM).

The channel represents the physical media on which the actual data transmission and storage take place. To develop signal processing techniques suited to their applications, one must be able to analyze the channel characteristics thoroughly. Several channel impairments can disturb reliable transmission. For example, impulse noise may corrupt the signal in a transmission channel and nonlinear effects may interfere with the data transition in a magnetic recording channel.

The next block indicates the channel equalization combined with an equalizer and a detector. To reduce the Inter-Symbol Interference (ISI), it is often desirable to equalize the channel output to the desired response. Then the detector (or decoder, corresponding to a channel coding scheme applied in encoder section) takes the equalized signal output and estimates the original input data stream.

From the perspective in Figure 1.1, the only difference in transmission and storage systems is the physical channel. Therefore, we can apply the same signal processing
Chapter 1. Introduction

techniques and digital communication theory, which have been widely used in many transmission systems, to the storage system without any distinction. All the channel equalization schemes in this dissertation can be applied to both transmission and storage systems.

1.2 Channel Equalization

As explained in Figure 1.1, a block representing an equalizer and a detector characterizes the channel equalization functionally in the digital transmission and storage channel.

As the data transmission rate or recording density increases, each symbol in the transmitted data becomes more closely spaced and begins to interfere with one another. Several signal processing techniques have been derived to combat the ISI. Channel equalization is one of the powerful techniques that is widely used to reduce the ISI and then to detect the input estimates.

Channel equalization methods are used to mitigate the effect of ISI [2]. The equalization schemes try to convert a bandlimited channel with ISI into one that appears memoryless. Although channel equalization is sub-optimal for detection, it has been applied in many systems where the optimal Maximum Likelihood (ML) receiver is too costly and complex to implement in practice. Therefore, channel equalization is an attractive, sub-optimal, and cost effective detector in many practical communication systems.

The simple equalization structure is the Linear Equalizer (LE), which will be explained in the following chapter in detail. The LE tries to convert the channel so that the input to a detector at the end of the LE appears memoryless. In general, two criteria can be considered for choosing the LE coefficients: Zero Forcing and Minimum Mean Square Error criteria. Each criteria optimizes the equalizer settings to compensate for the ISI given constraints.

One of other popular channel equalization techniques is the Decision Feedback Equalizer (DFE). This structure will be also analyzed in detail later. The DFE consists of two finite length filters: a feedforward filter and feedback filter. The
combined structure provides a performance improvement over the LE in terms of reducing the effect of ISI.

Especially for the magnetic recording channel, the Partial Response Maximum Likelihood (PRML) receiver has attracted widespread attention as a detector. In PRML, a feedforward filter is used to equalize the channel output to the fixed Partial Response and then a Viterbi detector tuned to the Partial Response is employed to estimate the input sequences. In this dissertation, a channel equalization scheme combined with a Viterbi detector is proposed to improve the performance of the PRML-like receiver.

There are other signal processing techniques besides channel equalization to increase the data rate or density. One popular choice in a magnetic recording channel is with Run-Length-Limited (RLL) codes. They have been used extensively in peak-detection systems to increase the minimum transition distance between two adjacent input symbols. An RLL code is specified by two parameters \( d \) and \( k \), where \( d \) is the minimum and \( k \) is the maximum numbers of zeros that can occur between ones in the input bit sequence. Usually, a \( d \) constraint is used to restrict the minimum space between two adjacent input bits so that the effect of ISI and nonlinear distortion can be reduced in a high density recording system. Also, a \( k \) constraint is applied to aid in timing recovery. While the use of RLL codes are fundamental to the optimization of a peak-detection system, benefits are made at the expense of the rate loss, which is critical in real products.

Another scheme used in a magnetic recording industry is a technique called "Zone-Bit Recording." When a magnetic media disk is manufactured, the disk is divided into several zones. As data is stored and retrieved in each zone, the sampling rate is adjusted accordingly for each zone. As a result, a higher data density is attained. In current products, disk drives typically use 3 to 16 zones. Figure 1.2\(^1\) explains the data density specified by \( pw_{SO}/T \) with and without the Zone-Bit Recording technique across the disk surface. Without dividing the surface, the data density decreases across the disk monotonically. In contrast, a higher data density is achieved when

\(^{1}\)We would like to thank John Kuklewicz of Philips Semiconductors for providing the data for Figure 1.2.
Figure 1.2: Comparison of a Zone-Bit Recording
five zones are partitioned. This model assumes equally spaced zones although real drives sometimes use unequally sized zones. The required symbol rates (in MHz) for each zone are:

\[ 72.3823 \quad 87.7635 \quad 103.1448 \quad 118.5260 \quad 133.9072 \]

By increasing the symbol rate for five zones, we can obtain the overall areal density gain.

1.3 Outline of Dissertation

This dissertation can be divided into two parts. One is the fast computation algorithm of the DFE and the other is the EML receiver in a magnetic storage system. We now summarize each chapter in the dissertation.

Chapter 2 characterizes the channel model that is used throughout this dissertation, then explains three popular receivers in digital communication systems: DFE, LE and PRML.

Chapter 3 introduces the fast computation algorithm for the DFE. The DFE has a performance advantage over the LE, but the slow convergence rate in adaptive equalization and high-cost in direct computation prevent the DFE from wide application in digital communication systems. These problems motivate the fast algorithm of the DFE coefficients. Approximation of the Toeplitz autocorrelation matrices to circulant matrices enables us to incorporate the DFT to compute the DFE settings very efficiently.

The fast algorithm in Chapter 3 is extended to the DMT system in Chapter 4. Several simulations have been carried out in high-speed subscriber loop. Superiority of the proposed algorithm is confirmed through various channel environments. Simulation results indicate that the fast algorithm yields a solution only a few tenths of a dB away from the optimal solution.

Chapter 5 introduces EML in a magnetic recording channel. A feedforward filter is used to convert the channel to a shortened length target channel and then a Viterbi detector matched to the target response is applied for data estimation. Compared to
conventional PRML which uses fixed integer coefficients in a target response, EML does not constrain the target response. Several constraints can be applied to obtain the optimal EML settings. We solve the EML coefficients under both unit-tap and unit-energy constraints and compare the performance with PRML at various recording densities. Simulations show that a 2 dB gain is achieved over PRML at high densities with a little increased complexity. Also, practical implementation issues are investigated including an adaptive algorithm to obtain the EML settings and a look-up table design in a Viterbi detector to enable computation of branch metrics without a real number multiplication.

Chapter 6 investigates the various effects of nonlinear distortion observed in a magnetic recording channel. Two main sources of nonlinearity are the partial-erasure effect and data-dependent noise corruption. We set up a nonlinear channel model to describe each nonlinearity and propose modified receivers to incorporate these nonlinear models.

Finally, Chapter 7 summarizes the main results of this dissertation and suggests areas for future research.

At the end of this dissertation, Appendix A analyzes the performance of the Maximum Likelihood Sequence Detector, taking into account the colored noise and the channel misequalization. The effective SNR is derived under the Gaussian distribution assumption. Also, Appendix B derives the fast computation algorithm for the DFE developed in Chapter 3 in the fractionally spaced equalizer case. The DFE coefficients are obtained by the DFT and IDFT operations.

1.4 Contributions

We conclude this chapter with a brief summary of the contributions made in this dissertation. They are the following:

- Analysis of the effect of the decision delay on the performance of the decision feedback equalizer in Chapter 2.
Chapter 1. Introduction

- Derivation of the fast algorithm for computing the Minimum Mean Square Error Decision Feedback Equalizer coefficients in Chapter 3.

- Design of an Equalized Maximum Likelihood receiver in a magnetic recording channel and its implementation in Chapter 5.

- Modeling of nonlinear distortions in a high density magnetic recording channel and design of improved receivers in Chapter 6.
Chapter 2

Receivers in Digital Transmission/Storage System

In the previous chapter, we have shown that digital storage systems are equivalent to digital transmission systems. Thus, we shall analyze both digital transmission and storage systems without any distinction. We will start with a channel model description that will be used throughout this dissertation, if not stated otherwise. Then, the following sections describe various receivers used in digital transmission and storage systems. They are the Minimum Mean Square Error Decision Feedback Equalizer (MMSE-DFE), the Minimum Mean Square Error Linear Equalizer (MMSE-LE), and finally the Partial Response Maximum Likelihood receiver (PRML). For all three receivers, we will assume a common linear time-invariant channel model described in the following section.

2.1 Channel model

We assume that the pulse response $h(t)$ extends over a finite interval $0 \leq t \leq \nu T$, where $\nu$ is called the "channel memory" and $T$ denotes the symbol period. The
standard additive white Gaussian noise channel model is used throughout this dissertation, where the input/output relation is given by

$$y(t) = \sum_n x_m h(t - mT) + n(t)$$

where $y(t)$ is the channel output, $\{x_m\}$ is the channel input sequence with signal power $\bar{E}_x$, and $n(t)$ is the additive Gaussian noise with variance $\sigma_n^2$. Here we assume the linear time-invariant channel.

With fractionally spaced equalizer sampling at time instants $t = kT - iT, i = 0, 1, \cdots, l - 1$, the input/output relation for the discrete time equivalent channel has the form:

$$y_k = \sum_m h_{m} x_{k-m} + n_k \quad (2.1)$$

where $l$ is the oversampling factor and

$$y_k = \begin{bmatrix} y(kT + \frac{1-1}{l}T) \\ y(kT + \frac{1-0}{l}T) \\ \vdots \\ y(kT) \end{bmatrix}, \quad h_k = \begin{bmatrix} h(kT + \frac{1-1}{l}T) \\ h(kT + \frac{1-0}{l}T) \\ \vdots \\ h(kT) \end{bmatrix}, \quad n_k = \begin{bmatrix} n(kT + \frac{1-1}{l}T) \\ n(kT + \frac{1-0}{l}T) \\ \vdots \\ n(kT) \end{bmatrix}.$$

We assume that the channel input sequence $\{x_k\}$ and the noise sequence $\{n_k\}$ are uncorrelated with each other.

Using equation (2.1) for $M$ successive $l$-tuples of samples of $y(t)$, we can form the following relation:

$$y_k = Hx_k + n_k \quad (2.2)$$

where

$$y_k = \begin{bmatrix} y_k \\ y_{k-1} \\ \vdots \\ y_{k-M+1} \end{bmatrix}, \quad x_k = \begin{bmatrix} x_k \\ x_{k-1} \\ \vdots \\ x_{k-M+1} \end{bmatrix}, \quad n_k = \begin{bmatrix} n_k \\ n_{k-1} \\ \vdots \\ n_{k-M+1} \end{bmatrix},$$
and

\[
H = \begin{bmatrix}
  h_0 & h_1 & \cdots & h_\nu & 0 & \cdots & 0 \\
  0 & h_0 & h_1 & \cdots & h_\nu & \cdots & \vdots \\
  \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
  0 & \cdots & 0 & h_0 & h_1 & \cdots & h_\nu
\end{bmatrix}.
\]

### 2.2 Design of Receivers

This section describes three receiver structures: MMSE-DFE, MMSE-LE and PRML. In Section 2.2.1, we include decision delay as an explicit parameter in analyzing the MMSE-DFE. Our derivation yields an algebraic interpretation of decision delay’s effect on DFE performance (measured by mean-squared error). It also allows the fast computation of the MMSE-DFE for several different values of both decision delay and the number of feedback taps. Our approach is especially useful for short filter lengths, when the decision delay can significantly affect DFE performance. In Section 2.2.2, the MMSE-LE is treated as a special case of MMSE-DFE and its performance is analyzed. Finally, Section 2.2.3 describes the structure of PRML and derives its theoretical performance.

#### 2.2.1 MMSE-DFE

The decision feedback equalizer (DFE) is a well-known receiver structure for communication channels. The DFE decodes channel inputs on a symbol-by-symbol basis and uses past decisions to remove trailing intersymbol interference. The DFE makes use of previous decisions in attempting to estimate the current symbol with a memoryless decision device. The DFE is inherently a nonlinear filter. However, it can be analyzed using linear techniques, assuming that all decisions are correct. In this section\(^1\), we address the problem of computing the optimal finite-length filters and the resulting performance for a DFE on a discrete-time linear channel.

There are two criteria that can be applied to the DFE: a Minimum Mean Squared Error (MMSE) criterion — which we define shortly, and a Zero Forcing (ZF) criterion.

\(^1\)We would like to thank Paul Voois of Stanford University for helpful discussions.
We shall use the MMSE criterion for the DFE to measure performance in this section. Under this criterion, the optimal DFE is known as the Minimum Mean Squared Error Decision Feedback Equalizer (MMSE-DFE). The zero forcing solution will be the special case of the least-squares solution with setting the input SNR to infinity in the MMSE solution.

Figure 2.1 shows the structure of DFE. The DFE consists of a feedforward filter \( \{w_k\} \) and a feedback filter \( \{b_k\} \), as well as a memoryless threshold device that computes decisions on a symbol-by-symbol basis. The input to the threshold device, \( z_k \), represents the optimal estimate of \( x_k \) given the channel output sequence \( \{y_k\} \) and past decisions \( \{x_{k-1}, x_{k-2}, \ldots\} \). The feedback filter accepts as input the decision from the previous symbol period and then subtracts the trailing ISI. The feedforward filter will try to shape the channel output signal so that its maximum energy content is contained within the current sample.

When no constraints are placed on filter length, the optimal DFE filters generally have infinite length [3, 4]. To reduce complexity, improve stability, or allow for adaptability, however, many designs use finite impulse response (FIR) filters in both the feedforward and feedback sections. To preserve the nonanticipatory nature of the feedback, the feedback filter is constrained to be strictly causal; that is, \( b_k = 0 \) for \( k \leq 0 \). The feedforward filter, however, may be noncausal. We let \( N_f \) denote the number of feedforward taps, of which \( \Delta \) taps are anticausal. That is, \( \{w_k\} = \{w_-, \ldots, w_{N_f-\Delta-1}\} \).
Chapter 2. Receivers in Digital Transmission/Storage System

The DFE in Figure 2.1 may be realized using a causal feedforward filter \( \{w_k\} = \{w_0, \ldots, w_{N_f-1}\} \), if the estimated sequence \( \{z_k\} \) is delayed by \( \Delta \) time units. In such a case, the input to the threshold device at time \( k \) is \( z_{k-\Delta} \), which is the optimal estimate of \( x_{k-\Delta} \) given the channel outputs \( \{y_k, y_{k-1}, \ldots\} \) and past decisions \( \{x_{k-\Delta-1}, x_{k-\Delta-2}, \ldots\} \). Thus we may interpret \( \Delta \) as the decision delay inherent in a causal realization of a DFE. The two interpretations of \( \Delta \) — as the number of anticausal feedforward filter taps and as the decision delay inherent in a causal DFE — are equivalent. Note, however, that \( \Delta \) is not associated with any delay in the threshold device itself.

Let \( N_b \) be the number of feedback taps. For a particular set of FIR parameters \( \{N_f, N_b, \Delta\} \), the solution for the optimal DFE filters and the corresponding mean-squared error (MSE) is straightforward [5, 6, 7]. The choice of \( \Delta \) can significantly affect performance, especially in a DFE with short filter lengths. There is no closed-form solution for the \( \Delta \) that minimizes MSE; it must be found by numerical search over all possible \( \Delta \).

Our derivation of the MMSE-DFE includes \( \Delta \) as an explicit parameter. While we do not find a closed-form expression for the optimal \( \Delta \), our solution yields an algebraic interpretation of \( \Delta \). We identify MSE as a Schur complement\(^2\) in a subblock of the matrix \( \mathbf{R}_{z|y} = (\mathbf{R}_{xx}^{-1} + \mathbf{H}^T \mathbf{R}_{nn}^{-1} \mathbf{H})^{-1} \), where \( \mathbf{R}_{xx} \) and \( \mathbf{R}_{nn} \) are the autocorrelation matrices of the channel input and noise sequences, and \( \mathbf{H} \) is the Toeplitz matrix of channel coefficients as defined as in the previous section. The delay \( \Delta \) determines the position of the subblock, and the number of feedback taps \( N_b \) determines its size. The matrix \( \mathbf{R}_{z|y} \) does not depend on \( \Delta \) or \( N_b \), so one need only compute \( \mathbf{R}_{z|y} \) once to determine MSE for several different values of \( \Delta \) and \( N_b \). Thus, our solution gives an efficient method for computing the optimal \( \Delta \).

Figure 2.1 shows the channel and DFE. We model the channel input as a stationary random sequence \( \{x_k\} \). We assume a linear time-invariant channel \( \{h_k\} \) that is nonzero only in the interval \( 0 \leq k \leq \nu \). Thus, the channel memory length is \( \nu \). The channel output is corrupted by a stationary noise sequence \( \{n_k\} \). \( z_k \) represents the

\(^2\)The Schur complement of \( \mathbf{D} \) in the block matrix \[ \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \] is \( \mathbf{A} - \mathbf{CD}^{-1}\mathbf{B} \) [8].
output sequence after the summing junction. We assume that the input and noise sequences have zero mean and are uncorrelated with each other. Each sequence may, however, be self-correlated. Then, the overall channel input/output relationship is defined as in equation (2.1).

The DFE filters have finite lengths and can be represented as vectors. The DFE consists of a fractionally spaced $N_f \cdot l$-tap feedforward vector

$$\mathbf{w} = [\mathbf{w}_0^T \mathbf{w}_1^T \cdots \mathbf{w}_{N_f-1}^T]^T$$

and a symbol spaced $N_b + 1$-tap feedback vector

$$\mathbf{b} = [1 \ b_1 \ b_2 \cdots b_{N_b}]^T,$$

where $\mathbf{w}_i = [w_{0,i} \ w_{1,i} \cdots w_{l-1,i}]^T$.

Our indexing of the feedforward taps (numbered 0 through $N_f - 1$) implies a causal feedforward filter and, hence, interprets $\Delta$ as the decision delay inherent in a causal DFE.\footnote{The alternate interpretation of $\Delta$ — as the number of anticausal feedforward taps — would change our indexing scheme, but would not otherwise affect the results of this section.} Our indexing of the feedback taps (numbered 1 through $N_b$) reflects the nonanticipative nature of the feedback.

We assume that all decisions are correct; that is, the nonlinear decision element maps the estimated sequence $\{z_k\}$ into the input sequence $\{x_k - \Delta\}$. In practice, this may not be true, and can be a significant drawback of the DFE that can not be overlooked. Nevertheless, the analysis becomes intractable if we try to consider errors in the feedback section. To date, the most efficient way to specify the effect of error propagation has often been via measurement. “Tomlinson Precoding” can be employed to eliminate the error propagation effect [9].

The least-squares approach requires expressions for $\mathbf{R}_{xy} = E[x_k y_k^*]$ and $\mathbf{R}_{yy} = E[y_k y_k^*]$ where $*$ denotes the conjugate transpose. Let $\mathbf{R}_{xx}$ and $\mathbf{R}_{nn}$ be the autocorrelation matrices of the channel input vector and noise vector respectively. ($\mathbf{R}_{xx}$ and $\mathbf{R}_{nn}$ are positive definite Toeplitz matrices.) Equation (2.2) implies

$$\mathbf{R}_{xy} = \mathbf{R}_{xx} \mathbf{H}^T \quad (2.3)$$

$$\mathbf{R}_{yy} = \mathbf{H} \mathbf{R}_{xx} \mathbf{H}^T + \mathbf{R}_{nn}. \quad (2.4)$$
Given any choice of feedback filter $b$, the error between the DFE estimate $z_k$ and the input $x_{k-\Delta}$ is

$$e_k = x_{k-\Delta} - z_k = x_{k-\Delta} - \left( \sum_{m=0}^{N_f-1} w_m^* y_{k-m} - \sum_{m=1}^{N_b} b_m^* x_{k-\Delta-m} \right) = b^* x_{k-\Delta} - w^* y_k$$

where $x_{k-\Delta} = [x_{k-\Delta} x_{k-\Delta-1} \cdots x_{k-\Delta-N_f}]^T$.

Note that the optimum feedback filter of length $\nu + N_f - \Delta - 1$ performs as well as any longer filter due to the finite lengths of the channel and the feedforward filter. Therefore we may specify that the feedback filter length satisfies $N_b \leq \nu + N_f - \Delta - 1$, so that the dimension of the above equation always makes sense. The optimal feedforward filter minimizes $\sigma^2 = E\{e_k^2\}$. The standard least-squares approach [10] shows that by the orthogonality principle, the MMSE is

$$\sigma^2 = b^T R_{\Delta} b,$$

where $R_{\Delta} = I_\Delta^T R_{x|y} I_\Delta$, $R_{x|y} = R_{xx} - R_{xy} R_{yy}^{-1} R_{xy}^*$ and

$$I_\Delta = \begin{bmatrix} 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}^T.$$

The position of the upper left 1 is determined by $\Delta$. Thus, $R_{\Delta}$ is the $(N_b+1)$ by $(N_b+1)$ subblock of $R_{x|y}$ whose upper left element corresponds to the $(\Delta+1)$ diagonal element of $R_{x|y}$.

Equations (2.3) and (2.4), along with the matrix inversion lemma\(^4\), yield the following expression for $R_{x|y}$ as a function of the channel parameters:

$$R_{x|y} = \left( R_{xx}^{-1} + H^T R_{mn}^{-1} H \right)^{-1}.$$

\(^4\)If $A$ and $C$ are nonsingular and $A$, $B$, $C$ and $D$ have consistent dimensions, then

$$(A + BCD)^{-1} = A^{-1} - A^{-1}B(DA^{-1}B + C^{-1})^{-1}DA^{-1}.$$
Then, it can be shown [11, 12] that the solutions of the optimum DFE are

\[ b = \frac{R_{\Delta}^{-1} e_0}{e_0^T R_{\Delta}^{-1} e_0} \]  

(2.7)

and

\[ w = R_{yy}^{-1} R_{yx} I_\Delta b \]  

(2.8)

where \( e_0 = [1 \ 0 \cdots 0]^T \).

Also, with the optimum \( b \), the MMSE becomes

\[ \sigma^2 = \frac{1}{e_0^T R_{\Delta}^{-1} e_0} \].

(2.9)

Along with the algebraic manipulation, it can be shown that the MMSE corresponds to the Schur complement of \( R_{\Delta} \).

In summary, the optimum finite-length MMSE-DFE is computed as follows. First obtain the channel matrix \( H \). Then obtain the matrix \( R_{x|y} \) in (2.6) and compute the Schur complement of the submatrix \( R_{\Delta} \). After the optimum \( \Delta \) that has the minimum Schur complement is attained, the corresponding DFE filters are computed from equations (2.7) and (2.8).

Our derivation of the finite-length MMSE-DFE yields an algebraic interpretation of the effect of decision delay on DFE performance. We identify MSE as a Schur complement in a submatrix \( R_{\Delta} \) of \( R_{x|y} \). The square matrix \( R_{x|y} \) has dimension \( \nu + N_f \), where \( \nu \) is the length of the channel taps and \( N_f \) is the number of feedforward DFE taps. Furthermore, \( R_{x|y} \) depends only on the channel and on \( N_f \) and is independent of \( N_b \) and \( \Delta \). The square matrix \( R_{\Delta} \) has size \( N_b + 1 \), and \( \Delta \) determines its position in \( R_{x|y} \).\(^5\)

The structure of our solution yields a method that efficiently computes the optimal DFE filter coefficients and corresponding MSE for several different values of \( \Delta \) and \( N_b \). The matrix \( R_{x|y} \) does not depend on \( \Delta \) or \( N_b \). Therefore, one need only compute \( R_{x|y} \) once for a given channel and feedforward filter length. Minimum MSE for various values of \( \Delta \) and \( N_b \) can then be computed as a Schur complement in \( R_{\Delta} \). Since

\(^5\)Belfiore and Park derived an expression for the optimal submatrix in the case of an infinite-length feedforward filter [13]. In this case, the submatrix does not depend on \( \Delta \).
Figure 2.2: The effect of decision delay

Each submatrix $R_\Delta$ overlaps the adjacent submatrices, computation of the Schur complement in each $R_\Delta$ can be reduced to some extent. Our method is especially efficient in the case of a long channel and a short feedback section ($\nu \gg N_b$), which is exactly the case where the choice of $\Delta$ and $N_b$ has the most significant effect on performance of the DFE.

Examples and Interpretation

This section presents a pair of numerical examples that demonstrate the effects of decision delay on DFE performance.

The first example is motivated by the application of decision feedback equalization to digital magnetic recording system [14]. Digital equalization methods are becoming increasingly popular in disk drive products. High data rates and low power specifications force the use of FIR equalizer filters with a small number of taps. As a result, the choice of decision delay can have a significant impact on performance of the DFE.

The digital magnetic recording process can be approximated by a linear channel. A typical channel response (known as the Lorentzian response) is given by the pair
of equations

\[ h_k = s_k - s_{k-1} \]

\[ s_k = \frac{1}{1 + k^2}. \]

For computational purposes we truncate the channel response so that \( \nu = 40 \). The channel input \( \{x_k\} \) is an independent, identically distributed (i.i.d.) sequence, and the additive noise is stationary and white with zero mean. The input and noise sequences are scaled so that the channel \textit{signal-to-noise ratio}, defined as

\[ \text{SNR}_{\text{chan}} = 10 \log \left( \frac{E[x_k^2] \sum h_k^2}{E[n_k^2]} \right), \]

equals 25 dB. We consider a DFE with eight feedforward \((N_f = 8)\) and four feedback \((N_b = 4)\) taps.

For convenience we measure DFE performance in terms of the \textit{equalizer signal-to-noise ratio}

\[ \text{SNR}_{\text{DFE}} = 10 \log \left( \frac{E[x_k^2]}{\sigma^2} \right), \]
where $\sigma^2$ is the MSE at the input to the threshold device. Figure 2.2 shows $\text{SNR}_{\text{DFE}}$ versus decision delay, for values of $\Delta$ in the interval $[0, N_f - 1]$. Note the the optimal decision delay is $\Delta = 3$, and that nonoptimal choice of $\Delta$ can result in a significant performance loss.

The second example deals with the effect of decision delay as filter lengths increase. In the infinite-length case, the optimum feedforward filter consists of a matched filter in cascade with an anticausal component [4]. This optimum filter structure is called “whitened matched filter” [15] under the Zero Forcing criterion and acts as an allpass filter changing the phase of the channel output. If the channel $\{h_k\}$ is causal and the noise $\{n_k\}$ is white, then the matched filter is anticausal, and so is the feedforward filter. It can be shown that under the conditions of a causal channel and white noise, the optimal finite-length feedforward filter is anticausal, as long as there are enough feedback taps. Our example demonstrates this property.

We consider the causal channel

$$h_k = \begin{cases} (k + 1)e^{-(k+1)} & k \geq 0 \\ 0 & k < 0 \end{cases}$$

We truncate the channel response so that $\nu = 20$. As before, the channel input is an i.i.d. sequence, and the noise is stationary and white with $\text{SNR}_{\text{chan}} = 20$ dB. We assume the number of feedforward taps is fixed at $N_f = 10$.

Figure 2.3 shows the optimal decision delay (defined as that which maximizes $\text{SNR}_{\text{DFE}}$) versus the number of feedback taps. When the number of feedback taps is small, the optimal feedforward filter is two-sided; that is, it contains both causal and anticausal taps. As expected, however, the optimal $\Delta$ converges to $N_f - 1$ as the number of feedback taps get large. Note that when $\Delta = N_f - 1$, the feedforward filter is anticausal; that is, $\{w_k\} = \{w_{-\Delta}, \ldots, w_0\}$.

### 2.2.2 MMSE-LE

The Minimum Mean Square Error Linear Equalizer (MMSE-LE) consists of a feedforward filter $w$. Thus, the MMSE-LE can be treated as a special case of the MMSE-DFE with no feedback filter. The MMSE-LE is designed to compromise a
reduction in ISI with noise enhancement, which is often encountered in a zero-forcing case. The structure of the LE appears in Figure 2.4. The MMSE-LE is a linear filter $w$ that acts on the channel output $y_k$ to form an output sequence $z_k$, which is the best MMSE estimate of $x_k$. Thus, we choose the filter $w_k$ to minimize the mean square error. The optimum MMSE-LE coefficients can be attained from the previous section with $N_b = 0$.

Assuming the input sequence $\{x_k\}$ is uncorrelated, the optimum MMSE-LE setting and the corresponding MSE are

$$w = (\tilde{E}_x H H^T + \frac{N_0}{2} I)^{-1} \tilde{E}_x H e_\Delta$$

and

$$\sigma^2 = e^{T}_\Delta R_{x|y} e_\Delta$$

where $e_\Delta$ is a unit column vector with 1 in a $\Delta$-th position.

Therefore, the optimum decision delay $\Delta$ is the position of the minimum diagonal element of $R_{x|y}$. In contrast to the DFE case, the optimum decision delay in the LE has a straightforward interpretation: Performance of the LE is optimized when $\Delta$ is chosen such that the element of the autocorrelation matrix corresponding to the MMSE vector in estimating $x_{k-\Delta}$ from $\{y_k\}$ is minimized.

### 2.2.3 PRML

Figure 2.5 shows the structure of the Partial Response Maximum Likelihood receiver (PRML). The PRML uses the linear equalizer $w$ to shape the channel to the fixed partial response $b$, and then the Viterbi detector estimates the input sequence
matched to the partial response \( b \). PRML has been a popular choice among detectors especially in a magnetic recording system. Similarly, the optimum feedforward filter \( w \) and the MSE can be attained as

\[
w = R_{yy}^{-1} R_{yx} b
\]

and

\[
\sigma^2 = b^* R_{\Delta} b
\]

where \( b \) is the fixed target response that the Viterbi detector is matched to.

One of the problems in PRML is that the error sequence in the Viterbi detector is correlated. Thus, the Viterbi detector designed for the additive white noise sequence deteriorates performance of the PRML. Another form of degradation can take place when the equalization to the target response is not perfect. This channel misequalization effect is no longer negligible when the feedforward filter is not long enough to cancel all the ISI.

Now we will analyze performance of the PRML, taking into account the effects of colored noise and the channel misequalization. Appendix A shows derivation of the effective SNR of the Maximum Likelihood Sequence Detector (MLSD) in the presence of misequalization and colored noise. Following the notations in the Appendix A, it can be shown that the effective SNR of the PRML is the squared value of the argument of \( Q \) function in the probability of error:

\[
P_e \approx Q \left( \min_{x, \hat{x}} \frac{<e, \hat{y} + \hat{y} - 2y>}{2\sqrt{e^T R_e e}} \right),
\]

where

\[
Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-z^2/2} \, dz,
\]
and $\langle \cdot, \cdot \rangle$ represents vector inner product.

Exhaustive search has to be carried out over all possible input sequence pairs $x, \hat{x}$ to find the effective SNR of the PRML detector.

2.3 Summary

This chapter has discussed three receivers which are widely used in digital transmission and storage systems. For each receiver, the optimal coefficients are solved in finite length case and the effective SNR is derived. These receivers will be compared with other new receiver structures throughout this dissertation.
Chapter 3

A Fast Computation Algorithm for the Decision Feedback Equalizer

A novel fast algorithm for computing the Decision Feedback Equalizer settings is proposed. The equalizer filters are computed indirectly, first by estimating the channel, and then by computing the coefficients in the frequency domain with the Discrete Fourier Transform (DFT). Approximating the correlation matrices by circulant matrices facilitates the whole computation with very small performance loss. The fractionally spaced equalizer settings are derived. The performance of the fast algorithm is evaluated through simulation. The effects of the channel estimation error and finite precision arithmetic are briefly analyzed. Results of simulation show the superiority of the proposed scheme.

3.1 Introduction

The Decision Feedback Equalizer (DFE) is a well-known receiver structure that is used to mitigate intersymbol interference (ISI) in communication channels. The DFE receiver structure has received considerable attention from many researchers because its performance is superior to the linear equalizer. It decodes channel inputs on a symbol-by-symbol basis and uses previous decisions to subtract trailing ISI. The minimum-mean-square-error decision feedback equalizer (MMSE-DFE) optimizes the
feedforward and feedback filter to minimize the mean-square error [16, 13, 4].

Adaptive equalization is essential when transmitting data over an unknown channel. The equalizer coefficients of the adaptive DFE can be adjusted recursively [17]. However, the convergence of the time-domain LMS adaptation algorithm is slow, due to the high eigenvalue spread inherent in the DFE structure. The recursive LMS adaptation of the DFE may take millions of iterations to converge to the optimal solution on a typical twisted copper wire loop, for example [18]. The eigenvalue spread in the DFE is very high, because the feedforward and feedback filters are associated with each other as shown in equation (2.8). When the eigenvalues of the input correlation matrix are widely spread, the settling time of the steepest-descent algorithm is limited by the smallest eigenvalues of the slowest modes resulting in the slow convergence. This problem has motivated an interest in non-recursive adaptive methods based on channel-estimates, in contrast to the traditional recursive adaptive methods for computing the equalizer settings [19, 20].

Non-recursive adaptive equalization can be decomposed into two stages: first estimate the channel, then compute the non-recursive DFE coefficients based on the channel estimates. This non-recursive approach has been shown to have a performance advantage over the direct adaptive equalization method [20, 21]. Compared to the recursive adaptive algorithm, the non-recursive equalization technique does not suffer from the high eigenvalue spread problem. Also, in a time-varying channel, the channel-estimate-based DFE is reported to have better tolerance in tracking to time variation than RLS adaptive DFE [22].

This chapter presents a novel technique to compute the MMSE-DFE coefficients very efficiently, based on channel estimates. The proposed algorithm requires knowledge of the channel pulse response, along with the noise variance. It is shown in the simulation, however, that the exact value of the noise variance is not important. The channel identification problem, on which the proposed fast algorithm is based, will not be pursued in this chapter. Simple methods include cross-correlating channel output samples with a known broadband training sequence [23]. Very accurate and efficient channel identification can also be carried out in the frequency domain using the Fast Fourier Transform (FFT) with special periodic training sequences [24]. Designs of
other training sequences for channel estimates are discussed in [20, 25, 26].

Fast computation of the MMSE-DFE coefficients exploiting structured matrices was derived in [19]. The authors in [19] described an algorithm for computing the feedback filter with $O(M\nu)$ operations where $M$ is the length of the feedforward filter and $\nu$ is the length of the channel pulse response. However, $O(M^2)$ operations are still required for obtaining the feedforward filter, and this method involves the fast Cholesky factorization algorithm which makes practical implementation challenging. Also, the number of feedback taps is restricted to $\nu$, which is not a practical assumption.

This chapter proposes a fast algorithm to compute the MMSE-DFE settings using only the Discrete Fourier Transform (DFT) and the inverse Discrete Fourier Transform (IDFT), so that the whole computation can be carried out with $O(M \log_2 M)$ operations, when the Fast Fourier Transform (FFT) is used. Direct matrix inversion can be avoided by approximating the correlation matrix by a circulant matrix. The proposed algorithm parallels a similar "cyclic equalization" technique applied to the linear equalizer, which uses a periodic pseudo-random training sequence [17, 26, 27, 28, 29, 30]. The proposed fast algorithm can be implemented very efficiently in practice. Simulation shows that the proposed method comes to within a few tenths of a dB in terms of minimum mean square error performance. Thus, the proposed non-recursive algorithm exhibits superiority in computational efficiency over the conventional recursive adaptive method. The non-recursive algorithm combined with the proposed fast algorithm is promising where a fast start-up is crucial.

In Section 3.2, we describe the proposed algorithm for the fractionally spaced MMSE-DFE. Section 3.3 briefly explains the channel identification method. Then, the performance of the fast algorithm is examined in the symbol spaced equalizer and the effects of channel estimation error as well as finite precision arithmetic are discussed in Section 3.4. The finite precision simulation shows that the proposed fast algorithm does not require high precision arithmetic, thus reducing complexity in real implementation. Finally Section 3.5 contains a concluding discussion.
3.2 Fast Equalization Algorithm

The structure of the MMSE-DFE is described in Section 2.2.1. The solutions of the optimum DFE are

\[ \mathbf{b} = \frac{1}{k} \mathbf{R}^{-1}_{x|y} \mathbf{e}_0 \]
\[ \mathbf{w} = \mathbf{R}^{-1}_{yy} \mathbf{R}_{yz} \mathbf{b} \]

where \( k \) is the (1,1) element of \( \mathbf{R}^{-1}_{x|y} \) and \( \mathbf{e}_0 = [1 \ 0 \ \cdots \ 0]^T \) and

\[ \mathbf{R}_{x|y} = \mathbf{R}_{xx} - \mathbf{R}_{xy} \mathbf{R}^{-1}_{yy} \mathbf{R}_{yz} \]  \hspace{1cm} (3.1)

where \( \mathbf{R}_{xx} = E[\mathbf{x}_{k-\Delta} \mathbf{x}_{k-\Delta}^*] \), \( \mathbf{R}_{xy} = E[\mathbf{x}_{k-\Delta} \mathbf{y}_k^*] \) and \( \mathbf{x}_{k-\Delta} = [x_{k-\Delta} x_{k-\Delta-1} \cdots x_{k-\Delta-N_f}]^T \).

We begin with a brief notational description. A plain lowercase variable denotes a scalar quantity, while a bold lowercase variable denotes a vector. A DFT column vector is represented by a plain uppercase variable, while a matrix is represented by a bold uppercase variable. The symbol \( \mathbf{I}_n \) represents the \( n \) by \( n \) identity matrix, and the symbol \( \mathbf{1}_n \) indicates an all one column vector of length \( n \).

First we describe a circulant matrix. A circulant matrix has the Discrete Fourier Transform basis vectors as its eigenvectors, and the Discrete Fourier Transform of its first column as its eigenvalues. Defining \( M' = M \cdot l \), an \( M' \) by \( M' \) circulant matrix can be decomposed as

\[
\begin{bmatrix}
  c_0 & c_{M'-1} & c_{M'-2} & \cdots & c_1 \\
  c_1 & c_0 & c_{M'-1} & & \vdots \\
  c_2 & c_1 & c_0 & & \ddots \\
  \vdots & \vdots & \ddots & \ddots & \\
  c_{M'-1} & \cdots & c_0 \\
\end{bmatrix}
= \frac{1}{M'} \mathbf{P^*}
\begin{bmatrix}
  C_0 & 0 \\
  C_1 & \ddots \\
  0 & \ddots & \ddots \\
  & & 0 & C_{M'-1} \\
\end{bmatrix}
\mathbf{P}
\]

where \( \mathbf{P} \) is the \( M' \) by \( M' \) discrete Fourier transform (DFT) matrix with

\[ p_{m,n} = e^{-j2\pi mn/M'}, \ 0 \leq m, n \leq M' - 1, \]

and the column vector \( \mathbf{C} = [C_0 \ C_1 \ \cdots \ C_{M'-1}]^T \) is the \( M' \)-point DFT of \([c_0, c_1, \cdots c_{M'-1}]\), the first column of the circulant matrix.
Chapter 3. A Fast Computation Algorithm for the Decision Feedback Equalizer

We will define $P_M$ as the $M'$ by $M$ submatrix of $P$ consisting of every $l$ th column of $P$ up to $(M-1)l$ th column: $P_M = [p_0 \ p_l \ p_{2l} \ \cdots \ p_{(M-1)l}]$ where $p_i$ is the $i$ th column of $P$. Similarly, we define $P_N$ as the $M'$ by $N+1$ submatrix of $P$ consisting of every $l$ th column of $P$ up to $Nl$ th column: $P_N = [p_0 \ p_l \ p_{2l} \ \cdots \ p_{Nl}]$. We note that premultiplying a column vector $y$ of length $M'$ by $P$ yields its DFT column vector, $Y = Py$, and premultiplying a column vector $x$ of length $N+1 \ (\leq M)$ by $P_N$ produces its zero-padded $M'$-point DFT column vector, since

$$P_Nx = P[x_0 \ 0 \ \cdots \ 0 \ x_1 \ 0 \ \cdots \ 0]^T = [X^T \ \cdots \ X^T]^T$$

where the column vector $X$ is the $M$-point DFT of $[x_0 \ x_1 \ \cdots \ x_N \ 0 \ \cdots \ 0]^T$. Similarly, $P_N^*Y$ represents the truncated IDFT vector $y$ since $P_N^*Y = [y_0 \ y_l \ y_{2l} \ \cdots \ y_{Nl}]^T$.

Note that from the orthogonal property of the DFT basis function $P^*P = PP^* = M'I_{M'}$ and $P_N^*P_N = M'I_{N+1}$, but $P_NP_N^* \neq M'I_N$. Also it can be shown that

$$P_M^*P_M = M[I_M \ \cdots \ I_M]^T[I_M \ \cdots \ I_M]. \quad (3.2)$$

It is convenient to denote $\Lambda_Z$ as the diagonal matrix whose diagonal element consists of the elements of a column vector $Z$. Here we assume that $Z$ is a DFT column vector. Using $P^{-1} = \frac{1}{M'}P^*$, the inverse of the above circulant matrix is $\frac{1}{M'}P^*\Lambda_Z^{-1}P$.

The main innovation of the fast startup equalization in this chapter arises from the approximation of a Toeplitz correlation matrix by a circulant matrix. A circulant matrix is asymptotically equivalent to a Toeplitz matrix [31]. By making this approximation, most computations can be implemented with the discrete Fourier transform and inverse discrete Fourier transform very efficiently.

Our derivation assumes that the length of the feedforward filter, $w$, exceeds that of the feedback filter $b \ (M' \geq N+1)$. This restriction may not be applicable in some channels such as the subscriber loop, where the feedback filter can be longer than the feedforward filter. However, a long feedforward filter can be accurately approximated by a pole-zero filter with fewer coefficients using a computationally-efficient algorithm described in [32] with little performance loss.

As derived in Appendix B, the fractionally spaced MMSE-DFE coefficients can be attained as follows.
Chapter 3. A Fast Computation Algorithm for the Decision Feedback Equalizer

Define the length-$M$ column vector $\Theta = \bar{e}_x M \sum_{i=1}^{i=M'} (\|H^i\|^2) + M' l_2^2 1_M$ where the length-$M$ column vector $H^i$ denotes the $i$th sub-vector of the $M'$ point DFT channel-response vector $H = [H_1^T H_2^T \cdots H_{M'}^T]^T$. Here $\| \cdot \|^2$ is defined as the element-wise norm square: $\| [a_0 a_1 \cdots a_{M-1}]^T \|^2 = [ |a_0|^2 |a_1|^2 \cdots |a_{M-1}|^2 ]^T$.

Then, defining $\tilde{B}_i$ as
\[
\tilde{B}_i = \frac{1}{M'} \Theta_i \mod M, \quad i = 0, 1 \cdots M' - 1
\]
(3.3)

where $\Theta_k$ indicates the $k$th element of the vector $\Theta$, the feedback filter can be obtained from the IDFT operation in equation (B.3):
\[
b_k = \frac{1}{M'} \sum_{i=0}^{M'-1} \tilde{B}_i e^{j 2 \pi k / M}, \quad k = 0, 1, \cdots N.
\]
(3.4)

Also, defining
\[
W_i = \frac{\bar{e}_x M H_i^* P_{\Delta_i} B_i \mod M}{\Theta_i \mod M}, \quad i = 0, 1 \cdots M' - 1
\]
(3.5)

where $H_i$ and $B_i$ are the $i$th element of column vectors $H$ and $B$ respectively, $P_{\Delta} = [\bar{P}_\Delta \cdots \bar{P}_\Delta]^T$ and $\bar{P}_\Delta = [1 e^{-j 2 \pi \Delta / M} e^{-j 2 \pi 2 \Delta / M} \cdots e^{-j 2 \pi (M-1) \Delta / M}]^T$, we can obtain $w_k$ using the IDFT:
\[
w_k = \frac{1}{M'} \sum_{i=0}^{M'-1} W_i e^{j 2 \pi k / M'}, \quad k = 0, 1 \cdots M' - 1.
\]

We notice that the whole computation of $b$ and $w$ can be performed using only the DFT and the IDFT with no matrix inversion and/or multiplication.

It is interesting that $b$ is independent of the decision delay, $\Delta$, which makes sense because $\bar{R}_{x|y}$ is now Toeplitz. Also, we do not need to compute $w$ for every $\Delta$. Once $w$ is computed for $\Delta = 0$, $w$ with other values for $\Delta$ can be obtained by a circular shift by the amount $\Delta$.

We note that cyclic equalization for the linear equalizer is a special case of the proposed algorithm. With $b = [1 0 \cdots 0]$, equation (3.5) becomes the similar equation derived in [17, 28].

When $M'$ is an integer power of 2, the DFTs are carried out by the Fast Fourier Transform to speed up the computation. An FFT can be computed very efficiently in $O(M' \log_2 M')$ operations.
Because the asymptotic behavior of eigenvalues of Toeplitz matrices approaches that of circulant matrices as the size increases, the proposed algorithm is expected to converge to the optimal solution as \( M' \) increases. This observation will be confirmed in the following section through simulations.

One choice of the training sequence in equation (B.2) is a pseudo-random noise sequence. Another attractive method is a chirp sequence since this has a flat power spectrum and low peak-to-average power ratio [33]. Thus we can describe a chirp sequence for channel identification by

\[
x_k = e^{j \frac{2\pi}{M} k^2} \quad k = 0, 1, \ldots, M - 1
\]

For multi-rate systems such as V.34, a chirp sequence is used for probing to also determine the maximum baud rate [34]. With a chirp sequence, the baud rate and the DFE coefficients can be estimated simultaneously.

Simulation results show that a better solution can be obtained by introducing a scaling factor \( \alpha \) to calibrate \( \mathbf{b} \). After \( \mathbf{b} \) is computed from the previous procedures, \( \mathbf{b} \) scaled by scalar \( \alpha \) can be used instead to compensate for the approximation. The mean square error \( \sigma_{DFE}^2 \) taking into account the approximations after calibration is then

\[
\sigma_{DFE}^2 = (1 \alpha \mathbf{b}^*) \mathbf{R}_{x|y} \begin{bmatrix} 1 \\ \alpha \mathbf{b} \end{bmatrix}
\]

where \( \mathbf{b} = [b_1 b_2 \cdots b_N]^T \) and \( \mathbf{R}_{x|y} = \mathbf{R}_{xz} - (2 \mathbf{R}_{xy} - \mathbf{R}_{xy} \mathbf{R}_{yy}^{-1} \mathbf{R}_{yy}) \mathbf{R}_{yy}^{-1} \mathbf{R}_{yz} \). Here \( \mathbf{R}_{yy} \) and \( \mathbf{R}_{xy} \) are computed from the input/output relationship in equation (2.2).

If we partition \( \mathbf{R}_{x|y} \) into

\[
\mathbf{R}_{x|y} = \begin{bmatrix} c & \mathbf{e}^* \\ d & A \end{bmatrix}
\]

The optimal \( \alpha \) that minimizes the mean square error can be found by differentiating equation (3.6) with respect to \( \alpha \):

\[
\alpha = -\frac{\mathbf{b}^* d + \mathbf{e}^* \mathbf{b}}{2 \mathbf{b}^* A \mathbf{b}}.
\]

Still we do not need to invert a matrix to compute \( \alpha \). Then \( 1 \alpha \mathbf{b}^* \) is used instead of \( \mathbf{b}^* \). A simpler scaling constant \( \alpha_0 \) can be derived if \( \mathbf{R}_{x|y} \) of equation (B.1) is used
Chapter 3. A Fast Computation Algorithm for the Decision Feedback Equalizer

instead of \( \hat{R}_{x|y} \) in equation (3.6). Using the same partition of \( \hat{R}_{x|y} \) as in equation (3.7), it can be shown that

\[
\alpha_0 = -\frac{\tilde{b}^*d}{\tilde{b}^*A\tilde{b}} = -\frac{\tilde{b}^*d}{\sum_{i=0}^{M-1}(|\tilde{B}_i|^2/\tilde{B}_i)}
\]

where \( d^* = [d_1 \; d_2 \; \cdots \; d_N] \), \( d_k = \frac{1}{M'} \sum_{i=0}^{M'-1} \frac{1}{\tilde{B}_i} e^{j2\pi ik/M} \), \( \tilde{B}_i \) is defined in equation (3.3), and \( \tilde{B}_i \) is the \( i \) th DFT element of \([0 \; \cdots \; 0 \; b_1 \; 0 \; \cdots \; 0 \; b_2 \; \cdots \; b_N \; 0 \; \cdots \; 0]\).

Compared with \( \alpha \), \( \alpha_0 \) can be obtained with only DFT operations with little expense of performance. Typical values of \( \alpha \) and \( \alpha_0 \) range from 1.1 to 1.8. Simple inspection of equation (3.3) will reveal that scaling \( b \) by \( \alpha \) has the same effect as having scaled \( \sigma_n^2 \). This indicates that the perfect knowledge of \( \sigma_n^2 \) is not required. Comparison between using \( \alpha \) and \( \alpha_0 \) is made in the following section.

In summary, the fast equalization for the symbol-spaced MMSE-DFE \( (l = 1) \) is carried out as follows:

1. Compute \( \tilde{e}_x|H_i|^2 + \sigma_n^2 \) from the channel estimate.

2. Compute \( b \) using the IDFT as described in equation (3.4). The scaling factor \( \alpha \) or \( \alpha_0 \) can be used to obtain a better solution.

3. Using \( b \) computed in the above step, perform the IDFT of \( \frac{\tilde{e}_x H_i^* P_{\Delta x} B_i}{\tilde{e}_x|H_i|^2 + \sigma_n^2} \) to get \( w \) as in equation (3.5).

The flow chart in Figure 3.1 describes the fast algorithm. The solution for the symbol-spaced equalizer with correlated noise is derived in the following chapter.

### 3.3 Channel Identification

In this section, we briefly analyze the channel identification method that the proposed fast algorithm can adopt. There are several schemes that carry out the channel identification efficiently.
Estimated the channel impulse response

\[
\bar{B}_i = \frac{1}{k'} (\bar{\varepsilon}_x |H_i|^2 + \sigma_n^2)
\]

Obtain the feedback filter b using (3.4)

Scale the feedback filter to maximize SNR

Using the feedback filter, compute

\[
W_i = \frac{\bar{\varepsilon}_x H_i^* p_{\Delta, i} B_i}{\bar{\varepsilon}_x |H_i|^2 + \sigma_n^2}
\]

Obtain the feedforward filter w using (3.5)

Figure 3.1: Flow chart of the fast algorithm
Channel identification methods measure the channel pulse response assuming the noise spectrum is flat. The channel noise is an undesired disturbance in channel pulse response estimation, and this noise needs to be averaged in determining the channel response.

One easy method is to use a periodic training sequence \( x_k \) with period \( M \), equal to or slightly longer than the length of the channel pulse response. The receiver measures the corresponding channel output averaging over \( L \) cycles, and then divides the DFT of the channel output by the DFT of the known training sequence. The channel estimate in the frequency domain is

\[
\hat{H}_n = \frac{1}{L} \sum_{i=1}^{L} \frac{Y_{i,n}}{X_n}
\]

where \( Y_{i,n} \) is the \( n \)th element of the DFT of the channel output on \( i \)th cycle and \( X_n \) is the \( n \)th element of the DFT of the input training sequence.

It can be shown [35] that the overall MSE of the estimation error is equal to \((1 + \frac{1}{L})\sigma_n^2\). The excess MSE then has variance \( \frac{1}{L} \sigma_n^2 \) that is reduced by a factor equal to the number of averaging. This means that performance of this channel identification scheme improves linearly with \( L \) and is independent of the eigenvalue spread that often deteriorates performance of the recursive adaptive algorithm.

For example, when \( L = 40 \), we can tolerate only \( 1 + \frac{1}{40} = 0.1 \) dB excess error in channel pulse response estimation. Even considering the worst channel pulse response in typical subscriber loops (in terms of the length of the channel memory), with total 5000 sample periods an accurate pulse response estimation can be carried out resulting in less than 0.1 dB excess MSE.

Combining this channel response estimation scheme, the proposed fast algorithm still produce the equalizer coefficients very efficiently in contrast to the traditional recursive adaptive algorithm. In the following section, simulation results using the fast algorithm are given.

### 3.4 Simulations
Chapter 3. A Fast Computation Algorithm for the Decision Feedback Equalizer 34

Figure 3.2: Pulse responses of the Lorentzian model

In this section, we show the simulation results performed by the proposed fast algorithm described in the previous section. Performance is computed assuming that the channel pulse response and the noise variance are given. Two channels are used in these simulations. The first channel is a magnetic recording channel modelled using a Lorentzian step response

\[ s(t) = \frac{1}{1 + (2t/pw_{50})^2} \]

where \( pw_{50} \) is the width of the pulse at 50% of its peak value. The pulse response, \( h_k \), in equation (2.1) is obtained from \( s(t) \) using \( h_k = s_k - s_{k-1} \) where \( s_k = s(kT) \). Figure 3.2 shows the pulse responses of the Lorentzian model with several recording densities.

The channel pulse response with \( \frac{pw_{50}}{2} = 1 \) is used in the simulation. The simulation result is shown in Figure 3.3. In this simulation, a feedback filter with 6 taps is used, and the feedforward filter taps are changed as indicated in \( x \)-axis. Decision delay, \( \Delta \), is optimized to get the best SNR, and \( \text{SNR}_{\text{MFB}} = \frac{||h||^2\xi}{\sigma_n^2} \) is set to 15 dB. The solid line represents the SNR of the DFE computed from the optimal \( b \) and \( w \), and the dashed line indicates the SNR computed from the fast algorithm with \( b \) scaled
Figure 3.3: Performance of the fast algorithm in Lorentzian channel with $N = 6$.
Figure 3.4: Performance of the fast algorithm in ADSL channel with $N = 25$
by $\alpha$, and the dotted line shows the SNR computed from the fast algorithm with $b$ calibrated by $\alpha_0$.

We next use a channel that models a 9 kft 26 AWG Asynchronous Digital Subscriber Loop (ADSL) sampled at 640 kHz. The result is shown in Figure 3.4. Similarly, the length of the feedback filter is set to 25 and the number of the feedforward filter taps is represented on the $x$-axis.

The gap between the optimal SNR and the SNR from the fast algorithm in both plots approaches zero as the number of feedforward filter taps increases, as expected. When $b$ is scaled by $\alpha_0$, performance is close to that scaled by $\alpha$. It is clear from both plots that the fast algorithm generates the equalizer settings approaching the optimal solution within a tenth of a dB.

While we do not address the channel estimation problem in this chapter, the effect of the channel estimation error is briefly analyzed in Figure 3.5. Again, the Lorentzian pulse response with $p w_{50}/T = 1$ is used with $M = 24$ and $N = 6$, and the optimized $\alpha$ is used when computing the proposed algorithm. The solid line represents the SNR of the optimal DFE with the perfect pulse response, and the
dashed line indicates the SNR of the proposed algorithm with the imperfect channel estimate. In this simulation, the channel response is estimated using the noniterative algorithm in [25] with the averaging factor equal to 10. This requires 80 symbol periods for the channel estimation. The plot shows that the proposed algorithm still yields performance within 1 dB of the optimum solution. As the number of averaging times increases, performance of the fast algorithm approaches the optimal case. Although a complete analysis of the channel mis-estimation effect is not carried out in this chapter, this would be an interesting research topic.

In Figure 3.6, the finite precision effect is investigated. The same channel response as in Figure 3.5 is used and the matched filter bound is set to 25 dB. The large eigenvalue spread for the DFE requires high precision arithmetic when direct matrix inversion is used to compute the DFE coefficients in equations (2.7) and (2.8). The plot shows the SNR difference between the infinite precision solution and the finite precision case. The plot indicates that the proposed algorithm is much less sensitive to the finite precision, compared to the solution using matrix inversion. This is not surprising since the direct matrix inversion is numerically sensitive. This can
be another huge benefit of the fast algorithm in real implementation because the proposed algorithm does not require high precision arithmetic.

3.5 Conclusion

We have proposed a novel fast algorithm for the minimum mean square error Decision Feedback Equalizer. Based on channel estimates, the fast algorithm computes the equalizer settings using the DFT and IDFT very efficiently. The overall computation can be carried out without a matrix operation with negligible performance loss as the number of the feedforward filter taps increases.

Simulations performed in a magnetic recording channel and ADSL channel show that the fast algorithm yields the near-optimal settings very efficiently. The proposed algorithm is shown to be robust to finite precision arithmetic compared to the direct matrix inversion solution.
Chapter 4

Performance of A Fast Algorithm in High-speed Subscriber Loop

The discrete multitone (DMT) modulation is considered to be a viable transmission scheme for High-speed Subscriber Loop. In this chapter, the fast algorithm for computing the decision feedback equalizer settings derived in the previous chapter is extended and applied for the DMT system in High-speed Subscriber Loop. Again, the channel pulse response is assumed to be given by the channel identification method, and then the equalizer filter settings are computed. In simulations, a fast algorithm for the symbol spaced equalizer in a colored noise channel is used. Simulation results performed in various CSA loops indicate that the fast algorithm yields the near-optimum settings for the DMT system.

4.1 Introduction

There has been growing interest in utilization of existing unshielded copper twisted pair for various applications including but not limited to ISDN, DSL, HDSL, and more recently, ADSL. The maximum achievable data rate on the carrier serving area (CSA) loops, containing approximately 80% of the lines between central offices (CO) and remote customer premises, has been improving and has exceeded 6 Mbps for ADSL application.
This chapter describes the Discrete Multitone (DMT) modulation in High-speed Subscriber Loop. DMT modulation has been chosen as the ANSI standard for data transmission for ADSL application.

The receiver of a DMT system consists of a time-domain equalizer (TEQ) that is an FIR filter and there are several methods to train the TEQ during startup, see [36] and [37]. However, these methods either involve intense computation such as matrix inversion [36], or require a significant amount of iterations and time before the algorithm reaches convergence [37]. The former method thus results in high complexity count and expensive cost, and the latter method may have limited applications, especially in areas where a fast training time is crucial.

In most practical situations, the channel response is not known a priori. Thus adaptation of the equalizer is essential when data is transmitted over an unknown channel. As discussed in the previous chapter, the conventional LMS algorithm of the TEQ settings as in [36] takes too long to converge to the optimal settings on a copper twisted pair loop channel. Figure 4.1 shows a learning curve of the TEQ on 400m 26
Chapter 4. Performance of A Fast Algorithm in High-speed Subscriber Loop

Figure 4.2: Impulse response of 400m 26 gauge loop

gauge copper loop channel using the LMS algorithm with 64 feedforward taps and 25 feedback taps. The solid line at the bottom of the figure indicates the minimum mean square error (MMSE). Even after 10,000 iterations, the mean square error of the LMS algorithm is still 3 dB away from the MMSE value. Figure 4.2 represents the impulse response of 400m 26 gauge copper loop.

The slow convergence motivates study of non-iterative methods to compute the optimum equalizer settings. Thus, in contrast to the conventional recursive adaptive technique, the previous chapter proposed a fast computation algorithm for the Decision Feedback Equalizer (DFE) that had a similar structure to the TEQ. Exploiting this structural similarity, this chapter extends the non-recursive fast algorithm for the DFE to the TEQ case and examines performance of the non-recursive fast algorithm in high-speed subscriber loop.

The fast algorithm is attractive especially in the DMT scheme since the Fast Fourier Transform (FFT) operation embedded in the DMT system can be shared to generate the solution of the fast algorithm without additional hardware cost.

Again, the channel pulse response is assumed to be given using some well-known
channel identification techniques. This identification can be done in a few thousand iterations and still takes much shorter time in comparison with the total convergence time of the conventional LMS algorithm. When noise is no longer white, the noise autocorrelation function can also be estimated with a little longer training sequences.

We will show that the proposed fast algorithm computes the TEQ coefficients with much reduced complexity compared with the method in [36] and obtains the results with a much shorter time period than the method described in [37] even after taking into account the process of channel identification.

In Section 4.2\(^1\), the DMT structure is explained and the TEQ is described in more detail. Section 4.3 briefly shows the optimum TEQ settings [36]. The only difference between the TEQ and the DFE is that the minimum phase constraint on the feedback filter is not required in the TEQ. Thus, the DFE is a special case of the TEQ solution with a minimum-phase constraint on the feedback filter. The computation of the TEQ coefficients can be easily extended from the DFE case by changing the position of a unit center tap in the feedback filter. (In the DFE, the position of a unit center tap is set to the first tap.)

In Section 4.4, we derive the fast algorithm in a colored noise case. Section 4.5 shows the simulation results for various copper loops and impairments.

### 4.2 Discrete Multitone (DMT) Modulation

Discrete multitone (DMT) modulation divides a channel into a set of parallel independent subchannels. The SNR of each subchannel is measured and an appropriate amount of information or bits is then assigned to each subchannel. As a result, the DMT system optimizes performance for any channel that it encounters, by virtue of the adaptive, selective bit/information allocation to each of the independent subchannels. Figure 4.3 shows the basic structure of DMT systems. At the transmitter, the data is modulated by a set of parallel, independent subchannels, \( p_i, i = 0, \ldots, N_c - 1 \), where \( N_c \) is the number of independent subchannels. The \( p_i \)'s divide up the channel and modulate the input data, \( x_i, i = 0, \ldots, N_c - 1 \). The modulated signal is then

\(^1\)We would like to thank Jacky Chow of Amati Corporation for helpful discussions.
summed together before it is sent to the channel, $h_k$. At the receiver, the received signal is demodulated by the corresponding set of demodulating vectors that separate the received signal into a set of parallel, independent subchannels.

A specific implementation of the DMT system is shown in Figures 4.4 and 4.5. The modulation and demodulation techniques we used are the Inverse Fast Fourier transform (IFFT) and the Fast Fourier transform (FFT), respectively. Efficient algorithms for IFFT and FFT are well-known and thus significantly reduce the complexity of implementing the modulation and demodulation functions.

The binary input data are parsed onto a set of parallel, independent subchannels, each of which is assigned a fixed number of bits during startup or system initialization. Given the measured SNR of each subchannel during startup, the number of bits for each subchannel is then determined. Each subchannel is encoded into a QAM constellation of the appropriate size. For example, if four bits are assigned to tone $i$, then tone $i$ will use a 16-QAM constellation for encoding. Optional coding can be inserted as shown in the encoder block in Figure 4.4. The modulation function,
IFFT, converts the encoded binary data in QAM constellation format from frequency domain to time domain signal for transmission. The demodulation function (FFT) at the receiver reverts the time domain signal back to frequency domain data.

The DMT system shown in Figures 4.4 and 4.5 also contains a guard band in the form of cyclic prefix. The length of the cyclic prefix determines the amount of the guard band. As long as the inter-symbol interference (ISI) of the channel is not longer than the length of the cyclic prefix, then the use of the cyclic prefix results in DMT symbols that are free of inter-block interference (IBI).

A potential problem in the implementation of the DMT modulation methods is the use of the cyclic prefix of an extra $\nu$ samples, where $\nu + 1$ is the length in sampling periods of the channel pulse response [35]. The required overhead with respect to the data rate is then $\nu/(N_c + \nu)$. On many practical channels including the high-speed subscriber loop, $\nu$ can be large. To minimize the data rate loss due to the overhead, $N_c$ needs to be very large, potentially 10,000 samples or more. Complexity is still minimal with the DMT methods with large $N_c$, when measured in terms of operation per unit time interval. However, large $N_c$ implies large memory requirement to store the bit allocation tables and intermediate FFT/IFFT results, often dominating the implementation complexity. Further, large $N_c$ implies longer latency in processing. Long latency can create problems with synchronization and can also be unacceptable with certain higher-level data protocols.
Chapter 4. Performance of A Fast Algorithm in High-speed Subscriber Loop

One solution to the latency problem is to use a combination of short-length equalization and the DMT to reduce $\nu$, and thereby the required $N_c$, to reach the highest performance levels with less memory and less latency [38]. The equalizer used is known as a time-domain equalizer (TEQ).

At the receiver front-end, the TEQ, $w(D)$, in the form of a $(M + 1)$-tap FIR filter is performed (as shown in Figure 4.5). This TEQ shortens the channel response so that the combined response of the channel and the TEQ taps is limited to a small number of samples, ideally equal to or less than the length of the cyclic prefix. Following the TEQ, the cyclic prefix samples are removed from the equalized signal. The remaining equalized time-domain samples, $z_k, k = 0, 1, \ldots, N_c - 1$, are then demodulated by the $N_c$-point FFT function to frequency-domain data, resulting in $N_c/2$ complex subsymbols. Appropriate gain and phase adjustment in the form of a set of $N_c/2$ 1-tap complex LMS filters $(A_n, n = 0, 1, \ldots, N_c/2 - 1)$ are multiplied to the demodulated signal $(Z_n, n = 0, 1, \ldots, N_c/2 - 1)$ before symbol decision and decoding. If the optional coding is used at the transmitter, then a corresponding Viterbi will be added at the receiver.

The TEQ and cyclic prefix for the DMT system are analogous to the feedforward taps and the feedback taps of a DFE system. Both the TEQ of a DMT system and the feedforward taps of a DFE system are FIR filters. The feedback taps of a DFE system are equal to the combined channel and feedforward FIR response. The length of the feedback filter is the same as the length of the cyclic prefix in terms of the functionality.

[36] suggests a method to compute the TEQ taps. However, this method requires matrix inversion computation and often results in fairly high cost of implementation, especially for cases of large size FFT. An alternative method is proposed in [37] that requires far less complexity. The method in [37] starts with specifying the number of FIR taps for the TEQ and the length of the cyclic prefix. It then iterates until a suitable criterion is satisfied. The drawback of this method is the length of time it takes to obtain a good set of TEQ taps. Even though the TEQ training is performed only once during startup or initialization, there are applications where a short startup time is critical.
Thus in Section 4.4, we will show a fast training algorithm that has less complexity than the method described in [36] and requires less training time than the method described in [37].

### 4.3 Equalizer settings for the TEQ

The structure of the TEQ is described in [36]. The goal of the TEQ is to reduce the size of the cyclic prefix so that the overall data rate loss is minimized. The TEQ uses the feedforward filter $w$ to shape the channel to the target response $b$ with a short length. There are several criteria that can be applied to the target response. In this chapter, we put a unit tap constraint on $b$ so that one of $b$ tap is set to 1. Actually, the TEQ proposed in [36] subsumes the DFE since the feedback filter in the DFE (corresponding to the target response in the TEQ) is restricted to being monic. With a unit tap constraint, the TEQ chooses the optimum settings $w$ and $b$ that minimize the mean square error.

Again, the desired target response in the TEQ [36] is

$$b = \frac{R_{zl|y}^{-1} e_m}{e_m^T R_{zl|y}^{-1} e_m} \quad (4.1)$$

where

$$R_{zx|y} = R_{xx} - R_{xy} R_{yy}^{-1} R_{yx}$$

and $e_m$ is the unit column vector with 1 in $m$ th position.

Also, the feedforward filter (the TEQ taps [36]) is

$$w = R_{yx} R_{yy} b. \quad (4.2)$$

The optimal $b$ and $w$ are determined by searching for all possible $m$ ($0 \leq m \leq N - 1$) to maximize the achievable data rate in the DMT system.

It is straightforward to check that with $l = 0$ the above solutions equal to the DFE solution. Since the DFE solution is a subset of the TEQ solution, it is easy to extend the fast algorithm for the DFE in the previous chapter to one for the TEQ settings. In the following section, we briefly revisit the fast algorithm and extends to the TEQ solution in a colored noise channel.
4.4 A Fast Equalization Algorithm

This section illustrates the fast algorithm for the TEQ in a symbol-spaced case with a colored noise. We will define $\mathbf{P}_1$ as a $M \times (N+1)$ submatrix of $\mathbf{P}$, consisting of first $N+1$ columns of $\mathbf{P}$. Following the definitions in Appendix B, it is easy to see that in a colored noise case, the Toeplitz autocorrelation matrices are approximated to the circulant matrices:

$$
\mathbf{R}_{yy} = \frac{1}{M^2} \mathbf{P}^* \Lambda \tilde{\mathbf{\epsilon}}_x \mathbf{M}^{-1} \mathbf{H}^\dagger M \mathbf{H} + \mathbf{N}^2 \mathbf{P}
$$

and

$$
\mathbf{R}_{yx} = \frac{1}{M} \mathbf{P}^* \Lambda \tilde{\mathbf{\epsilon}}_x M \mathbf{H}^\dagger \mathbf{P}_1
$$

where $H_i$ is $i$th element of $M$-point DFT of the channel impulse response, $N_i$ is the $i$th element of the DFT of $[n_{M-1}, n_{M-2}, \ldots, n_0]$ and $p_{\Delta, i} = e^{-j2}\pi i/M$.

Substituting the above equations into equations in the previous section yields

$$
\mathbf{R}_{xz|y} = \frac{1}{M} \mathbf{P}_1^* \Lambda \frac{\tilde{\mathbf{\epsilon}}_x^2 |N|^2}{\tilde{\mathbf{\epsilon}}_x M \mathbf{H}^\dagger |N|^2} \mathbf{P}_1
$$

and

$$
\mathbf{R}_{yy}^{-1} \mathbf{R}_{yx} = \mathbf{P}_1^* \Lambda \frac{\tilde{\mathbf{\epsilon}}_x^2 |N|^2}{\tilde{\mathbf{\epsilon}}_x M \mathbf{H}^\dagger |N|^2} \mathbf{P}_1.
$$

We note that $\mathbf{R}_{xz|y}$ is an $N \times N$ upper left submatrix of a circulant matrix whose first column is the inverse DFT of $\frac{\tilde{\mathbf{\epsilon}}_x^2 |N|^2}{\tilde{\mathbf{\epsilon}}_x M \mathbf{H}^\dagger |N|^2}$.

It can be shown using the Schur complement, the inverse of $\mathbf{R}_{xz|y}$ is

$$
\left( \frac{1}{M} \mathbf{P}_1^* \Lambda_{X|Y} \mathbf{P}_1 \right)^{-1} = \frac{1}{M} \mathbf{P}_1^* \Lambda_{X|Y}^{-1} \mathbf{P}_1 - \frac{1}{M} \mathbf{P}_1^* \Lambda_{X|Y}^{-1} \mathbf{P}_2 \left( \mathbf{P}_2^* \Lambda_{X|Y}^{-1} \mathbf{P}_2 \right)^{-1} \mathbf{P}_2^* \Lambda_{X|Y}^{-1} \mathbf{P}_1
$$

where $\Lambda_{X|Y} = \frac{\tilde{\mathbf{\epsilon}}_x^2 |N|^2}{\tilde{\mathbf{\epsilon}}_x M \mathbf{H}^\dagger |N|^2}$.

Neglecting the last term of the right hand side in the above equation, $\mathbf{R}_{xz|y}^{-1}$ is approximated by

$$
\mathbf{R}_{xz|y}^{-1} = \frac{1}{M} \mathbf{P}_1^* \Lambda_{X|Y}^{-1} \frac{\tilde{\mathbf{\epsilon}}_x^2 |N|^2}{\tilde{\mathbf{\epsilon}}_x M \mathbf{H}^\dagger |N|^2} \mathbf{P}_1 = \frac{1}{\tilde{\mathbf{\epsilon}}_x M} \mathbf{P}_1^* \Lambda \frac{\tilde{\mathbf{\epsilon}}_x \mathbf{H}^\dagger |N|^2}{|N|^2 + 1} \mathbf{P}_1
$$
Note that when \( w \) and \( b \) have the same length \( (M = N) \), \( R_{xy}^{-1} \) is exactly \( \frac{1}{M} P^* \Lambda_{x|y}^{-1} P \) and no approximation takes place.

Plugging the above equation into equation (4.1) yields

\[
\begin{align*}
    b &= \frac{1}{k} \frac{1}{\phi_2 M} P^* \Lambda_{x|y}^{-1} c \Lambda_{x|M|y|^2 + |N|^2} + 1 \phi_1 e_m \\
    &= \frac{1}{k'} P^* \Lambda_{x|M|y|^2 + |N|^2} + 1 \phi_m \\
    &= \frac{1}{k'} [I 0] P^* \\
    &= \begin{bmatrix}
        \text{SNR}_0 + 1 \\
        (\text{SNR}_1 + 1)e^{-j2\pi m/M} \\
        \vdots \\
        (\text{SNR}_{M-1} + 1)e^{-j2\pi (M-1)m/M}
    \end{bmatrix}
\end{align*}
\]

where \( k' \) is a scaling constant to make the \( m \) th element of the target response \( b \) equal to 1 \( (b_m = 1) \), \( \phi_m \) is the \( m \) th column of \( P \), \( I \) is the \( N \) by \( N \) identity matrix and \( \text{SNR}_i = \frac{\phi_2 M |H_i|^2}{|N_i|^2} \).

Then, defining \( \tilde{B}_i \) as

\[
\tilde{B}_i = \frac{1}{k'} \text{SNR}_i + 1)e^{-j2\pi im/M}, \quad i = 0, 1 \ldots M - 1,
\]

the target response can be obtained from the IDFT operation:

\[
b_k = \frac{1}{M} \sum_{i=0}^{M-1} \tilde{B}_i e^{j2\pi ik/M}, \quad k = 0, 1 \ldots N - 1.
\]

With \( b \), we can compute the feedforward filter \( w \) from equation (4.2):

\[
w = P^* \Lambda \frac{c \Lambda_{x|M|y|^2 + |N|^2}}{\phi_2 M |H_i|^2 + |N|^2} P \begin{bmatrix} b \\ 0 \end{bmatrix}.
\]

Then, multiplying both sides by \( P \) yields

\[
W_i = \frac{\tilde{\phi}_2 M H_i^* \phi_i \cdot B_i}{\phi_2 M |H_i|^2 + |N|^2}, \quad i = 0, 1 \ldots M - 1
\]

where \( W_i \) is the \( i \) th DFT element of \( w \) and \( B_i \) is the \( i \) th \( M \)-point DFT element of \([b_0 \ b_1 \ldots b_{N-1} \ 0 \ldots 0] \).
From this equation, we can obtain $w_k$ using the IDFT:

$$w_k = \frac{1}{M} \sum_{i=0}^{M-1} W_i e^{j2\pi ik/M}, \quad k = 0, 1, \cdots, M - 1.$$ 

In summary, we compute the TEQ settings for each $m$ value and pick the optimum settings to achieve the highest throughput in the DMT system. In the following section, we present simulation results of the fast algorithm performed on various High-speed Subscriber Loops of the fast algorithm.

4.5 Simulations

This section presents simulation results using the fast algorithm described in the previous section and compares them with the optimal TEQ settings obtained by a direct matrix inversion. A symbol-spaced equalizer is assumed in this simulation. Also the target response $b$ is properly scaled so that performance is optimized. To simplify the analysis in simulations, we restrict $m$ to 0. Also, instead of the data rate, $\text{SNR} = \frac{\tilde{e}_x}{E[|e_k|^2]}$ is used as a performance measure to indicate the performance of the fast algorithm.

Several length copper loops with 26 gauge have been used for simulations. Matched Filter Bound (MFB) $\text{SNR}_{\text{MFB}} = \frac{||h||^2 \tilde{e}_x}{\sigma_n^2}$ is set to 20 dB throughout this study, if not stated otherwise. In the plots, the solid line represents the optimal SNR computed from the direct matrix inversion and the dashed line indicates the SNR computed from the fast algorithm.

1) Length of the feedforward filter: In Figures 4.6 and 4.7, the number of the feedforward filter taps ($M$) is changed for a 26 gauge 400$m$ and a 10$m$ loop, respectively. For each simulations, the lengths of the target response are set to 8 and 10. It is obvious that as the length of the feedforward filter increases, performance of the proposed algorithm approaches that of the optimal TEQ coefficients within a tenth of a dB. In a shorter loop, the proposed algorithm works better. As shown in Figure 4.7, with feedforward filters longer than 20 taps, the gap between the optimal settings and the coefficients obtained from the fast algorithm becomes indiscernible. We have not shown in the plots for shorter feedforward filter taps, but simulations indicate
Chapter 4. Performance of A Fast Algorithm in High-speed Subscriber Loop

that the fast algorithm requires roughly twice longer feedforward filter taps than the exact solution settings to reach the optimal solution.

2) Length of the target response: We performed the fast algorithm on 26 gauge 9 kft and 100m loop in Figures 4.8 and 4.9 with various choices of the target response length. For simulation, the length of the feedforward filter is set to an integer power of 2 (64 and 32 in Figures 4.8 and 4.9 respectively) so that the DFT operations are carried out by the Fast Fourier Transform (FFT) to speed up the computation. 9 kft 26 gauge loop channel is used as an example for ADSL loop, since this loop represents one of the worst case loops among the set of CSA loops [39]. For both cases, the proposed algorithm generates near-optimal solutions.

3) Transmit power: We changed the transmit power in 400m 26 gauge loop. Figure 4.10 shows the performance of the proposed algorithm from MFB = 10 dB to 30 dB. The lengths of the feedforward filter and target response are set to 64 and 18, respectively. Simulation indicates the fast algorithm generates solutions close to the optimum settings consistently regardless of the transmit power.
4) Crosstalk coupling: The near-end crosstalk (NEXT) term is modeled [40] with a coupling function of the form: $|H_{NEXT}(f)|^2 = K_{NEXT} f^{3/2}$ where $f$ is the frequency in Hz and $K_{NEXT}$ is determined through empirical measurement. In this simulation, the far-end crosstalk (FEXT) is assumed to be negligible because we consider short loops here. We changed $K_{NEXT}$ from $10^{-15}$ to $10^{-13}$ in 400m 26 gauge loop and the fast algorithm in a colored noise case is used for simulation. In this simulation, an input signal power of 10mW and a noise power of -30 dBm across a two-sided bandwidth are assumed. As the effect of the near-end crosstalk increases, the output SNR decreases in Figure 4.11. Again, the SNR from the proposed fast algorithm is only a few tenth of a dB away from the optimum solution in a subscriber loop corrupted by the near-end crosstalk.

Our performance evaluations showed that the proposed algorithm performs very well in various situations. The gap between the optimal SNR and the SNR from the proposed algorithm is shown to be a few tenths of a dB and becomes negligible as
the length of the feedforward filter increases. Thus, for any sufficiently long feedforward filter, the proposed algorithm can compute the optimal equalizer setting very efficiently without any performance loss.

4.6 Conclusion

We have illustrated a fast algorithm for the time-domain equalizer (TEQ) in the discrete multitone (DMT) modulation for high performance copper networks. The overall computation is carried out with negligible performance loss as the number of the feedforward filter taps increases.

Simulations have been performed in various copper loop channels. Effects of the feedforward and target response length, the matched filter bound and the near-end crosstalk coupling have been examined. Simulation results show that the fast algorithm yields near-optimal settings very efficiently in various environments.
Figure 4.8: Performance of the fast algorithm with different target response length

Figure 4.9: Performance of the fast algorithm with different target response length
Figure 4.10: Performance of the fast algorithm with different MFB

Figure 4.11: Performance of the fast algorithm with different Knext
Chapter 5

Equalized Maximum Likelihood Receiver in a Storage System

As the length of a channel response increases, the complexity of implementing the Maximum Likelihood Sequence Detector (MLSD) grows exponentially. Equalized Maximum Likelihood (EML) with finite length has been proposed to approach optimum performance with significantly reduced complexity. Two different constraints on the EML settings are analyzed. In this scheme, the performance degradation due to the colored noise and misequalization is taken into account. EML achieves an excellent trade-off between complexity and performance. A comparison with Partial Response Maximum Likelihood (PRML) is made for a magnetic recording channel. Our results indicate that significant performance improvement over PRML can be achieved especially at high recording densities. Implementation issues such as the adaptive algorithm for finding the optimum settings recursively and the complexity of the Viterbi detector are also investigated. Multiplication operations in the Viterbi detector can be avoided by employing a look-up table.

5.1 Introduction

Maximum Likelihood Sequence Detection (MLSD) is known to have optimum performance on an additive white Gaussian noise (AWGN) channel. However, as the length
of a channel increases, the complexity of implementing an MLSD receiver grows exponentially as $M^{\nu}$, where $M$ is the number of input level and $\nu$ is the length of the channel.

Many schemes have been proposed to reduce the complexity of the Viterbi detector. One common method is to use a linear equalizer to shape the channel to one having a shorter length [41, 42, 43, 44, 45]. In some of these approaches, the target response is chosen as a truncated version of the infinite length response. However this choice is not optimal for the finite length case. Also, the performance degradation of the Viterbi detector due to noise colorization and misequalization is neglected.

In this chapter, we have proposed the Equalized Maximum Likelihood (EML) system with finite length for a magnetic recording channel. The similar idea of maximum likelihood receiver for multicarrier systems was first described by Chow and Cioffi [36]. Also, performance analysis for EML detector with infinite length case was made in [46].

The goal of EML is to find an equalizer, $w$, and a target channel, $b$, that maximize performance. Performance of EML is characterized by the effective minimum distance of $b$ and the mean square error (MSE) between the equalizer output and the output of the target channel response. The equalizer and its target response can be optimized using the Minimum Mean Square Error (MMSE) solution. Then, the Viterbi algorithm is tuned to the target response, $b$, for detection. By doing so, we can reduce the complexity of the Viterbi detector significantly, especially when the length of $b$ is small compared with the length of the original channel.

Several criteria can be applied for the target response. Falconer et. al. [42] and Fitzpatrick [45] proposed the similar structure with different criteria in choosing the target response. In [42], the mean square error between the equalizer output and the target channel response is minimized subject to a unit-energy constraint on the target response. In [45], the target response having a form of $(1 - D)(1 + a_0D + a_1D^2 + \cdots)$ is proposed, where $a_i$'s are selected by limited search over the finite range, not by optimizing over all possible values. In contrast, the optimization in EML is not constrained to any specific class of target responses. In this work, we analyze EML with two constraints: a unit tap constraint and a unit energy constraint. EML tries
Chapter 5. Equalized Maximum Likelihood Receiver in a Storage System

58

to minimize the MSE with a given constraint on $b$.

Zero Forcing (ZF) criterion is not useful for a magnetic recording channel, since the channel spectrum approaches null at the Nyquist frequency at high densities. Thus, the equalizer settings using the ZF criterion lead to unacceptable performance, due to noise enhancement.

Partial Response Maximum Likelihood (PRML) has been proposed as an alternative method of reduced state sequence detection. Over the last decade, there has been an increasing interest in the use of partial response detection for magnetic recording channels. PRML uses a linear equalizer to equalize the channel to the given partial response. Thus the noise spectrum is colored, which deteriorates performance in Viterbi detector. PRML differs from EML in that the target channel response is chosen first and then an optimal equalizer is found. Since EML optimizes over both the equalizer and the target channel response simultaneously, we expect EML to have performance superior to that of PRML.

In Section 5.2, we explain the structure of EML in both infinite and finite length cases and describe the magnetic recording channel model. Next, we derive the EML settings with a unit tap constraint and a unit energy constraint on the target response in Sections 5.3 and 5.4 respectively.

In Section 5.3, EML tries to minimize the MSE with a unit-tap constraint on $b$. Also, the position of a unit-tap is not restricted to the first tap. The target response, $b$, in EML with this constraint is not restricted to be minimum phase, which is the case for the feedback filter in Decision Feedback Equalizer (DFE). Thus, we can equalize the channel more easily to the target response with a short feedforward filter, since a magnetic recording channel is a non-minimum phase channel.

Similarly, a unit energy constraint on the target response in EML will be described in Section 5.4. Also, design of an unbiased receiver is briefly considered. Section 5.5 depicts the complexity issue of the Viterbi detector in the EML receiver. A brief comparison between two constraints are made in Section 5.6.

In both Sections 5.3 and 5.4, we compare EML with PRML in various recording densities. Simulation shows that EML outperforms PRML at every recording density. Superiority of EML is verified by bit simulation.
5.2 Structure of EML and channel model

The basic structure of the Equalized Maximum Likelihood system with the infinite is shown in Figure 5.1. From the figure, the channel output $y_p(t)$ is modelled as

$$y_p(t) = \sum_{m} x_m p(t - mT) + n_p(t) \tag{5.1}$$

where $x_m$ is the channel input symbol which takes on values $\{+1, -1\}$, $p(t)$ is the channel pulse response equal to $||p||\varphi_p(t)$, $T$ is the symbol period, and $n_p(t)$ is additive white Gaussian noise with power $N_0/2$.

We will define the D-transform of a sequence $x_k$ as $x(D) = \sum_m x_m D^m$. Note that we can also use $x^*(D^{-*}) = \sum_m x_{-m}^* D^{-m}$. Then the sampled output sequence $y(D)$ is defined as

$$y(D) = ||p||Q(D)x(D) + n(D)$$

where $Q(D) = \varphi_p(D)\varphi_p^*(D^{-*})$ and $n(D) = \varphi_p^*(D^{-*})n_p(D)$.

Also we can write the error sequence as

$$e(D) = b(D)x(D) - w(D)y(D)$$

where $b(D)$ is the channel target response and $w(D)$ is the equalizer. It is also convenient to use $R_{xx}(D) = E[x(D)x^*(D^{-*})]$ and $R_{xy}(D) = E[x(D)y^*(D^{-*})]$.

As shown in the figure, the equalizer $w(D)$ tries to shape the channel to the target response $b(D)$ which satisfies the given constraint, and then a Viterbi detector is used for data estimation. We optimize $b(D)$ and $w(D)$ jointly to minimize the mean square error, $\sigma_e^2$. By a well-known orthogonality principle, the error sequence $\{e_k\}$ should
be orthogonal to the channel observation \( \{y_k\} \). Thus,

\[
R_{xy}(D) = E[e(D)y^*(D^{**})] = b(D)R_{xy}(D) - w(D)R_{yy}(D) = 0.
\]

Then,

\[
w(D) = b(D)\frac{R_{xy}(D)}{R_{yy}(D)}. \tag{5.2}
\]

Using a canonical factorization of \( Q(D) + \frac{1}{\text{SNR}_{\text{MFB}}} = \gamma_0^2 g(D)g^*(D^{**}) \) where \( \text{SNR}_{\text{MFB}} \) is the matched filter bound signal–to–noise ratio, the error autocorrelation function is

\[
R_{ee}(D) = \frac{\alpha_0}{\|p\|^2\gamma_0^2} b(D)b^*(D^{**}) g(D)g^*(D^{**}).
\]

In [36], it was shown that the MMSE is

\[
\sigma_e^2 = \frac{\alpha_0}{\|p\|^2\gamma_0^2} \frac{\|b\|^2}{\|g\|^2}.
\]

and is achieved by any \( b(D) \) satisfying

\[
b(D)b^*(D^{**}) = \frac{\|b\|^2}{\|g\|^2} g(D)g^*(D^{**}).
\]

Note that \( b(D) \) need not to be monic or causal. From equation (5.2), we find

\[
w(D) = \frac{1}{\|p\|^2\gamma_0^2} \frac{b(D)}{g(D)g^*(D^{**})}.
\]

Note that with this choice of \( w(D) \), the error sequence is white, which is important in sequence detection.

In [47], the signal–to–noise ratio of the EML system was found to be

\[
\text{SNR}_{\text{EML}} = \frac{d_{\text{min},b}^2}{4\sigma_e^2} \tag{5.3}
\]

\[
= \frac{\gamma_0^2 d_{\text{min},g}^2}{4\text{SNR}_{\text{MFB}}}
\]

where \( d_{\text{min},b} \) and \( d_{\text{min},g} \) are the minimum distance in a trellis description of \( b(D) \) and \( g(D) \) respectively. Note that \( \text{SNR}_{\text{EML}} \) is independent of the choice of \( b(D) \) as long as \( b(D) \) satisfies equation (5.2).
Figure 5.2 shows the structure of the EML in finite length case. In an FIR equalizer, a low pass filter is often assumed as a front-end filter replacing the matched filter in Figure 5.1.

A magnetic recording channel is modelled using a Lorentzian step response

$$s(t) = \frac{1}{1 + (2t/pw_{50})^2}$$

where $pw_{50}$ is the width of the pulse at 50% of its peak value.

In a magnetic recording system, $pw_{50}/T$ indicates the recording density. The pulse response $p(t)$ in equation (5.1) is obtained from $s(t)$ by

$$p(t) = s(t) - s(t - T).$$

It is obvious from the above equation that a magnetic recording channel is DC free.

For deriving the EML settings, we follow the general channel model described in Section 2.1. The optimum settings with a unit tap constraint and a unit energy constraint are derived in the following sections in finite length case.

### 5.3 EML with a unit tap constraint

In this section\(^1\), we derive the equalizer settings which optimize performance for the given complexity. With a unit tap constraint, the technique used to calculate the

---

\(^1\)We would like to thank Cory Modlin of Stanford University for helpful discussions.
equalizer and target channel response is similar to that used to compute the feed-forward and feedback filters for the Minimum Mean Square Error Decision Feedback Equalizer (MMSE-DFE) described in Chapter 2. While the problem formulations are the same, EML differs in that the target channel response is not restricted to being causal or monic as in the case of the feedback filter in MMSE-DFE.

5.3.1 Equalizer settings

From the figure 5.1, the equalizer output is \( z_k = w^T y_k \) and the error signal is

\[
e_k = b^T x_{k-\Delta} - w^T y_k \tag{5.4}
\]

where \( b \) is the target response vector with length \( \nu + 1 \), \( w \) is the equalizer vector with length \((N_f+1)l\) and \( x_{k-\Delta}^T = [x_{k-\Delta} x_{k-\Delta-1} \ldots x_{k-\Delta-\nu}] \). Note that \( b \) does not have to be causal as in MMSE-DFE. In the above equation, \( \Delta \) represents the decision delay. The optimum decision delay \( \Delta \) has been studied in [48].

We set \( b_i = 1 \) and define new vectors \( \tilde{w} \) and \( \tilde{y}_k \) as

\[
\tilde{w} = \begin{bmatrix} w \\ b_i \end{bmatrix}, \quad \tilde{y}_k = \begin{bmatrix} y_k \\ x_{k-\Delta,i} \end{bmatrix}
\]

where \( b_i \) and \( x_{k-\Delta,i} \) are defined respectively as \( b_i^T = [-b_0 \ldots -b_{i-1} - b_{i+1} \ldots - b_{\nu}] \) and \( x_{k-\Delta,i}^T = [x_{k-\Delta} \ldots x_{k-\Delta-i+1} x_{k-\Delta-i-1} \ldots x_{k-\Delta-\nu}] \).

The value of \( i \) determines to which tap the equalizer tries to concentrate the channel energy, and represents the relative phasing of the two filters \( b \) and \( w \). For example, setting \( i = \nu \) makes \( b \) anti-causal. We can rewrite equation (5.4) as

\[
e_k = x_{k-\Delta-i} - \tilde{w}^T \tilde{y}_k. \tag{5.5}
\]

From the orthogonality principle, we have the following solutions:

\[
\tilde{w} = R_{\tilde{y}\tilde{y}}^{-1} R_{\tilde{y}x}
\tag{5.6}
\]

and

\[
\sigma_e^2 = \bar{E}_x - R_{x\tilde{y}} R_{\tilde{y}\tilde{y}}^{-1} R_{\tilde{y}x}
\]
where \( R_{\tilde{y}\tilde{y}} = R_{\tilde{y}x}^T = E(x_k - \Delta \tilde{y}^T) \), \( R_{\tilde{y}\tilde{y}} = E(\tilde{y}\tilde{y}^T) \) and \( \tilde{e}_x = E(|x|^2) \). In these expressions, the equalizer settings are a function of \( \Delta \) and \( i \).

The non-white error sequence and misequalization degrade performance of MLSD designed for an AWGN channel. The error sequence of EML system with infinite length is white [46]. Thus, as the length of the filters increases, the error sequence of finite-length EML approaches a white sequence, and degradation due to the colored noise is not severe in EML. When misequalization takes place, the probability of mistaking two sequences depends both on the difference between the sequences and on the specific transmitted sequence. The effective signal-to-noise ratio (SNR) for the sequence detector with a colored error sequence and misequalized channel was derived in Appendix A. The general expression for performance of sub-optimal receiver with misequalization has been analyzed in [49, 50, 51, 52].

Again, following the definitions in Appendix A, the effective SNR can be defined as

\[
\text{SNR}_{\text{eff}} = (\min_{x, \tilde{x}} \frac{||e||^2 - 2e \cdot (y - \tilde{y})}{2\sqrt{e^TRe}})^2
\]

Note that for a white error sequence with perfect equalization, the above expression reduces to a familiar expression \( Q(\frac{d_{\min}}{2\sigma}) \).

Here we assume that the error sequence has a Gaussian distribution. In [53], the assumption of Gaussian Inter-Symbol Interference (ISI) is shown to give only a slightly larger probability of error than calculated using the actual distribution.

We can note that when setting \( b_0 = 1 \), equation (5.6) simply represents the familiar MMSE-DFE solution. For \( b_0 = 1 \), the feedforward equalizer for MMSE-DFE is the same as the equalizer for EML and the feedback equalizer for MMSE-DFE is the same as the target channel response for EML. The feedback filter of the MMSE-DFE tries to convert the channel to minimum phase. In the case where the channel is already near to minimum phase, we do not need a long feedforward filter. However, if it is not, we may need a very long feedforward filter. In contrast, EML can take advantage of the fact that \( b \) can be chosen to be non-causal (or mixed phase). If the channel is maximum phase (or at least mixed phase as in the magnetic recording channel), then for a given finite length filter, EML can shape the channel more easily to the target channel than can the MMSE-DFE. Non-causality of \( b \) can be simply
implemented in the Viterbi algorithm with delay.

The filter \( w \) and \( b \) can be optimized for a given finite complexity by changing \( \Delta \) and \( i \) in equation (5.6). From that optimization, we can compute an autocorrelation matrix \( R \) and search for \( w \) and \( b \) which maximize the effective SNR for EML. The solution corresponding to this maximum is the optimum equalizer setting. Then, we can add a Viterbi detector tuned to the target response \( b \) at the end of the equalizer.

EML settings derived in this section is biased. We will derive an unbiased receiver in the following section.

### 5.3.2 Design of an unbiased receiver

First, we examine the bias problem in sequence detection. Since EML uses the MMSE criterion to choose the equalizer settings, the estimation becomes biased. Removing bias leads to an improved receiver from error probability point of view.

Define \( y \) as the noiseless channel output vector and \( z \) as the output vector corrupted by the AWGN vector \( n \). Then,

\[
z = y + n.
\]  

(5.7)

Maximum Likelihood Sequence Estimation is defined as the choice of that \( y \) for which the conditional probability density \( P_{z|y} \) is maximum. Then, the derivation of the Viterbi error metric \( ||z - y||^2 \) comes from

\[
P_{z|y}(z|y) = P_n(z - y).
\]

In this derivation, there are two basic assumptions: First, \( n \) should be white. Secondly, \( y \) should be uncorrelated with \( n \).

We can rewrite equation (5.4), analogously to equation (5.7), as

\[
z_k = b^T x_{k-\Delta} - e_k.
\]

To satisfy the above assumptions for an unbiased estimator, \( b^T x_{k-\Delta} \) and \( e_k \) should be uncorrelated with each other.

A bias of EML can be defined as \( E(z_k|b^T x_{k-\Delta}) - b^T x_{k-\Delta} \). This bias violates the second assumption. To obtain an unbiased estimation, we need to get \( E(e_k|b^T x_{k-\Delta}) = 0 \).
As shown in equation (5.5), $e_k$ contains the term $x_{k-\Delta-i}$ which should be regarded as input data for $z_k$. To arrive at the unbiased EML solution, we start with the biased solution described earlier. Derivation of a bias for EML is the same as in finite-length MMSE-DFE case [11]. The bias can be expressed in terms of $\sigma_e^2/\bar{\xi}_x$. Then, the channel output is expressed as

$$z_k = b^T x_{k-\Delta} - \frac{\sigma_e^2}{\bar{\xi}_x} x_{k-\Delta-i} + \alpha_k.$$ 

where $\alpha_k$ represents both the remaining ISI terms and the noise.

Thus, we can include the bias term into the target response by setting $b_i = 1 - \sigma_e^2/\bar{\xi}_x$ instead of $b_i = 1$. Following this procedure, $b^T x_{k-\Delta}$ and $e_k$ do not have any overlapped input data term, thus the unbiased EML setting is obtained.

### 5.3.3 Comparison with PRML

In this section, we compare performance of EML to that of the Partial-Response Maximum Likelihood (PRML) in a magnetic recording channel model described in Section 5.2.
The extended partial response (EPR) class of polynomials was introduced by Thapar [54] to provide a set of models for the saturation recording channel as linear density increases. We denote this class as $P_n(D)$:

$$P_n(D) = (1 + D)^n(1 - D)$$

where $D$ represents a delay operator.

We are especially interested in the PR4 ($n = 1$) and EPR4 ($n = 2$) cases. Figure 5.3 shows spectra of the partial responses with $n = 1, 2$ and 3. PR4 is presently the most popular choice of the PR target in current magnetic recording channels. PR4 and EPR4 provide the best choice of $n$ for respectively $1 \leq pw_{50}/T \leq 2.5$ and $1.5 \leq pw_{50}/T \leq 3$ in a practical magnetic recording channel. The PR4 allows the interleaving decoding so that the design of the Viterbi detector is simplified. At present densities, the PR4 target response provides a good matching to the natural channel response of the magnetic channel so that the noise enhancement due to equalization is kept to an acceptable level. However, as the linear density increases, the EPR4 provides a better match to the magnetic recording channel where $pw_{50}/T$ is up to 3.
Note that EML subsumes PRML because the target response of EML includes any $P_n(D)$.

Comparisons between EML and PRML are shown in Figures 5.4 and 5.5. In all simulations, a 5 tap feedforward filter, $w$, is used. ($l = 1$ and $N_f + 1 = 5$) The noise power $\sigma^2$ is scaled so that Matched Filter Bound (MFB), defined as $\|p\|^2 \sigma_x$, equals to 25 dB at $pw_{50}/T = 1$. The $x$-axis represents the recording density. In order to compare the two systems at the same complexity, we set the length of the target response $b$ to 3 and 4 in EML, which corresponds to the 4 and 8 state Viterbi detector respectively.

As shown in Figures 5.4 and 5.5, EML outperforms PRML especially at high densities. We notice that as the number of the states in the Viterbi detector increases, the performance gap between EML and PRML also increases at high densities. Simulation shows that performance of PRML degrades rapidly because of colored noise as the density increases. The Viterbi loss factor for PRML introduced in [53] is calculated to take the noise colorization effect into account.
Table 5.1: Target response b of EML

<table>
<thead>
<tr>
<th>$\frac{\text{pw}50}{T}$</th>
<th>EML (4 states)</th>
<th>EML (8 states)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$-0.130 D^{-1} + 1 - 0.841 D$</td>
<td>$1 - 0.439 D - 0.521 D^2 - 0.024 D^3$</td>
</tr>
<tr>
<td>1.5</td>
<td>$1 - 0.479 D - 0.512 D^2$</td>
<td>$-0.711 D^{-2} - 0.413 D^{-1} + 1 + 0.158 D$</td>
</tr>
<tr>
<td>2</td>
<td>$1 - 0.479 D - 0.525 D^2$</td>
<td>$0.267 D^{-1} + 1 - 0.400 D - 0.793 D^2$</td>
</tr>
<tr>
<td>3</td>
<td>$-0.534 D^{-2} - 0.460 D^{-1} + 1$</td>
<td>$-0.844 D^{-2} - 0.400 D^{-1} + 1 + 0.340 D$</td>
</tr>
</tbody>
</table>

Figure 5.6: Spectra comparison at pw50/T = 1.5

As the length of the linear equalizer w grows, the error sequence of EML becomes white asymptotically as we have shown in the infinite length case [46]. With PRML, however, a severely colored error sequence remains even with a long linear equalizer. Thus we see performance degradation at high densities. Simulation indicates that the performance loss of PRML due to the colored error sequence is about 1 - 2 dB.

Table 5.1 shows the optimized target response b of EML for the given complexity. Since a magnetic recording channel has a spectral null at DC, it is desirable for the target response to be DC free, to avoid misequalization. If it is not DC free, then misequalization takes place at low frequencies and therefore, bit patterns with low frequency content are the most likely patterns to be mistaken for one another. In
Chapter 5. *Equalized Maximum Likelihood Receiver in a Storage System*  

![Comparison of spectrum (pw50/T = 3.5)](image)

**Figure 5.7:** Spectra comparison at pw50/T = 3.5

Table 5.1, most EML target channel responses are nearly DC free and thus misequalization is not large. Degradation due to misequalization can be alleviated using other methods such as forcing the target response to have a DC null (i.e. \((1 - D)^{P(D)}\)) or selecting a DC free modulation code [45].

Figures 5.6 and 5.7 compare the spectra at \(pw_{50}/T = 2\) and 2.5, respectively. The \(x\)-axis indicates the frequency \(\omega\). The target responses \(b\) for EML shown in the plots are \(1 - 0.479D - 0.525D^2\) and \(0.267D^{-1} + 1 - 0.400D - 0.793D^2\), respectively. The plots show that the optimized target response \(b\) fits to the channel spectrum better than the PR4 or EPR4 spectra at each density. The PR4 and EPR4 channels are designed to have spectral nulls at DC and the Nyquist frequency. Thus, especially near at the Nyquist frequency, there are discrepancies in spectra.

Figure 5.8 shows the simulated Bit Error Rate (BER) of EML and EPR4. BER estimation has been carried out by Monte Carlo simulation over 1,000,000 symbol bits. The \(x\)-axis indicates the channel SNR defined as the ratio of the power of the channel output to the noise power. In this simulation, the target response of 8 states EML has been optimized as the channel SNR varies. We can note that EML...
Figure 5.8: Comparison of Bit Error Rate
Figure 5.9: Effective SNR of EML
outperforms EPR4 by more than 1.5 dB. The simulation result confirms superiority of EML over PRML.

Figure 5.9 shows the effective SNR of EML with target response of length 3 and 4. MFB is also shown on this plot. MFB serves as an upper bound for ideal detector performance. As the number of the states increases, the performance of EML approaches that of the optimal receiver.

From simulations, we show that EML outperforms PRML at all recording densities. It is, however, noted that complexity of Viterbi detector will increase in EML because the target response is composed of floating point coefficients and generates more output levels in comparison with PRML.

5.4 EML with a unit energy constraint

This section proposes the Equalized Maximum Likelihood (EML) receiver with a unit energy constraint on the target response for a magnetic recording channel. EML settings derived with this constraint has a solution similar to the one derived by Falconer and Magee [42]. The difference is that EML optimizes the performance, taking into account the noise colorization and channel misequalization effects, while the receiver in [42] tries to minimize the Mean Square Error (MSE) with a unit energy target response neglecting any performance degradation in Viterbi detector.

In this section, the target response, b, in EML is constrained to have unit energy, and the solution with an additional DC free constraint is also derived. Noting that a magnetic recording channel has a null at DC, an additional DC free constraint can be applied on the target response to avoid performance degradation due to the channel misequalization. If the target response is not DC free, then the usual linear equalizer enhances noise at DC to shape to the target response given a channel with no power at DC.

Section 5.4.1 solves equations for EML with both unit energy and DC free constraints. The performance comparison between EML and PRML appears in Section 5.4.2. Again simulation indicates that EML has a substantial gain over PRML.
at every recording density. Actual bit simulation is carried out to verify the performance of EML. Section 5.4.3 presents the adaptive algorithm to implement EML.

5.4.1 EML settings

Similarly as done in the previous section, using the orthogonality principle, the feedforward filter for EML with a unit energy becomes

\[ w = R_{yy}^{-1} R_{yx} b \]  \hspace{1cm} (5.8)

where \( R_{yy} = E(yy^*) \) and \( R_{yx} = E(yx_{k-\Delta}^*) \).

Then, the corresponding minimum mean square error is

\[ \sigma_e^2 = E(|e_k|^2) = b^* R_\Delta b \]

where \( R_\Delta = \tilde{\epsilon}_x I - R_{yx} R_{yy}^{-1} R_{yx} \) and \( \tilde{\epsilon}_x = E(|x_k|^2) \). Note that \( R_\Delta \) is a function of the decision delay \( \Delta \).

First we set two conditions on the target response \( b \): unit energy and DC free constraint. To obtain the solution subject to the above constraints, we form a cost function using Lagrange multipliers:

\[ J = b^* R_\Delta b + \alpha (b^* b - 1) + \beta (e^* b - 0) \] \hspace{1cm} (5.9)

where \( e = [1 \ 1 \ \cdots \ 1]^T \) and \( \alpha \) and \( \beta \) are the Lagrange multipliers corresponding to the above two constraints respectively.

To minimize the MSE subject to the two constraints, we set \( \frac{\partial J}{\partial b} = 0 \):

\[ \frac{\partial J}{\partial b} = 2R_\Delta b + 2\alpha b + \beta e = 0 \]

Then, the solution is

\[ b = -\frac{\beta}{2} (R_\Delta + \alpha I)^{-1} e. \] \hspace{1cm} (5.10)

From the DC free constraint, we obtain

\[ -\frac{\beta}{2} e^* (R_\Delta + \alpha I)^{-1} e = 0. \] \hspace{1cm} (5.11)
and from the unit energy constraint, we get
\[
\frac{\beta^2}{4} e^* (R_\Delta + \alpha I)^{-2} e = 1
\]  
(5.12)

Since \( R_\Delta \) is symmetric, we can find a unitary matrix \( U \) by similarity transformation that diagonalizes \( R_\Delta \) such that \( R_\Delta = U^* \Lambda U \) where \( U^* U = I \). Note that \( \Lambda \) is a diagonal matrix whose diagonal elements are eigenvalues \( \lambda_k \) of \( R_\Delta \).

Substituting \( R_\Delta = U^* \Lambda U \) to equation (5.11) yields
\[
e^* U^* (\Lambda + \alpha I)^{-1} U e = 0
\]
since \( \beta \neq 0 \) for non-trivial solution.

Defining \( v = U e \) yields
\[
\begin{bmatrix}
\frac{1}{\lambda_1 + \alpha} \\
\ldots \\
\frac{1}{\lambda_\nu + \alpha}
\end{bmatrix}
\begin{bmatrix}
0 \\
\vdots \\
0
\end{bmatrix}
v = \sum_{i=1}^{\nu} \frac{|v_i|^2}{\lambda_i + \alpha} = 0
\]  
(5.13)

where \( v_i \) is the \( i \) th element of a column vector \( v \).

After solving for \( \alpha \) in the above equation, \( \beta \) is computed directly from the equation (5.12). Since multiple solutions for \( \alpha \) exist in the equation (5.13), \( \alpha \) is chosen such that it minimizes the MSE. Once \( \alpha \) and \( \beta \) are obtained, the target response \( b \) and the feedforward filter \( w \) can be computed.

As stated in the preceding section, both misequalization and the non-white error sequence deteriorate performance of the MLSD designed for an AWGN channel. It is also mentioned that the error sequence of EML system with infinite length is white when \( b \) has a unit tap constraint. It can be shown that the same is true when \( b \) has a unit energy constraint. Thus, as the length of the filters increases, the error sequence of finite-length EML approaches a white sequence, and performance degradation due to the noise colorization is not as severe as observed in PRML.

We use the same procedure employed in the previous section to obtain the optimal \( w \) and \( b \) to maximize the effective signal-to-noise ratio for EML.

When we relax a DC free constraint from the target response, the solution in equation (5.9), plugging \( \beta = 0 \), is reduced to the eigenvector corresponding to the
minimum eigenvalue $\lambda_{\text{min}}$ of $R_\Delta$:

$$b = \text{eigenvector}(\lambda_{\text{min}}).$$

This is the same result obtained in [42].

The solution obtained here is biased. We can improve performance from an error probability point of view by removing bias. It can be shown that the unbiased receiver setting is obtained by simply setting

$$b_{\text{unbiased}} = (1 - \frac{\sigma^2}{||b||^2})b.$$

5.4.2 Comparison with PRML

This section compares performance of EML with a unit energy constraint to that of the Partial–Response Maximum Likelihood (PRML) in Lorentzian channel described in Section 5.2. To simplify simulations, a unit energy constraint solution is used without a DC free constraint on the target response.
Figure 5.11: SNR gain of EML over EPR4

Comparisons between EML and PRML are shown in Figures 5.10 and 5.11. In all simulations, a symbol-spaced 5 tap feedforward filter, \( w \), is used. Again, both plots show that EML performs better than PRML as a recording density increases.

Table 5.2 indicates the optimized target response \( b \) of EML with a unit energy constraint. Degradation caused by misequalization at DC can be avoided by imposing a DC free condition, as done in Section 5.2. At current density of interest, the misequalization problem at DC is not severe, but as the recording density exceeds \( pw_{50}/T = 3 \), the target response should be chosen carefully.

Figure 5.12 represents the simulated Bit Error Rate (BER) of EML and EPR4 carried out by Monte Carlo method. The \( x \)-axis shows the channel SNR defined as \( \frac{E(|y|^2)}{\sigma_n^2} \). It is clear from the plot that EML has more than 1.5 dB gain over EPR4.

EML achieves the performance gain over PRML at the expense of the increased complexity in Viterbi detector. However, increase in the complexity of the Viterbi detector in EML can be minimized using the look-up table. The complexity of implementing the Viterbi detector in EML is further investigated in the following section.
Figure 5.12: Bit Error Rate Comparison
<table>
<thead>
<tr>
<th>γ</th>
<th>EML (4 states)</th>
<th>EML (8 states)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.78 - 0.62D - 0.11D²</td>
<td>-0.70 + 0.72D - 0.02D³</td>
</tr>
<tr>
<td>1.5</td>
<td>0.80 - 0.26D - 0.55D²</td>
<td>0.42 + 0.57D - 0.57D² - 0.42D³</td>
</tr>
<tr>
<td>2</td>
<td>-0.47 - 0.37D + 0.8D²</td>
<td>0.64 + 0.26D - 0.70D² - 0.20D³</td>
</tr>
<tr>
<td>2.5</td>
<td>-0.48 - 0.38D + 0.79D²</td>
<td>0.63 + 0.30D - 0.67D² - 0.27D³</td>
</tr>
<tr>
<td>3</td>
<td>-0.55 - 0.33D + 0.76D²</td>
<td>0.63 + 0.31D - 0.63D² - 0.33D³</td>
</tr>
<tr>
<td>3.5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.2: Target response b of EML

### 5.4.3 An adaptive algorithm for EML

This section examines how EML receiver can be implemented in a practical system. An adaptive algorithm to obtain the optimum feedforward filter and target response is considered. In order to make the receiver structure practical, the procedure of choosing the EML settings must be made adaptive since the channel pulse response is not known prior to the start of transmission.

However, finding the optimum settings that maximize the performance of the EML in adaptive way is difficult because performance evaluation involves the exhaustive search of the minimum error distance. Instead, we adopt an adaptive method that reaches the minimum $\frac{\sigma^2_{EML}}{||b||^2}$ as an alternative. Therefore, the standard Least Mean Square (LMS) algorithm with normalized b, along with an error signal defined in equation (5.4), is used:

$$\hat{b}_{k+1} = b_k + 2\mu_b e_k x_k$$

$$b_{k+1} = \frac{\hat{b}_{k+1}}{||\hat{b}_{k+1}||}$$

$$w_{k+1} = w_k + 2\mu_w e_k y_k$$

where $\mu_b$ and $\mu_w$ are step sizes.

Simulation results show that at high densities the equalizer settings obtained with the minimum $\frac{\sigma^2_{EML}}{||b||^2}$ maximize the performance in most cases, justifying the use of the above algorithm.

A similar adaptive algorithm achieving the optimum settings with normalization is introduced in [37]. In their work, the coefficients are updated by FFT in frequency domain, and then windowing is carried out in time domain to constrain a unit energy.
5.5 The complexity of the Viterbi detector

In the preceding sections, we have shown that the EML outperforms the PRML. However, the complexity of the EML receiver increases even at the same number of states in the Viterbi detector, because the programmable Viterbi detector should be used to match the arbitrary real valued target response, while the PRML can use the fixed Viterbi detector. Also the integer coefficients in the PRML often eliminate the need for actual multiplication in metric computation.

General description of the programmable Viterbi decoder is extensively made in [55]. The “Add-Compare-Select” operation in the Viterbi detector is the same for both PRML and EML. The only difference is the “path metric generating” operation.

The Viterbi algorithm recursively minimizes a state metric where a path metric is defined as

\[ |y_k - \tilde{y}_k|^2 = |y_k|^2 - 2 y_k \cdot \tilde{y}_k + |\tilde{y}_k|^2 \]  \hspace{1cm} (5.14)

where \( y_k \) is the received channel output and \( \tilde{y}_k \) is the channel output corresponding to the trellis path. For example, in EPR4, \( \tilde{y}_k \) takes on values \{0, \pm 2, \pm 4\}.

Since the minimization takes place for all metrics, minimizing the above path metric is equivalent to maximizing

\[ y_k \cdot \tilde{y}_k - \frac{|\tilde{y}_k|^2}{2}. \]

Once \( b \) is obtained from the adaptive algorithm, \( |\tilde{y}_k|^2 \) can be calculated. Usually, a look up table which pre-stores the last term in the above expression is used, so that no square operation is carried out. The main bottle neck in generating the path metric in EML is computation of \( y_k \cdot \tilde{y}_k \) term. Compared with PRML, EML takes on real valued \( \tilde{y}_k \). Thus, generating the path metric in EML involves real value multiplications. Actually, for example in 8 states EML, there may exist 16 distinct output levels. However, since the input level is \( \pm 1 \), the number of the absolute value level is reduced to 8. In addition, if \( b \) is scaled so that one of output levels equals to 1, then 7 real value multiplications are required to compute the path metric in each stage. The further simplification in fixed point multiplication is possible as implemented in RAM-DFE read channel chip [56].
Instead of multiplication operations, a look-up table can be used so that no multiplication is required at all. As shown in Figure 5.13, a look-up table that stores the square value of the number corresponding to the look-up table address can be used to compute a path metric (5.14). For example, if $y_k$ is represented by 8 bits, then a 256 location look-up table is used to generate the path metric. Since this look-up table can be generated in advance, it can be implemented efficiently by RAM or ROM. Therefore, the whole computation is reduced to look-up, add, compare and select and no multiplications is required at all. Overall complexity is, thus, manageable in the EML structure.

A further simplification is possible when a 1-norm metric is used instead of a 2-norm metric in (5.14). In this case, a path metric is defined as $|y_k - \bar{y}_k|$ and can be computed very easily. A 2-norm metric in (5.14) is based on the assumption that the error sequence in the Viterbi detector have a Gaussian distribution. However, in real system, the error sequence may not have a Gaussian distribution due to distortion including non-Gaussian media noise. Even though a 1-norm metric does not preserve the maximum likelihood property, this metric could replace the 2-norm metric at the expense of a little performance degradation while it reduces the computation significantly. Further analysis on the choice of the metric should be carried out under actual recording system.
5.6 Comparison between a unit tap and a unit energy constraint

Theoretically, compared to a unit tap constraint, the target response with a unit energy constraint has better performance for Maximum Likelihood (ML) receivers. In order to verify the above statement, we simplify the case neglecting noise colorization and channel misalignment effect for now. In this simplified situation, the performance of receiver with the Viterbi detector matched to the target response \( \mathbf{b} \) is bounded by

\[
\text{SNR}_{\text{EML}} = \frac{d_{\text{min}}^2}{4\sigma_e^2} \leq \frac{\| \mathbf{b} \|^2 d^2}{4\sigma_e^2} = \frac{\| \mathbf{b} \|^2}{\sigma_e^2}
\]

where \( d_{\text{min}} \) is the minimum distance of the target response \( \mathbf{b} \) and \( d \) is the Euclidian distance of two input level in PAM. In a magnetic recording system, \( d = 2 \).

There exist sufficient conditions that achieve the equality in the above equation [57]. For example, when the length of \( \mathbf{b} \) is 3: \( \mathbf{b} = [1 \ b_1 \ b_2]^T \), it can be shown that if \( b_2 \leq 0 \), then the Matched Filter Bound (MFB) is guaranteed to be attained [58], where the MFB is the “one-shot” channel SNR and represents the upper bound we can achieve.

Even though \( \frac{\sigma_e^2}{\| \mathbf{b} \|^2} \) determines merely the upper-bound of \( \text{SNR}_{\text{EML}} \), minimizing \( \frac{\sigma_e^2}{\| \mathbf{b} \|^2} \) appears more reasonable than minimizing \( \sigma_e^2 \), when maximizing the performance of the receiver. Noting that minimizing \( \frac{\sigma_e^2}{\| \mathbf{b} \|^2} \) is equivalent to minimizing \( \sigma_e^2 \) with constant \( \| \mathbf{b} \|^2 \), the above observation justifies choosing the target response with a constant energy constraint in contrast to other constraints on the target response. Similar conclusion is made in [59]. However, comparing to the solution with a unit tap constraint [46], there is little difference in the effective SNR at low recording densities. As the density increases over \( pw_{50}/T = 3 \), the EML with a unit energy constraint begins to perform better. Also it is a little easier to find a solution in a unit energy constraint case since the search is completed by changing only the decision delay \( \Delta \), while in a unit tap constraint case the search is made for both the decision delay and the position of the tap that the energy is concentrated.
5.7 Conclusion

We have proposed the EML in a magnetic storage system. EML minimizes the MSE with the target response for the given complexity. Both a unit tap and a unit energy constraints on the target response are analyzed. This system outperforms PRML on a Lorentzian channel at high recording densities. At $pu_{50}/T = 3$, with the same number of states in the Viterbi detector, performance of EML exceeds that of EPR4 by 2 dB for both constraints. Also, the adaptive algorithm to reach the optimum solution is derived for real implementation. The complexity of the Viterbi detector in EML is analyzed. A look-up table which generates a branch metric is introduced to replace the multiplication operations. In conclusion, EML can reduce the complexity for MLSD significantly while achieving near optimal performance.
Chapter 6

A Nonlinear Channel Model for Digital Storage System

As the recording density grows, nonlinear distortion in a magnetic recording system becomes dominant. In this chapter, we analyze two important nonlinear channel models: a simple partial erasure model and a data-dependent noise channel. The former model describes the effective reduction of the transition width in a recording track, while the latter model explains a channel with noise dependent to the transition of the input sequence. For both models, performance of various receivers are compared and the modified Maximum Likelihood receivers are proposed.

6.1 A Simple Partial Erasure Model

As recording densities grow in magnetic storage, nonlinear distortion becomes dominant, especially with thin-film disks. The nonlinear behavior is predominantly of two types: a transition shift, and an anomalous amplitude reduction or partial-erasure [60]. Yamauchi has shown that partial-erasure can be construed as the effective reduction of transition width across a track and has proposed a transition-width-reduction model based on both transition shift and transition-width reduction [61].

A magnetic recording channel model that can predict nonlinear distortion accurately enables one to design an improved receiver. However, if the channel model is
too complicated, the complexity of receivers for that model may be prohibitive. A simpler model is appropriate for signal processing, while a more complex model is better for the study of magnetic disk material. In this section\(^1\), we present a simple partial-erasure model, which is simplified from the transition-width-reduction model in [61]. Our model is shown to preserve the accuracy, while it is much simpler compared to the transition-width reduction model. Thus, we can design an improved detection scheme with small increase in complexity. The following receivers are considered in this section:

- Minimum Mean Square Error Decision Feedback Equalizer (MMSE-DFE)
- DFE with a tree search
- linear Partial Response Maximum Likelihood (PRML)
- nonlinear PRML
- nonlinear Equalized Maximum Likelihood (EML)

First, we introduce the simple partial-erasure channel model in Section 6.1.1. Then, in the following sections, we examine the above receivers on a nonlinear channel. The simulation results show that the nonlinear EML detector exhibits the best performance in a nonlinearity-dominant channel.

### 6.1.1 Nonlinear Channel Model

In this study, we will assume that transition shifts are compensated by appropriate time shifting of the write current reversals. Then, the noiseless output of a magnetic recording channel can be expressed as follows, using the transition-width-reduction model:

\[
y(t) = \sum_{k} r_k \cdot b_k \cdot s(t - kT) = \sum_{k} c_k \cdot s(t - kT)
\]

\(^1\text{We would like to thank Takashi Yamauchi of Mitsubishi Kasei for helpful discussions.}\)
where $r_k$ is the effective transition-width ratio, $b_k$ represents the NRZI modulated input data which is related to the binary data $a_k$ by $b_k = a_k - a_{k-1}$, $s(t)$ is the step response, $T$ is the symbol period, and $c_k = r_k \cdot b_k$. The input data, $b_k$, takes on values \{+2,0,-2\}; a non-zero $b_k$ indicates a transition.

One future and several previous values of $b_k$ are required to determine $r_k$ [61]. The structure of this nonlinear model is shown in Figure 6.1. In order to simplify our analysis, we will make a further assumption that the effective transition-width ratio, $r_k$, is determined only by the number of neighboring transitions and is expressed as follows:

$$
   r_k = \begin{cases} 
   1, & \text{if no transition occurs in adjacent neighbors} \\
   \gamma, & \text{if a transition occurs in one adjacent neighbor} \\
   \gamma^2, & \text{if a transition occurs in both adjacent neighbors} 
\end{cases}
$$

(6.2)

where $\gamma$ is a parameter which specifies the amount of partial erasure effect. For example, if $b_{k-1} = -2$ and $b_{k+1} = 0$, then $r_k = \gamma$.

The nonlinearity can be described by one parameter, $\gamma$. We will refer to this model as the simple partial erasure model and refer to the parameter, $\gamma$, as a reduction parameter. Note that when $\gamma$ equals to 1, we are left with a linear superposition model. Also as density increases, $\gamma$ decreases.

Table 6.1 compares the performance of the simple partial-erasure model and the linear superposition model as a function of normalized density, defined as $\frac{pw_{50}}{T}$ where $pw_{50}$ is the width of the step response at 50% of its peak value. The performance measure is the signal-to-distortion ratio (SDR), which is defined as

$$
   SDR = 10 \log_{10} \left[ \frac{E[y^2]}{E[(y_k - \tilde{y}_k)^2]} \right],
$$
where $y_k$ is the output of the transition-width-reduction model and $\hat{y}_k$ is the output of the simple partial model or the linear superposition model. The reduction parameter, $\gamma$, and the outputs of both models were calculated with the experimental data in [61]. This result indicates that the difference between the output of the transition-width-reduction model and that of the simple partial-erasure model is small if transition shifts are compensated. Our new model yields much more accurate estimation of nonlinear distortion than the linear superposition model does.

In the following sections, we describe several detection schemes and evaluate their performance in the presence of nonlinear distortion. Performance of each receiver is expressed in terms of the effective signal-to-noise ratio (SNR).

### 6.1.2 Minimum Mean Square Error Decision Feedback Equalizer (MMSE-DFE)

The structure and performance of the MMSE-DFE are described in Chapter 2. As seen in equation (6.1), the effective transition-width ratio, $r_k$, changes the input correlation. Defining the autocorrelation function $R_c(k) = E(c_m c_{m-k})$, it can be shown that $R_c(k)$ is

$$R_c(0) = \frac{1}{\gamma + 1}, \quad R_c(1) = -\frac{\gamma^2}{4}, \quad R_c(2) = \frac{(\gamma - 1)(\gamma + 1)^3}{8}$$

and

$$R_c(k) = 0, \text{ for } k \geq 3.$$  

However, the MMSE-DFE designed for a linear channel neglects the change of input correlation. Performance deteriorates as a result of neglecting nonlinearity. Also, the

<table>
<thead>
<tr>
<th>normalized density</th>
<th>$\gamma$</th>
<th>simple partial erasure model</th>
<th>linear superposition model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>0.90</td>
<td>41.55 dB</td>
<td>15.18 dB</td>
</tr>
<tr>
<td>1.25</td>
<td>0.83</td>
<td>32.13 dB</td>
<td>10.53 dB</td>
</tr>
<tr>
<td>1.5</td>
<td>0.775</td>
<td>26.25 dB</td>
<td>7.86 dB</td>
</tr>
<tr>
<td>1.75</td>
<td>0.73</td>
<td>22.06 dB</td>
<td>6.11 dB</td>
</tr>
<tr>
<td>2.0</td>
<td>0.70</td>
<td>18.90 dB</td>
<td>4.80 dB</td>
</tr>
</tbody>
</table>

Table 6.1: Comparison of SDR
change of input constellation (from $b_k$ to $c_k$) leads to a higher probability of error when estimating input bits in a slicer fixed to ternary input levels. Performance of the MMSE-DFE can be expressed in terms of $d_{min}$ of the input constellation.

6.1.3 DFE with a tree search

The simple partial-erasure model requires one future bit $b_{k+1}$ to determine the current bit $c_k$. It is difficult to estimate the future bit in a symbol-by-symbol detection scheme like the DFE. Thus, the DFE with a tree search is proposed to obtain a better estimation. The structure replacing the slicer and feedback path in the DFE is shown in Figure 6.2. Actually, settings for the feedforward and feedback filters are the same as in the MMSE-DFE. The structure and algorithm of the DFE with a tree search are similar to the Fixed Delay Tree Search with Decision Feedback (FDTS/DF) described in [62]. Note that we have only four tree branches in the tree since the NRZI constraint forces nonzero values of $b_k$ to alternate between +2 and -2. The basic algorithm of tree search is to look ahead in the tree and find the path that has the smallest metric. We compute the metric at each tree branch according to the values of $b_k$ and $b_{k+1}$. Actually, we "guess" the future bit and find the most likely path in the mean-square-error criterion. After estimating the current input bit, $\hat{b}_k$, the trailing ISI terms are subtracted with feedback. Since $r_k$ depends on $b_{k+1}$, two
separate feedback paths are used for two possible values of $b_{k+1}$ (i.e. $b_{k+1} = \pm 2$ or $b_{k+1} = 0$). Then new branch metrics are computed by subtracting the corresponding feedback signals from the output of the feedforward filter. For example, the upper path in the feedback loop is associated with the two upper branches in the tree, which corresponds to $\hat{b}_{k+1} = \pm 2$. If we ignore error propagation, the effective SNR is easily obtained as in [62].

6.1.4 Linear Partial Response Maximum Likelihood (PRML)

The Viterbi detector, which performs maximum likelihood (ML) sequence detection under the ideal front-end filter, sampler and discrete-time filter combination, has been used in conjunction with partial response (PR) linear equalization to increase noise immunity in data storage channels. Partial response equalization is typically achieved by a linear filter which shapes the natural channel response into a predetermined target response, whose coefficients are integer-valued. The Viterbi algorithm is then applied to the equalized channel output samples to identify the most likely input data sequence. This technique is called the PRML and is described in chapter 2. In PRML, we equalize the channel to the class IV partial response (PR4), characterized by the discrete time impulse response $1 - D^2$.

Since the NRZI modulation already includes the $1 - D$ factor, the channel in equation (6.1) should be equalized to $1 + D$.

When a Viterbi detector tuned to $1 - D^2$ is used ignoring the nonlinearity, performance deteriorates since a nonlinear channel produces the channel output different from $1 - D^2$. This situation causes a mismatched channel problem because the target channel of Viterbi detector is mismatched.

Performance degradation of linear PRML in a nonlinearity-dominant channel can be analyzed as done in section 2.2.3.

6.1.5 Nonlinear PRML

In order to estimate the current input $b_k$, we need to know both $b_{k+1}$ and $b_{k-1}$. When MLSD is used, this estimation scheme can be implemented with a delay. The
incoming current sampled channel output is considered as a future output, which is used for estimation of the previous input bit. Thus, this scheme incorporates a $D + D^2$ channel instead of $1 + D$. The corresponding trellis description considering the nonlinear distortion can be obtained as shown in Figure 6.3. In the trellis, X denotes a "don't care" input, and '*' indicates that the output is possible only for the valid input sequences. 3 digits in the left hand side of the trellis represents $b_{k-2}, b_{k-1}, b_k$, respectively, and the values in the right hand side of the trellis indicate the trellis output. Some input states are ruled out, since the non-zero input bits +2 and -2 should alternate according to the NRZI code constraint. Note that $\gamma$ illustrates the partial-erasure effect in the trellis. When $\gamma$ equals to 1, which is a linear channel case, this trellis is equivalent to the normal PR4 channel description. Thus, the modified MLSD using this trellis is proposed to compute the error metric.

Performance of nonlinear PRML is determined by the minimum distance, $d_{\text{min}}$, of
any possible output sequence pairs from the trellis description. It can be shown that the minimum distance is

\[ d_{\text{min}}^2 = 8\gamma^2. \]

The output error sequences corresponding to this \( d_{\text{min}}^2 \) are shown in Figure 6.4 with trellis.

### 6.1.6 Nonlinear Equalized Maximum Likelihood (EML)

The structure of the nonlinear EML is given in Figure 5.2, and the performance analysis of the general EML has been made in chapter 5. The goal of EML is to find an equalizer and a target channel, \( b \), that maximize performance of EML. Performance of EML is characterized by the minimum distance of \( b \) and the mean square error between the equalizer output and the output of the target channel response. The equalizer and its target response can be optimized using the Minimum Mean Square Error (MMSE) solution. Then, the Viterbi algorithm is tuned to the target response, \( b \), for detection. Since the target response, \( b \), in EML is not constrained to be minimum phase, we can equalize the channel more easily to the target response with a short equalizer. Essentially, EML subsumes PRML since the target response, \( b \), can be set to any partial response class. The derivation of the optimized settings for a feedforward filter and a target response is made in [47].
Once the target channel is chosen, a corresponding trellis description incorporating the simple partial-erasure model can be made that is similar to the trellis of Figure 6.3. Then, we add the Viterbi decoder tuned to the target channel and use the nonlinear trellis description for computing the error metric in sequence detection. The squared value of the argument of Q function in equation (A.1) is used for the effective SNR for EML as in [47] for evaluation.

6.1.7 Performance comparison and discussion

The performance comparison among receivers described in the previous section is shown in Figure 6.5. In this simulation, a Lorentzian channel is assumed so that a step response is

\[ s(t) = \frac{1}{1 + (2t/pw_{50})^2}. \]

The x-axis represents the normalized density \( \frac{pw_{50}}{T} \). The effective SNRs of several systems are computed including the performance degradation due to a colored error sequence. A feedforward filter with five taps is used and Matched Filter Bound (MFB) is set to 30 dB at \( \frac{pw_{50}}{T} = 1 \). To get a fair comparison between the DFE and sequence detection, the number of feedback taps in the DFE is set to 5. From the plot, we notice that the modified receivers that consider the nonlinear effect (such as DFE with tree search, nonlinear PRML and nonlinear EML) achieve a 3-4 dB gain over the receivers designed for the linear channel at high densities. Among all receivers that we consider in this section, the nonlinear EML exhibits the best performance. Since PRML is a specific instance of EML, performance of the nonlinear PRML is upper bounded by that of the nonlinear EML.

Also the error patterns for nonlinear and linear PRML are investigated and compared in Figures 6.6 and 6.7.\(^2\) The error event profiles are generated using 100,000 simulation data points with the noise power equal to 0.05 and \( \gamma = 0.8 \). We observe that longer error events are more likely for linear PRML than for nonlinear PRML.

\(^2\)We would like to thank Cory Modlin of Stanford University for providing the simulation program for Figures 6.6 and 6.7.
Figure 6.5: Comparison of SNR
Chapter 6. A Nonlinear Channel Model for Digital Storage System

![Error events distributions for the nonlinear PRML](image)

Figure 6.6: Error events distributions for the nonlinear PRML

We have shown that the simple partial-erasure model based on the effective reduction of a transition width predicts nonlinear distortion quite accurately. Its simpler structure leads to an efficient architecture for an improved receiver with small extra cost in implementation. Our new model is represented by a single parameter $\gamma$ and requires knowledge of one future and one past input bit. Based on this channel model, several detection schemes have been proposed, and the performance has been compared. Simulation indicates that the nonlinear EML achieves a significant improvement over other receivers. It has been shown also that receivers designed for a linear channel deteriorate severely in the presence of nonlinear distortion.
Figure 6.7: Error events distributions for the linear PRML

6.2 Data-dependent noise channel

It has been recognized by many researchers [63, 64] that the transition-dependent noise in thin-film media becomes dominant as recording density increases. Non-stationary noise characteristics may significantly degrade the performance of the receivers designed for stationary AWGN channel.

There are two main sources of data-dependent media noise. The first is non-deterministic transition shift. At the boundary of transition, inter-reaction of the magnetic material causes transition shift, depending on write patterns. The second is pulse amplitude fluctuation, caused by fluctuation of transition width with data pattern.

In this section, we have developed a method to cope with data-dependent noise. First, we present a modified maximum likelihood sequence detector that is optimal but impractical. We then modify this detector to a simpler error metric without significant loss of performance. Then, we compute the error rate of the proposed error metric using a Chi-square distribution. The error rate plots with various values
for jitter noise term show that a small error-rate reduction can be made for class-IV partial response channel model.

6.2.1 Modified detection scheme

Let \( \{x_k\} \) be an independent identically distributed (i.i.d.) input sequence supplied to channel \( h(t) \) at some symbol rate \( 1/T \). Then the channel output \( z(t) \) can be expressed as

\[
z(t) = \sum_{k=1}^{M} x_k h(t - kT) + n(t) = y(t) + n(t)
\]

(6.3)

where \( y(t) \) is noiseless output.

The discrete model sampled at \( t = kT \) is expressed in the vector form as

\[ z = y + n \]

where \( z \) is a column vector of length \( M \).

Maximum Likelihood Sequence Detection (MLSD) is defined as the choice of that \( y \) for which the probability density \( P_{z|y}(z|y) \) is maximum. The basic notion of MLSD is well described in [15].

Assuming the noise sequence \( n \) has a Gaussian distribution, the probability density function is

\[
P_{z|y}(z|y) = P_n(z - y) = \frac{1}{\sqrt{(2\pi)^N \det R}} \exp\left( -\frac{(z - y)^T R^{-1} (z - y)}{2} \right)
\]

(6.4)

where \( R \) is the covariance matrix of noise \( n \).

Since \( R \) is symmetric, we can find a unitary matrix \( U \) by similarity transformation that diagonalizes \( R \) such that \( \Lambda = U^T RU \) where \( U^T U = I \). Note that \( \Lambda \) is a diagonal matrix whose diagonal elements are eigenvalues \( \lambda_k \) of \( R \), so that \( \det R = \prod_{k=1}^{M} \lambda_k \). Inserting \( R^{-1} = U \Lambda^{-1} U^T \) into equation (6.4) and defining \( e = U^T (z - y) \), we can rearrange equation (6.4) as

\[
P_{z|y}(z|y) = \frac{1}{\sqrt{(2\pi)^N \prod_k \lambda_k}} \exp\left( -\frac{e^T \Lambda^{-1} e}{2} \right)
\]
\[
\frac{1}{\sqrt{(2\pi)^N \prod_k \lambda_k}} \exp\left(-\frac{1}{2} \sum_k \frac{e_k^2}{\lambda_k}\right)
\] (6.5)

Note that \(e_k\) is Gaussian and independent at different times, since the covariance matrix of \(e\) is \(E[ee^T] = E[U^T nn^T U] = U^T R U = \Lambda\), while original noise \(n\) is correlated. This fact will be used in analysis in the following section.

Take the logarithm on equation (6.5). Then maximizing \(P_{z|y}(z|y)\) is equivalent to minimizing
\[
\sum_{k=1}^{M} \left( \log \lambda_k + \frac{e_k^2}{\lambda_k} \right)
\] (6.6)

The expression inside the parenthesis becomes the new error metric, which replaces \((z_k - y_k)^2\) when the noise is non-white. Note that when the noise is assumed to be white and stationary, the covariance matrix \(R\) becomes an identity matrix multiplied by noise power \(\sigma^2\), thus the new error metric is reduced to \((z_k - y_k)^2\), the original error metric of the Viterbi detector. In practice, however, it is impractical to use the new error metric (6.6), since it requires computation of \(\lambda_k\) and \(e_k\) from the covariance matrix \(R\) of size \(M\) by \(M\). For a large \(M\), computational complexity prevents practical implementation at baud-rate. Therefore, the possible alternative is to assume that the noise is white (still noise variance is varying over the samples). This simplification leads to a diagonal matrix \(R\). Then the eigenvalues \(\lambda_k\) of \(R\) are equivalent to the noise power \(\sigma_k^2\) at each sampling time \(t = kT\). Now we can rewrite the new error metric (6.6) as
\[
\sum_{k=1}^{M} \left( \log \sigma_k^2 + \frac{n_k^2}{\sigma_k^2} \right)
\] (6.7)

where \(n_k = z_k - y_k\)

A further simplification can be made if we neglect a logarithm term in equation (6.7). Then another simplified error metric is
\[
\sum_{k=1}^{M} \frac{(z_k - y_k)^2}{\sigma_k^2}
\] (6.8)

In the next section, we analyze the theoretical performance based on the above modified Maximum Likelihood Sequence Detection error metrics.

\textit{Chapter 6. A Nonlinear Channel Model for Digital Storage System}
6.2.2 Performance analysis

We now calculate the probability that a particular error event will occur when we use the new error metric, and then we derive the probability of error of the normal Maximum Likelihood Sequence Detection in the presence of data-dependent noise.

Let us first analyze the case with error metric (6.6) briefly. The explicit equation will not be given here, since we point out in the previous section that this error metric is actually impractical. The probability of error is equal to \( P_e = Pr\{e \in D\} = \int_D f(e)de \), where \( D \) is the region in which the error event lies and \( f(e) \) indicates the joint probability density function of \( e \). It can be shown that the error event region \( D \) is expressed by quadratic forms consisting of \( e \)'s. Using the fact that \( e_k \) is an uncorrelated Gaussian random variable as shown in the previous section, \( f(e) = \prod_k f_k(e_k) \), where \( f_k(\cdot) \) denotes the Gaussian distribution function.

Next, we shall derive the error rate for other modified error metrics. Let \( y \) be the noiseless output vector corresponding to the correct input vector \( x \) and \( \hat{y} \) the corresponding output vector to the erroneous input vector \( \hat{x} \). The probability of error is the probability that \( \hat{y} \) has greater likelihood than \( y \) or, in terms of the previous notation,

\[
Pr\left\{ \sum_{k=1}^{m} (\log \sigma_k^2 + \frac{(z_k - y_k)^2}{\sigma_k^2}) > \sum_{k=1}^{m} (\log \hat{\sigma}_k^2 + \frac{(z_k - \hat{y}_k)^2}{\hat{\sigma}_k^2}) \right\} \quad (6.9)
\]

where \( m \) is the length of the error sequence. Define \( v_k = \hat{y}_k - y_k \) and substitute \( n_k = z_k - y_k \) into equation (6.9):

\[
Pr\left\{ \log \prod_{k=1}^{m} \sigma_k^2 + \sum_{k=1}^{m} \left( \frac{n_k}{\sigma_k^2} \right)^2 > \log \prod_{k=1}^{m} \hat{\sigma}_k^2 + \sum_{k=1}^{m} \left( \frac{n_k - v_k}{\hat{\sigma}_k^2} \right)^2 \right\}
\]

\[
= Pr\left\{ \sum_{k=1}^{m} \left( \frac{1}{\sigma_k^2} - \frac{1}{\hat{\sigma}_k^2} \right) n_k^2 - \frac{2v_k \cdot n_k}{\sigma_k^2} + \frac{v_k^2}{\sigma_k^2} > \log \prod_{k=1}^{m} \frac{\sigma_k^2}{\hat{\sigma}_k^2} \right\} \quad (6.10)
\]

Assuming that \( n_k \) is Gaussian random variable, we notice that the distribution of the left-hand side of above inequality can be derived from \textit{Chi-square} \((\chi^2)\) distribution. Following is the definition of \textit{Chi-square} distribution.

**Definition**: Let \( X_n \) be Gaussian random vector normalized by its standard deviation. Then \( Y = \|X_n\|^2 \) is \( \chi^2 \)-distributed with \( n \) degrees of
**freedom** and the probability density function of \( Y \) is

\[
f_Y(y) = \frac{1}{\Gamma(n/2) \cdot 2^{n/2} y^{n/2 - 1}} \exp(-y/2) \quad y \geq 0
\]

with \( \Gamma(x) \) = gamma function.

We can rearrange equation (6.10) to show that the LHS of equation (6.10) has a correlated *noncentral Chi-square* distribution. This correlated *noncentral Chi-square* distribution p.d.f. can be derived from the above p.d.f. The explicit expression can be found in [65]. Once the distribution p.d.f. is given, calculation of probability (6.10) is straight-forward. The above expression (6.10) has, obviously, varying coefficients for each \( t = kT \) since \( \sigma_k^2 \) and \( \hat{\sigma}_k^2 \) are different at every sampling time. Thus we now compute the probability of error by averaging over all possible \( \sigma_k^2 \) and \( \hat{\sigma}_k^2 \).

The probability of error of another error metric (6.8) can be computed in a similar manner. The only difference is that the logarithm term in equation (6.10) is neglected.

Let us compare the theoretical performance of the normal MLSD case. If we use the normal error metric \((z_k - y_k)^2\) in noise variance changing channel, the probability of error is

\[
Pr\{ \sum_{k=1}^{m} (z_k - y_k)^2 > \sum_{k=1}^{m} (z_k - \hat{y}_k)^2 \}.
\]

Using vector notation, the above equation becomes

\[
Pr\{ \|z - y\|^2 > \|z - \hat{y}\|^2 \}
\]

where \( \| \cdot \| \) denotes vector 2-norm. Define \( \mathbf{v} = \hat{y} - y \) and substitute \( \mathbf{n} = z - y \) into the above equation.

\[
Pr\{ \|\mathbf{n}\|^2 > \|\mathbf{n} - \mathbf{v}\|^2 \} = Pr\{ \mathbf{n} \cdot \mathbf{v} > \frac{1}{2} \|\mathbf{v}\|^2 \}
\]

where \( \cdot \) denotes vector inner product.

The variance of the LHS of the above inequality is \( \sum_k v_k^2 \sigma_k^2 + 2 \sum_{i,j(i \neq j)} v_i v_j C_n(i,j) \) where \( C_n(i,j) \) is cross-correlation of \( n \) equal to \( E(n_i n_j) \). Assuming a Gaussian random variable, the probability of error is

\[
Q\left( \frac{\sum_k v_k^2}{2 \sqrt{\sum_k v_k^2 \sigma_k^2 + 2 \sum_{i,j(i \neq j)} v_i v_j E(n_i n_j)}} \right) \tag{6.11}
\]
where

\[ Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-x^2/2} dx \]

At a high SNR, the above equation is dominated by minimum argument of Q function. In the following section, we set up a simple channel model and compare the performance of each case.

### 6.2.3 Channel model

In this section, the same channel model as in [66, 67] has been defined for performance analysis. The output of class-IV partial response (PR4) channel corrupted by jitter and additive noise is expressed as

\[ z(t) = \sum_{k=1}^{M} a_k h(t - kT - \Delta_k T) + n(t) \quad (6.12) \]

The jitter term \( \Delta_k \) is assumed to be a white Gaussian random variable with variance \( \sigma^2_{\Delta} \) and to be small. The input sequence \( \{a_k\} \) is converted via the formula \( a_k = (b_k - b_{k-1})/2 \) where \( b_k \) is i.i.d. with value \( \pm 1 \). Assume \( h(t) \) to be the response of the channel to a single transition. Then,

\[ h(t) = 2 \left[ \frac{\sin(\pi t/T)}{\pi t/T} + \frac{\sin(\pi(t-T)/T)}{\pi(t-T)/T} \right] \]

For every \( k \), \( h(t) \) can be expanded in a Taylor series in terms of \( \Delta_k \). Approximate the series by its first two terms. Then,

\[ h(t - kT - \Delta_k T) = h(t - kT) - \Delta_k T h'(t - kT) \]

Sample \( z(t) \) at \( t = kT \), then approximate it by leaving the most important two terms \( \Delta_k \) and \( \Delta_{k-1} \). We obtain

\[ z_k = y_k - 2a_k \Delta_k + 2a_{k-1} \Delta_{k-1} + n_k \]

where \( y_k \) is the noiseless output of the channel \( 1 - D^2 \) equal to \( b_k - b_{k-2} \). We assume that \( \Delta_k \) and \( n_k \) are uncorrelated with each other and have zero means. Since both \( \Delta_k \) and \( n_k \) are assumed to be Gaussian, the overall noise also becomes Gaussian. Then,
we can apply the algorithm developed in Section 6.2.1 which is based on a Gaussian distribution.

The assumption that $\Delta_k$ are uncorrelated with each other at each sample time is not true in general. A better model can be made if we assume that the jitter term $\Delta_k$ is also data-dependent. In this case, $\Delta_k$ may be replaced by $\Delta(i_k)$, i.e. a function of $i_k$ where $i_k$ indicates past data history. But in this section, a simple white model is assumed for analytical convenience. In our model, we find that $-2a_k \Delta_k + 2a_{k-1} \Delta_{k-1}$ term acts like the data-dependent noise. While $1-D^2$ channel, in general, can be decoded as two $1-D$ channels using the interleaving property, note that no interleaving is possible in this model, because the noise is no longer white and stationary.

Based on the above model, performance of the modified MLSD’s with error metrics (6.7) and (6.8) are analyzed for different noise variances using the equations derived in the previous section. In the next section, simulation results in the modified MLSD will be presented.

### 6.2.4 Results and Discussion

The probability of error for each error metric is simulated in Figure 6.8 with various values of $\sigma_\Delta$. In this simulation, the variance of white noise $n_k$ is set to $10^{-6}$, which is the same as in [67]. In Figure 6.8, $x$ axis indicates $\sigma_\Delta$ and $y$ axis indicates the error rate in logarithmic scale.

We have observed that the probability of error of the modified MLSD is reduced at least 10 times compared to that of the normal MLSD, and also found that there is no noticeable error rate difference whether a logarithm term in the modified MLSD is used or not. This plot confirms the previous experimental result shown in [67].

We have proposed the modified maximum likelihood sequence detector in a data-dependent noise channel. The new error metric has been derived based on non-stationary transition-dependent noise characteristics. Simplified error metrics have been suggested in order to reduce complexity. Analytical performance analysis has been made noticing that the error rate can be derived from a Chi-square distribution.
Figure 6.8: Probability of error
Simulation results show that the modified MLSD performs better than normal MLSD in jitter dominant channel. Also it is shown that the logarithm term in the new error metric can be neglected without any loss of performance and, by doing so, we can reduce complexity.
Chapter 7

Conclusion

In this dissertation, we have discussed several channel equalization techniques to improve performance in digital transmission and storage systems.

Section 7.1 summarizes the main results of the dissertation. Section 7.2 discusses areas for future research.

7.1 Summary of Results

Chapter 2 presented a digital communication channel model. As a comparison purpose, three common receivers are explained. For DFE and LE, a decision delay $\Delta$ is introduced as an explicit parameter and a method to determine the decision delay that optimizes performance is briefly explained. For PRML, a theoretic performance is evaluated to include degradation due to the effects of misequalization and noise colorization which are often encountered in practical situations.

Chapter 3 introduced a novel fast algorithm to compute the MMSE-DFE coefficients very efficiently using the DFT operations. The main innovation of the fast algorithm is to approximate a Toeplitz matrix by a circulant matrix. We derived the fast computation algorithm for a fractionally spaced case. The MMSE-DFE solutions are obtained in computationally efficient way. Channel identification problem which the fast algorithm is based on and the finite precision effect are briefly investigated.

Chapter 4 carries extensive simulations of the fast algorithm on High-speed Digital
Subscriber Loops. As a viable choice, the Discrete Multi-tone Transceiver (DMT) is considered. Exploiting similarity between the time-domain equalizer in the DMT and the DFE, the fast algorithm developed in Chapter 3 is extended to the DMT system. Various simulation results indicate that in most cases the fast algorithm solves the TEQ settings in the DMT system only a few tenth of a dB away from the optimal solutions.

Chapter 5 introduced the EML receiver in a magnetic recording channel. Two criteria (a unit tap and a unit energy constraints on the target response) were considered. Performance comparison between the EML and the PRML was made in a high density magnetic recording channel. Especially at high densities, substantial improvement over the PRML were achieved by 2 dB. As well as the adaptive algorithm to obtain the optimal EML settings, real design issue such as computing a branch metric with a look-up table was presented, so that practical implementation becomes feasible.

Chapter 6 developed two nonlinear channel models for a high density magnetic recording channel. The partial-erasure effect and the data-dependent noise accounts for most nonlinear distortion. For each nonlinear model, modified receivers were introduced and compared in nonlinear channel model. Simulations showed that modified receivers achieved substantial performance gain over linear channel receivers.

### 7.2 Topics for Future Research

The following is a partial list of issues for future research.

- *Filter length optimization in the DFE.* When the total number of taps of the feedforward and feedback filter is fixed, what are the optimum filter lengths for the feedforward and feedback filters in the DFE that maximize the output SNR?

- *Decision delay optimization in the DFE.* The closed form for the optimal decision delay is not known yet. Therefore, to optimize the decision delay, the
recursive algorithm is still required. A fast algorithm to compute the optimal
decision delay would be interesting.

- **Generalization of the fast algorithm.** In chapter 3, the fast algorithm is derived
  for the case where the feedforward filter is longer than the feedback filter. It
  would be a challenging work to generalize the fast algorithm so that no restric-
tion is applied to the length of the feedforward and feedback filter.

- **Effect of channel mis-identification on the fast algorithm.** The fast algorithm is
  based on the assumption that perfect knowledge of the channel pulse response
  is known. A more extensive study is necessary to see performance of the fast
  algorithm with imperfect channel pulse response.

- **New criterion for the EML.** In chapter 5, two criteria in the target response were
  investigated. However, those criteria may not be the best to optimize the EML
  performance. A more study on a new constraint would result in performance
  improvement in the EML.

- **Nonlinear distortion in a magnetic recording channel.** As a recording density
  grows, the nonlinear distortion becomes dominant in a thin-film magnetic disk.
  Perfect analysis of the nonlinear effect should be made to design a better receiver
  for a nonlinearity dominant magnetic channel.
Appendix A

The effective SNR in the Maximum Likelihood Sequence Detector

In this appendix, we analyze the performance of the Maximum Likelihood Sequence Detector (MLSD), taking into account the colored noise and the channel misequalization. Define the discrete sampled channel output \( z \) expressed in the vector form as

\[
z = Hx + n,
\]

where \( H \) is the channel response including the nonlinearity, \( x \) is the input sequence, and \( n \) is an additive Gaussian noise with covariance matrix \( R_n \).

Define the noiseless channel output \( y = Hx \), the mismatched channel output \( \tilde{y} = \tilde{H}x \) and the erroneous mismatched channel output \( \tilde{y} = \tilde{H} \tilde{x} \), which corresponds to the erroneous input \( \tilde{x} \). Here we denote \( \tilde{H} \) as the mismatched channel equal to \( b \), the given target response. The probability of error is the probability that in Viterbi detection, \( \tilde{y} \) is more likely than \( \tilde{y} \), or

\[
Pr\{\|z - \tilde{y}\|^2 > \|z - y\|^2\} = Pr\{\|y - \tilde{y} + n\|^2 > \|y - y + n\|^2\}
\]

\[
= Pr\{\|\tilde{y}\|^2 - 2\tilde{y} \cdot y - 2\tilde{y} \cdot n > \|\tilde{y}\|^2 - 2\tilde{y} \cdot y - 2\tilde{y} \cdot n\},
\]

where \( \| \cdot \| \) denotes vector 2-norm and \( \cdot \) represents vector inner product.
Appendix A. The effective SNR in the Maximum Likelihood Sequence Detector

Define \( e = \hat{y} - \bar{y} \) and substitute it into the above equation. Then,

\[
Pr\{2e \cdot n > \|\hat{y}\|^2 - \|\bar{y}\|^2 - 2e \cdot y\} = Pr\{e \cdot n > \frac{1}{2} e \cdot (\hat{y} + \bar{y} - 2y)\}.
\]

Since \( n \) is the only random variable, the variance of the LHS of the above inequality is \( e^TR_n e \). If we neglect the number of the nearest neighbors, the probability of error is

\[
P_e \approx Q\left(\min_{x, \hat{x}} \frac{<e, \hat{y} + \bar{y} - 2y>}{2 \sqrt{e^T R_n e}}\right),
\]

where

\[
Q(x) = \frac{1}{\sqrt{2\pi}} \int_{x}^{\infty} e^{-x^2/2} dx,
\]

and \(<\cdot, \cdot>\) represents vector inner product.

Thus, the effective SNR is the squared value of the argument of \( Q \) function. Note that when \( y = \bar{y} \), which corresponds to the perfect channel equalization case, the above expression reduces to

\[
Q\left(\min_{x, \hat{x}} \frac{\|e\|^2}{2 \sqrt{e^T R_n e}}\right),
\]  \hspace{1cm} (A.1)

which is the general expression for sequence detection as shown in [47]. Exhaustive search has to be carried out over all possible input sequence pairs \( x, \hat{x} \) to find the effective SNR of the MLSD.
Appendix B

Derivation of The Fast Algorithm

In this appendix, we follow the notation described in section 3.2. We denote $\tilde{R}_{yy}, \tilde{R}_{yx}, \tilde{R}_{xx}$ and $\tilde{R}_{x|y}$ as the approximation to the matrices $R_{yy}, R_{yx}, R_{xx},$ and $R_{x|y}$, respectively. The autocorrelation matrix $R_{yy}$ is computed as $R_{yy} = E[\tilde{y}_k \tilde{y}_k^*] + l \cdot \sigma^2 \text{I}_M$ from equation (2.2). To approximate $R_{yy}$ as a circulant matrix, we assume that $\{\tilde{y}_k\}$ is cyclic. Using equation (3.2), $E[\tilde{y}_k \tilde{y}_k^*]$ is computed as a time-averaged autocorrelation function:

$$E[\tilde{y}_k \tilde{y}_k^*] = \frac{1}{M} \sum_{k=0}^{M-1} \tilde{y}_k \tilde{y}_k^*$$

$$= \frac{1}{M} \begin{bmatrix} \tilde{y}_{M-1} & \tilde{y}_0 & \tilde{y}_1 & \cdots & \tilde{y}_{M-2} \\ \tilde{y}_{M-2} & \tilde{y}_{M-1} & \tilde{y}_0 & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tilde{y}_0 & \cdots & \cdots & \tilde{y}_{M-1} \end{bmatrix} \begin{bmatrix} \tilde{y}_{M-1}^* & \tilde{y}_{M-2}^* & \tilde{y}_{M-3}^* & \cdots & \tilde{y}_0^* \\ \tilde{y}_0^* & \tilde{y}_{M-1}^* & \tilde{y}_{M-2}^* & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tilde{y}_{M-2}^* & \cdots & \cdots & \tilde{y}_{M-1}^* \end{bmatrix}$$

$$= \frac{1}{M} \left( \frac{1}{M^*} P^* \Lambda_{\tilde{y}} P_M \right) \left( \frac{1}{M^*} P_M^* \Lambda_{\tilde{y}}^* P \right)$$

$$= \frac{1}{M^2} P^* \begin{bmatrix} \Lambda_{\tilde{y}}^{-1} & 0 & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \begin{bmatrix} I_M \\ \vdots \\ I_M \end{bmatrix} \begin{bmatrix} \Lambda_{\tilde{y}}^{-1} & 0 & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & \cdots & \cdots \end{bmatrix} P$$

$$P$$
Appendix B. Derivation of The Fast Algorithm

\[ = \frac{1}{M^2} \mathbf{P}^* \begin{bmatrix} \Lambda_{\tilde{Y}^1} & \cdots & \Lambda_{\tilde{Y}^i} \\ \vdots & \ddots & \vdots \\ \Lambda_{\tilde{Y}^l} \end{bmatrix} [\Lambda_{\tilde{Y}^1} \cdots \Lambda_{\tilde{Y}^i}] \mathbf{P} \]

where the column vector \( \tilde{Y} \) is the \( M' \) point DFT of \( [\tilde{y}_{M-1}^T \cdots \hat{y}_0^T \cdots \tilde{y}_0^T] \) and the length \( M \) column vector \( \tilde{Y}^i \) denotes the \( i \) th sub-vector of \( \tilde{Y} \): \( \tilde{Y} = [\tilde{Y}^1 \tilde{Y}^2 \cdots \tilde{Y}^l]^T \).

Similarly, \( \mathbf{R}_{yx} = E[\tilde{y}_k \tilde{x}_{k-\Delta}^*] \) is equal to \( E[\tilde{y}_k \tilde{x}_{k-\Delta}^*] \) since \( x_{k-\Delta} \) and \( n_k \) are assumed to be uncorrelated with each other, thus \( \mathbf{R}_{yx} \) is approximated by

\[
\mathbf{R}_{yx} = \frac{1}{M} \sum_{k=0}^{M-1} \tilde{y}_k \tilde{x}_{k-\Delta}^* \approx \frac{1}{M} \begin{bmatrix} \tilde{y}_{M-1} & \tilde{y}_0 & \tilde{y}_1 & \cdots & \tilde{y}_{M-2} \\ \tilde{y}_{M-2} & \tilde{y}_{M-1} & \tilde{y}_0 & \cdots & \vdots \\ \vdots & \cdots & \ddots & \ddots & \vdots \\ \tilde{y}_0 & \cdots & \cdots & \tilde{y}_{M-1} \end{bmatrix} \begin{bmatrix} x_{M-\Delta}^* & x_{M-\Delta-2}^* & \cdots & x_{M-\Delta-N-1}^* \\ x_{M-\Delta} & x_{M-\Delta-1}^* & \cdots & \vdots \\ x_{M-\Delta+1}^* & x_{M-\Delta} & \cdots & \vdots \\ \vdots & \cdots & \ddots & \vdots \\ x_{M-\Delta-2}^* & \cdots & \cdots & x_{M-\Delta-1} \end{bmatrix}
\]

\[
= \frac{1}{M} \left( \frac{1}{M'} \mathbf{P}^* \Lambda_{\tilde{y}} \mathbf{P}_M \right) \left( \frac{1}{M'} \mathbf{P}^*_M \Lambda_{\tilde{x}} \mathbf{P}_N \right)
\]

\[
= \frac{1}{M^2} \mathbf{P}^* \begin{bmatrix} \Lambda_{\tilde{y}}^1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \Lambda_{\tilde{y}}^l \end{bmatrix} \begin{bmatrix} \mathbf{I}_M \\ \vdots \\ \mathbf{I}_M \end{bmatrix} \begin{bmatrix} \Lambda_{\tilde{x}}^1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \Lambda_{\tilde{x}}^l \end{bmatrix} \mathbf{P}_N
\]

\[
= \frac{1}{M^2} \mathbf{P}^* \begin{bmatrix} \Lambda_{\tilde{y}}^1 \\ \vdots \\ \Lambda_{\tilde{x}}^l \end{bmatrix} [\Lambda_{\tilde{x}} \cdots \Lambda_{\tilde{x}}] \mathbf{P}_N.
\]

where \( \tilde{X} \) is the \( M' \) point DFT column vector of \( [x_{M'-\Delta-0} \cdots 0 x_{M'-\Delta-2} \cdots x_{M'-\Delta} 0 \cdots 0] \).

Thus \( \tilde{X} = [X^T \cdots X^T]^T \) where \( X \) is the DFT vector of \( [x_{M-\Delta} x_{M-\Delta-2} \cdots x_0 x_{M-1} \cdots x_{M-\Delta}] \).

Defining \( \otimes \) and \( \div \) as the element-wise multiplication and division respectively, the inverse of \( \mathbf{R}_{yy} \), along with the matrix inversion lemma, is then

\[
\begin{align*}
\mathbf{R}_{yy}^{-1} & = (E[\tilde{y}_k \tilde{y}_k^*] + l \cdot \sigma_n^2 \mathbf{I}_{M'})^{-1} \\
& = \frac{1}{M^2} \mathbf{P}^* \begin{bmatrix} \Lambda_{\tilde{y}}^1 \\ \vdots \\ \Lambda_{\tilde{x}}^l \end{bmatrix} [\Lambda_{\tilde{x}} \cdots \Lambda_{\tilde{x}}] \mathbf{P}_N.
\end{align*}
\]
\[ \begin{align*}
&= \left\{ \frac{1}{M'^2} P^* (\Gamma_Y \Gamma_Y^* + M' \sigma_n^2 I_{M'}) P \right\}^{-1} \\
&= P^* (\Gamma_Y \Gamma_Y^* + M' \sigma_n^2 I_{M'})^{-1} P \\
&= P^* \left( \frac{1}{M' \sigma_n^2} I_{M'} - \frac{1}{M' \sigma_n^2} \Gamma_Y \left( \frac{1}{M' \sigma_n^2} \Gamma_Y^* \Gamma_Y + I_M \right)^{-1} \Gamma_Y^* \frac{1}{M' \sigma_n^2} \right) P \\
&= \frac{1}{M' \sigma_n^2} P^* (I_{M'} - \Gamma_Y \Lambda_{\Theta}^{-1} \Gamma_Y^*) P
\end{align*} \]

where \( \Gamma_Y = [\Lambda_Y \cdots \Lambda_Y]^T, \Theta = \tilde{\Theta} + M' \sigma_n^2 I_M \), and \( \tilde{\Theta} = \sum_{i=1}^l \| \tilde{Y}^i \|^2 \). Here \( \| \cdot \|^2 \) is defined as the element-wise norm square: \( \|[a_0 a_1 \cdots a_{M-1}]^T\|^2 = |a_0|^2 |a_1|^2 \cdots |a_{M-1}|^2 \). Also, note that \( \Gamma_Y^* \Gamma_Y = \Lambda_{\Theta} \).

For simplicity, we assume that the input sequence \( \{x_k\} \) is a white sequence: \( R_{xx} = \tilde{\xi}_x I_{N+1} \). This assumption can be satisfied by setting \( |X_i|^2 = \tilde{\xi}_x M \) for all \( i \), which implies a constant amplitude zero autocorrelation (CAZAC) sequence.

Substituting the above equations into equation (2.8) yields

\[
\begin{align*}
\tilde{R}_{yy}^{-1} \tilde{R}_{yz} &= \frac{1}{M'^2 \sigma_n^2} P^* (\Gamma_Y \Gamma_Y^* - \Gamma_Y \Lambda_{\Theta} \Theta \Theta \Gamma_Y^*) P_N \\
&= \frac{1}{M'^2 \sigma_n^2} P^* (\Gamma_Y \Lambda_{(\Theta \Theta)\Theta} \Theta \Theta \Gamma_Y^*) P_N \\
&= \frac{1}{M'} P^* (\Gamma_Y \Lambda_{\Theta}^{-1} \Gamma_Y^*) P_N
\end{align*}
\]

Also, using the above results, matrices in equation (3.1) become

\[
\begin{align*}
\tilde{R}_{xy} &= \tilde{\xi}_x I_{N+1} - \frac{1}{M'^2} P_N^* \Gamma_Y \Gamma_Y^* \Gamma_Y \Lambda_{\Theta}^{-1} \Gamma_Y^* P_N \\
&= \frac{1}{M'^2} P_N^* (M' \tilde{\xi}_x I_{M'} - \Gamma_Y \Lambda_{\Theta} \Theta \Theta \Gamma_Y^*) P_N \\
&= \frac{\tilde{\xi}_x}{M'} P_N^* (I_{M'} - \Gamma_I \Lambda_{\Theta} \Theta \Theta \Gamma_Y^*) P_N
\end{align*}
\]

(B.1)

where \( \Gamma_Y = [\Lambda_X \cdots \Lambda_X]^T \) and \( \Gamma_I = [I_M \cdots I_M]^T \).

From the equation (2.7), the feedback filter \( b \) can be computed directly. Since \( \tilde{R}_{xy} \) is block Toeplitz, its inversion can be carried out efficiently using the Levinson-Trench algorithm using \( O(l^3 N^2) \) recursions as in [21]. However, implementing the Levinson-Trench algorithm is still not attractive in terms of complexity and cost in real time applications. Thus we make a further approximation in inversion of matrix
in equation (2.7), so that the whole computation can be done with only the DFT. Here the inverse of $\hat{R}_{x|y}$ is approximated by

$$
\hat{R}_{x|y}^{-1} = \frac{1}{\epsilon_x M'} P_N^* (I_{M'} - \Gamma_I \Lambda_{\theta} \Theta \Gamma_I^*)^{-1} P_N
$$

$$
= \frac{1}{\epsilon_x M'} P_N^* (I_{M'} - \Gamma_I (\Gamma_I^* \Gamma_I - \Lambda_{\theta})^{-1} \Gamma_I^*) P_N
$$

$$
= \frac{1}{\epsilon_x M'} P_N^* (I_{M'} + \frac{1}{M' l^2 \sigma_n^2} \Gamma_I \Lambda_{\theta} \Gamma_I^*) P_N
$$

This approximation turns out to be quite accurate from the simulation as will be shown later. Note that a scalar constant of $\hat{R}_{x|y}^{-1}$ is not important, since scaling takes place in equation (2.7).

Plugging the above equation into equation (2.7) yields

$$
b = \frac{1}{k} P_N^* (I_{M'} + \frac{1}{M' l^2 \sigma_n^2} \Gamma_I \Lambda_{\theta} \Gamma_I^*) I_{M'}
$$

$$
= \frac{1}{k} P_N^* (\Gamma_I I_M + \frac{1}{M' l^2 \sigma_n^2} \Gamma_I \Lambda_{\theta} I_M)
$$

$$
= \frac{1}{k'} P_N^* \Gamma_I \Lambda_{\theta} I_M
$$

$$
= \frac{1}{k'} P_N^* \Gamma_I \Theta
$$

where $k'$ is a scaling constant to make $b$ monic ($b_0 = 1$).

Once $b$ is obtained, we can compute the feedforward filter $w$ from equation (2.8):

$$
w = \frac{1}{M'} P^* Y \Lambda_{\theta}^{-1} X^* \begin{bmatrix}
B \\
\vdots \\
B
\end{bmatrix}
$$

$$
= \frac{1}{M} P^* \Gamma_X \begin{bmatrix}
\tilde{Y}^1 \otimes X^* \otimes B \otimes \Theta \\
\vdots \\
\tilde{Y}^l \otimes X^* \otimes B \otimes \Theta
\end{bmatrix}
$$

$$
= \frac{1}{M} P^* \tilde{Y} \otimes \tilde{X}^* \otimes \begin{bmatrix}
B \otimes \Theta \\
\vdots \\
B \otimes \Theta
\end{bmatrix}
where the column vector \( B \) is the \( M \)-point DFT of [1 \ \ 0 \ \ \cdots \ \ 0].

Without loss of generality, we can represent the noiseless channel output sequence \( \{ \tilde{y}_k \} \) in discrete frequency domain as a column vector form:

\[
\tilde{Y} = H^* \otimes P_\Delta \otimes \tilde{X}
\]

(B.2)

where the column vector \( H \) represents the \( M' \)-point DFT of the channel impulse response \( \{ h_k \} \), \( P_\Delta = [\tilde{P}_\Delta^T \cdots \tilde{P}_\Delta^T]^T \) and \( \tilde{P}_\Delta = [1 \ e^{-j2\pi\Delta/M} \ e^{-j2\pi2\Delta/M} \ \cdots \ e^{-j2\pi(M-1)\Delta/M}]^T \). Here \( \tilde{P}_\Delta \) accounts for the decision delay \( \Delta \).

Substituting \( \tilde{Y} \) into the above equations yields

\[
b = \frac{1}{k'} P_N^* \begin{bmatrix} \Theta \\ \vdots \\ \Theta \end{bmatrix}
\]

(B.3)

and

\[
w = P^* \tilde{\epsilon}_x H^* \otimes P_\Delta \otimes \begin{bmatrix} B \otimes \Theta \\ \vdots \\ B \otimes \Theta \end{bmatrix}
\]

(B.4)

where \( \Theta = \tilde{\epsilon}_x M \sum_{i=1}^{M'} (||H^i||^2) + M' \sigma^2 \mathbf{1}_M \) and the length \( M \) column vector \( H^i \) denotes the \( i \) th sub-vector of \( H \): \( H = [H^1^T \ H^2^T \ \cdots \ H^{M'}^T]^T \).

Defining \( \bar{B}_i \) as

\[
\bar{B}_i = \frac{1}{k'} \Theta_i \mod M, \quad i = 0, 1 \cdots M' - 1
\]

(B.5)

where \( \Theta_k \) indicates the \( k \) th element of the vector \( \Theta \), the feedback filter can be obtained from the IDFT operation in equation (B.3):

\[
b_k = \frac{1}{M'} \sum_{i=0}^{M'-1} \bar{B}_i e^{j2\pi ik/M}, \quad k = 0, 1, \cdots N.
\]

(B.6)

To compute \( w \), we multiply both sides of equation (B.4) by \( P \):

\[
W_i = \frac{\tilde{\epsilon}_x M H_i^* P_{\Delta,i} \bar{B}_i \mod M}{\Theta_i \mod M}, \quad i = 0, 1, \cdots M' - 1
\]

(B.7)

where \( H_i \) and \( B_i \) are the \( i \) th element of the DFT column vectors \( H \) and \( B \), respectively.
From this equation, we can obtain $w_k$ using the IDFT:

$$w_k = \frac{1}{M'} \sum_{i=0}^{M'-1} W_i e^{j2\pi ik/M'}, \quad k = 0, 1, \ldots, M' - 1.$$
Bibliography


[34] ITU-T (CCITT) Recommendation V.34, A modem operating at data signaling rates of up to 28800 bit/s for use on the general switched telephone network and on leased point-to-point 2-wire telephone-type circuits, 1994.


Bibliography


