Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be "used for any purpose other than private study, scholarship, or research." If a, user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of "fair use" that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select "Pages from: first page # to: last page #" on the print dialog screen

Order Number 9426995

Blind detection in channels with intersymbol interference

Kamel, Raafat Edward, Ph.D.

New Jersey Institute of Technology, 1994

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

ABSTRACT

BLIND DETECTION IN CHANNELS WITH INTERSYMBOL INTERFERENCE

by
Raafat Edward Kamel

In high speed digital transmission over bandlimited channels, one of the principal impairments, besides additive white Gaussian noise, is intersymbol interference. For unknown channels, adaptive equalization is used to mitigate the interference. Different types of equalizers were proposed in the literature such as linear, decision feedback equalizers and maximum likelihood sequence estimation. The transmitter embeds sequences with the data regularly to help the equalizer adapt to the unknown channel parameters.

It is not always appropriate or feasible to send training sequences; in such cases, self adaptive or blind equalizers are used. The past ten years have witnessed an interest in the topic. Most of this interest, however, was devoted to linear equalization

In this dissertation we concentrate on blind decision feedback equalization and blind maximum likelihood sequence estimation. We propose a new algorithm: the "decorrelation algorithm," for controlling the blind decision feedback equalizer. We investigate properties such as convergence and probability of error.

A new algorithm is also proposed for blind maximum likelihood sequence estimation. We use two trellises: one for the data and the other for the channel parameters. The Viterbi algorithm is used to search the two trellises for the best channel and data sequence estimates. We derive an upper bound for this scheme.

We also address the problem of ill convergence of the constant modulus algorithm and propose a technique to improve its convergence. Using this technique,

global convergence is guaranteed as long as the channel gain exceeds a certain critical value.

The question of the Viterbi algorithm's complexity is important for both conventional and blind maximum likelihood sequence estimation. Therefore, in this dissertation, the problem of reducing the complexity of the Viterbi algorithm is also addressed. We introduce the concept of state partitioning and use it to reduce the number of states of the Viterbi algorithm. This technique offers a better complexity/performance tradeoff than previously proposed techniques.

BLIND DETECTION IN CHANNELS WITH INTERSYMBOL INTERFERENCE

by
Raafat Edward Kamel

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy

Department of Electrical and Computer Engineering

May 1994

APPROVAL PAGE

BLIND DETECTION IN CHANNELS WITH INTERSYMBOL INTERFERENCE

Raafat Edward Kamel

Dr. Y. Bar-Ness, Dissertation Advisor — Date
Distinguished Professor of Electrical and Computer Engineering,
NJIT

Dr. D. Blackmore, Committee Member — Date
Professor of Math.
NJIT

Dr. A. Haimovich, Committee Member — Date
Associate Professor of Electrical and Computer Engineering,
NJIT

Dr. J. Mazo, Committee Member — Date
Department Head.
AT&T Bell Labs

Dr. Z. Siveski, Committee Member — Date
Assistant Professor of Electrical and Computer Engineering,
NJIT

BIOGRAPHICAL SKETCH

Author: Raafat Edward Kamel

Degree: Doctor of Philosophy

Date: May 1994

Undergraduate and Graduate Education:

- Doctor of Philosophy in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 1994
- Master of Science in Electrical Engineering,
  New Jersey Institute of Technology, Newark, NJ, 1990
- Bachelor of Science in Electrical Engineering,
  University of Khartoum, Sudan, 1987

Major: Electrical Engineering

This dissertation is dedicated to
my parents for their love, prayers and support.

ACKNOWLEDGMENT

My graduate study period at NJIT has been the most challenging and interesting part of my life. I would like to thank all those who made it possible for me to finish my Ph. D. studies.

The utmost gratitude goes to my advisor, Professor Yeheskel Bar-Ness, whose valuable advice, encouragement, and friendship made this work possible. He constantly provided suggestions when I was bouncing around from one problem to the other, which made me settle down and solve one problem at a time. He gave me the freedom to expand my horizons but at the same time, he was always there to guide me away from dead ends. I would also like to thank him for the time he spent on the other end of the computer network offering his feedback during his sabbatical leave at Delft. He is a brilliant teacher.

Sincere thanks also go to Professor Denis Blackmore, Professor Alex Haimovich, Dr. Jim Mazo, and Professor Zoran Siveski for serving in my dissertation committee. Professor Zoran Siveski has given me several comments on my work and was very generous with his books.

My gratitude is extended to Dr. Peter Rha, Mr. Stan Kay, Dr. Andrew McDonald and Dr. Youngky Kim of Hughes Network Systems for giving me a fruitful and worthwhile time during my internship. Dr. Peter Rha provided valuable advice as well as an education in Wireless Communication Systems. His questions were always very challenging to answer. Mr. Stan Kay and Dr. Andrew McDonald helped with the frequency planning assignment at Hughes and with Simscript. Dr. Youngky Kim introduced me to the problem of reduced complexity sequence estimation.

I would like to thank the Acting Chair and the personnel of the Electrical and Computer Engineering Department at NJIT for offering help in many forms during my studies. I am indebted to Ms Lisa Fitton for being a constant and patient proof

reader, despite of her busy schedule. I appreciate the time she took listening to my problems. I would like to thank my fellow graduate students at the Center for Communications and Signal Processing Research, who have created an enjoyable working environment for me. I enjoyed Mr. Andrew Bateman's interesting and challenging questions regarding his MS thesis work. I am grateful to Mr. Christopher Peckham's help with the LaTeXstyle sheet and for always being there to answer questions and solve problems on computers in general.

My love and thanks are extended to my parents, my brother and sister and their families for their constant love, prayer and support. I would to like to express my gratitude to my fiancée Hana Saba, for her understanding and for excusing me when I was on the edge over my research.

I would also like to thank Father Athanasious Farag, pastor of St. Mina and St. Antonious Coptic Orthodox Church of East Rutherford–for his prayer, wisdom and advice in both spiritual and personal life.

Last but not least I would like to express my sincere gratitudes to all my friends especially, Joseph Aboukhalil, Bob Czech, Michael Hajj, Georges Kalife, and Rindala Saliba for their friendship and help.

This work was supported in part by a grant from Rome Air Force Lab (AFSC, Griffiss Air Force Base, NY under contract F30602-88-D-0025, Task C-2-2404) and by the 1993/94 IEEE Communication Society Student Scholarship.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Many communication channels are subject to dispersion to a greater or lesser degree. The received signal is spread or dispersed in time. This effect results in a non-ideal frequency response in the form of a non-constant amplitude and/or non-linear phase.

An example of this includes bandwidth efficient digital communication systems, where the effect of each transmitted symbol extends beyond one symbol period. The distortion caused by the overlapping symbols is called intersymbol interference (ISI). This distortion limits the speed of reliable transmission on band-limited channels.

Severe distortion occurs in radio channels, such as mobile radio and terrestrial microwave systems, due to multipath. In such channels more than one path exists between the receiver and the transmitter, each arriving with different propagation delays. In the mobile environment, multipath occurs due to reflection from buildings, moving objects, *etc.* In terrestrial microwave telecommunication systems, multipath arises from reflection off the ground and atmospheric refractions.

Multipath channels are characterized by a delay spread which is defined as the reciprocal of the coherence bandwidth. When the delay spread exceeds the symbol period, frequency-selective fading produces ISI [1]

1.1 Adaptive Equalization

Equalization is used to mitigate ISI in data communications systems. Figure 1.1 depicts a model of a data communication system. Equalization is defined as the problem of restoring the transmitted symbols ($\{I_k\}$) by processing the output of the channel. Since the channel ($H(z^{-1})$) is usually unknown to the transmitter and the receiver, a form of adaptive equalization is always considered. The past three decades

have witnessed progress in the theory and applications of adaptive equalization [2] [3]. Equalization techniques can be divided into two categories: linear and nonlinear [4].

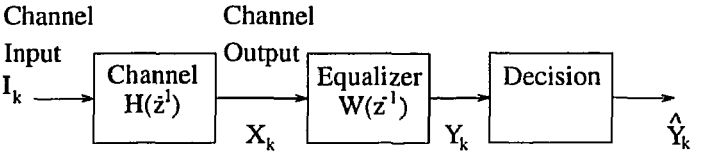

Figure 1.1 Channel Equalization

A linear equalizer is implemented using a transversal filter whose weights are adaptively adjusted according to a certain criterion. This criterion is, in general, a minimization of a given cost function $\Phi(\cdot)$ (The derivative of the cost function with respect to weight vector $\phi(\cdot)=\Phi'(\cdot)$ is known as the error function.). Referring to Figure 1.1, the adaptation rule can be expressed as

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \mu \frac{\partial \Phi(I_k, X_k, Y_k)}{\partial \mathbf{W}(k)},$$

where $\mathbf{W}(k)$ is the equalizer weight vector at the kth instant. A common criterion is the least mean squared (LMS) error between the desired output and the actual output. The rate of convergence of such an adaptive equalizer is determined by the eigenvalue spread of the input covariance matrix [5]. An improved algorithm was developed in [6] [7], which is based on the recursive least squares (RLS) algorithm. The algorithm minimizes a weighted sum of squared errors. Compared to the LMS, the RLS converges faster at the expense of computational complexity. In effect the RLS replaces the constant gain μ of the LMS update equation by a variable gain. The complexity is reduced through the use of other RLS-based algorithms, which include the so-called "square-root RLS algorithm" [8] and the "fast RLS algorithm" [9]. Another technique to increase the speed of convergence is to orthogonalize the received signal. This is done using a lattice filter [10][11] [12].

For a severely distorted channels, the linear equalizer cannot effectively cancel the ISI. For a channel with spectral nulls, the linear equalizer compensates for the distortion by placing a high gain on that frequency range, thus, enhancing noise at that frequency.

It was this type of channel that motivated the use of nonlinear equalization techniques. These include decision feedback equalization (DFE), maximum aposteriori probability (MAP) and maximum likelihood sequence estimation (MLSE).

The basic idea of the DFE is to cancel the ISI caused by the previously detected data [4][13]. The DFE has a forward filter for cancelling the precursor and a feedback filter to cancel the postcursor. The weights of the DFE can also be adaptively controlled using either the LMS [14] or the different versions of the RLS and lattice type [1]. Because of the structure of the DFE, it can cancel ISI with minimal enhancement of noise. A major problem with the DFE is error propagation. If an error is made in the decision, it will propagate down the feedback filter and, therefore result in residual ISI and a reduced margin against noise at future decisions [1][15]. DFE is still an active area of research where there are different attempts to reduce the effect of error propagation [16]. In [16] a block decision feedback equalizer is proposed, where the equalization is performed on a block of data samples rather than one sample. By varying the length of the block, the block decision feedback equalizer can emulate different forms of equalization ranging from the conventional DFE to maximum likelihood sequence estimation.

In [17] an algorithm is developed that is based on the *maximum a posteriori* probability criterion. The technique is optimum in the sense of minimizing the probability of symbol error. The performance of the MAP is superior to that of the DFE. However, the large computation burden of this technique is its major disadvantage.

Another estimation technique that has been proposed for ISI channels is the maximum likelihood sequence estimation (MLSE) [18]. This technique is based on maximizing the likelihood function of the received sequence. The Viterbi algorithm (VA), which is used for decoding convolutional codes, is also used here. An adaptive version of the VA was proposed in [19], where the channel parameters are first estimated and then the estimated parameters are used in the metric calculation of the VA. The main disadvantage of the VA is that its complexity grows exponentially with the channel span. Recently there has been a lot of research concerned with reduced complexity VA which compromises performance for complexity [20][21][22].

All of the above adaptive equalization techniques use training sequences sent by the transmitter to help the equalizer adapt to the unknown channel. Such an approach is not always appropriate or feasible [23]. The process of embedding training sequences in the transmitted data complicates the transmitter design. A remedy for this problem is to use blind equalization, *i.e.*, adaptation to the channel without the use of a training sequence.

The blind equalization problem is more formally defined as that of recovering the original input signal to an unknown system based on the observation of the system's output and some of the characteristics of the input signal. Blind equalization is the main focus of this research.

1.2 Blind Equalization

The problem of blind equalization is that of finding an appropriate cost function (or equivalent error function) that reflects the amount of ISI introduced by the channel, and which does not involve the transmitted symbols [23][24][25][26][27]. Optimization of the cost function should lead to minimization of the ISI. In other words, optimization of such a function should be consistent with the minimization

of ISI. In what follows, we give a review of different error functions considered in the literature.

The first known blind equalization algorithm was introduced by Sato [24]. Sato's error function is given by

$$\phi(Y_k) = Y_k - R_1 \mathrm{sgn}(Y_k), \tag{1.1}$$

where R_1 is defined as $R_1 \triangleq \frac{E|I_k|^2}{E|I_k|}$, I_k is the transmitted symbol. The above error function was later generalized by Benveniste *et al.,* [25] into a class of error functions given by

$$\phi(Y_k) = \psi(Y_k) - R\ \mathrm{sgn}(Y_k), \quad R \triangleq \frac{E\{\psi(I_k)I_k\}}{E\{|I_k|\}}, \tag{1.2}$$

where $\psi(x)$ is an odd, twice differentiable function, with $\psi''(x) \geq 0, \forall x \geq 0$. The function $\psi(\cdot)$ generalizes Sato's linear function.

Sato's cost function can be written as $\Phi(Y_k) = \frac{1}{2}(|Y_k| - R_1)^2$. Godard [23] then described a class of cost functions given by

$$\Phi(Y_k) = \frac{1}{2p}\left(|Y_k|^p - R_p\right)^2, \qquad p = 1, 2, \cdots, \tag{1.3}$$

where $R_p \triangleq \frac{E\{|I_k|^{2p}\}}{E\{|I_k|^p\}}$. It is clear that for $p = 1$, Godard's cost function is that of Sato. It is also worth mentioning that for $p = 2$ the algorithm is the constant modulus algorithm (CMA) developed separately by Treichler *et al.,* [28][29].

Bellini *et al* [26] followed a different approach and developed what they termed "Bussgang Techniques." Based on some assumptions about the equalizer and the channel parameters, they derived a maximum likelihood estimator of the reference signal. This estimator depends on the type of modulation used and the signal-to-noise ratio (SNR). Writing the equalizer's output Y_k as

$$\begin{aligned} Y_k &= f_\mu I_{k-\mu} + \sum_{i \neq \mu} f_i I_{k-i} \\ &= f_\mu I_{k-\mu} + n_k, \end{aligned}$$

where f_k is the convolution of the channel and equalizer ($f_k = \{h_k * w_k\}_k$, h_k and w_k being the channel and equalizer parameters, respectively), $|f_\mu| \triangleq \max|f_k|$ and n_k is the ISI. Bellini *et al* [26] derived a function $g(Y_k)$ defined as

$$g(Y_k) \triangleq \hat{I}_{k-\mu}^{ML}$$

Under the assumption that n_k is normally distributed with a zero mean and variance of σ^2, and $M-$ary PAM (M is even) alphabet with I_k is uniformly distributed over $[-(M-1)d, -(M-3)d, \cdots, -d, d, \cdots (M-3)d, (M-1)d]$, $g(Y_k)$ is given by

$$g(Y_k) = \frac{\sum_{i=1}^{M/2}(2i-1)de^{\frac{f_\mu^2 d^2(2i-1)^2}{2\sigma^2}}\sinh\left(\frac{f_\mu d(2i-1)}{\sigma^2}Y_k\right)}{\sum_{i=1}^{M/2}e^{\frac{f_\mu^2 d^2(2i-1)^2}{2\sigma^2}}\cosh\left(\frac{f_\mu d(2i-1)}{\sigma^2}Y_k\right)}.$$

$g(Y_k)$ would be used as an estimate of $I_{k-\mu}$, thus, the weights update equation would be

$$\mathbf{W}(k+1) = \mathbf{W}(k) + \mu\,(g(Y_k) - Y_k)\,\mathbf{X}_k, \tag{1.4}$$

where $\mathbf{X}_k$ is the vector of the channel outputs.

The error function of the Bussgang algorithm may be written as

$$\phi(Y_k) = Y_k - g(Y_k),$$

which has the same form as the Sato, Benveniste and Godard algorithms. For this reason all of the aforementioned algorithms are considered as special cases of the Bussgang technique.

All of the above functions are non-convex which, in turn, imply the existence of local minima to which the blind equalizer might converge. Some of these equilibria may be undesirable, *i.e.*, at those equilibria the equalizer will not be able to remove ISI. This was shown and demonstrated by Ding *et al.,* for the Godard algorithm [30][31] and for the Sato algorithm [32]. In [33], the ill convergence of the Benveniste *et al* algorithms [25] was also considered, thus proving that none of the previous algorithms were globally convergent. For these algorithms equalizer initialization

becomes an important issue. One would initialize the equalizer away from the neighborhood of the ill-convergent minima.

Verdu, *et al.*, [34] developed a technique that insures global convergence of blind equalizers. The key observation in [35] is that overparameterizing the blind equalizer is the prime cause of ill-convergence, and anchoring (setting the first coefficient to one) the blind equalizer is proposed [34][36]. This together with using a convex function, guarantees convergence. Verdu used the minimized energy as a cost function. Vembu *et al.*, [36] used the l_1 norm of the equalizer weight as a cost function, which was approximated by the l_p norm of the equalizer output. Kennedy and Ding, [37] applied the concept of anchored equalization to a QAM transmission. Due to the complex nature of the signal constellation, they performed joint equalization and carrier recovery. This was done by anchoring the sum of the real and imaginary parts of the center tap to 1, and using the maximum of either the real or the imaginary part of the equalizer output as a cost function. The cost function was implemented using the l_p norm of the real or imaginary part of the equalizer's output.

Another family of blind equalization algorithms that appeared in the literature is that based on high-order moments and polyspectra [38] and [39]. In general, these algorithms give better performance at the expense of higher arithmetic complexity. Basically, these algorithms use the received samples to estimate the channel parameters and reconstruct the transmitted data via inverse filtering. The computational complexity of these algorithms makes them inappropriate for on-line processing.

Another technique of blind equalization is that which is based on maximum likelihood sequence estimation [40]. The channel and the data are jointly estimated. One would initially assume certain channel parameters then use that to calculate the branch metric and retain the best K surviving paths into each state. Associated with

each of the K surviving paths is a least channel estimate which is updated at every time instant. These channel estimates are then used to calculate the branch metric for the following time instant. The computational complexity of such an approach is substantially higher than the VA [18]. The storage requirement is also higher since it retains more survivors.

In [41] an iterative procedure was devised which processes a frame of received data. An initial guess of the channel is made which is used by the VA, to find the maximum likelihood estimate from the frame of received data samples. The output of the VA is then used to find a better channel estimate using a least square approach. The new channel estimate together with the received frame are then used by the VA, to obtain a better sequence estimate. The process is iterated until the channel estimate converges. The channel estimation step requires a matrix inversion of the correlation matrix of estimated data. The dimensions of this matrix are proportional to the frame length (The frame length used in [41] was 1000 data symbols). This problem was avoided in [41] by assuming that such a matrix can be closely approximated by an identity matrix, since the input data is independent and identitically distributed (iid). However this argument would only be true if the VA outputs reliable data.

Another approach similar to [41] was used in [42] to jointly recover data and estimate the parameters of an underwater channel. The Expectation Maximization algorithm [43] was used to estimate the channel instead of the least squares approach in [41].

The main emphasis of blind equalization was on the linear equalizer structure. We on the other hand, concentrate on other structures and techniques which received less attention, such as decision feedback equalization and maximum likelihood sequence estimation.

This dissertation is organized as follows, in Chapter 2 we introduce a new blind equalization criteria "decorrelation." We then develop the decorrelation blind equalization algorithm. We prove the convergence of the algorithm for autoregressive (AR) type channel. Although this channel model is not as widely used in digital communications as the moving average (MA) type, it gives good insight into the problem of convergence. The decorrelation algorithm is used in conjunction with the decision feedback equalizer. The convergence to zero ISI is investigated and established in this chapter. We also describe a faster converging form of the decorrelation algorithm by using an RLS-like decorrelation algorithm.

In Chapter 3 we study the probability of error of the blind decision feedback equalizer for the additive white Gaussian noise case. In our derivation, we use a reduced complexity state machine proposed by Duttweiler *et al.*, [15]. Lower and upper bounds on the probability of error are found.

The constant modulus algorithm exhibits ill convergence due to the multimodal nature of its error function. Ill convergence is defined as the convergence to local minima that are incapable of removing ISI. In Chapter 4 we investigate the effect of anchoring the blind equalizer on the convergence of the constant modulus algorithm. By considering the AR model, we show that the convergence of the constant modulus algorithm is improved. We demonstrate that as long as the equalizer gain exceeds a certain critical value, the algorithm is guaranteed to globally converge to the global minima. The constant modulus algorithm is also used with the decision feedback equalizer.

In Chapter 5 a new blind maximum likelihood sequence estimation technique is introduced. In this setting, we quantize the channel parameters and develop a channel trellis for the discrete channel. We propose an iterative algorithm whereby the Viterbi algorithm is used to search for the most likely data sequence and channel parameter vector. We formulate the probability of an error event for the blind

Viterbi following Forney's approach. An upper bound on the probability of bit error is then derived. Compared with other techniques, this method prevails by its reduced computational complexity.

Motivated by the work done with the probability of error of the decision feedback equalizer, we use similar concepts to reduce the complexity of the Viterbi algorithm. State partitioning concepts and their applications to reduced state sequence estimation are introduced in Chapter 6. It is shown that the state partitioning technique offers a better complexity/performance tradeoff than previously known methods.

Conclusions and future directions are given in Chapter 7.

CHAPTER 2

DECORRELATION ALGORITHM FOR BLIND EQUALIZATION

In the previous chapter, we defined the blind equalization problem as that of recovering the transmitted signal through an unknown channel based solely on the observation of the channel's output and the characteristics of the transmitted signal. Algorithms based on exploiting a special property of the original input signals are known as "property restoral" [30]. One example is the constant modulus algorithm [28], where one exploits the constant constellation of the original signal in order to adapt the blind equalizer.

In this chapter, we assume that the original data is independent and identically distributed (iid). This is a valid and widely used model. At the output of the channel, the data is no longer independent. The channel introduces the correlation in the form of the ISI. We exploit the white noise like characteristics of the original signal and adapt the blind equalizer using decorrelation. This was motivated by [44] which provided a simple test to show that an adaptive equalizer has converged to the correct settings. In [44] it was shown that if the input data is binary and iid, then the decorrelation at the output of the slicer of a decision directed equalizer is a necessary and sufficient condition for the correct convergence of the equalizer. In this chapter we show that the decorrelation at the input of a slicer is a necessary and sufficient condition for the perfect cancellation of ISI.

This chapter is organized as follows. In section 2.1, we discuss the equalization of autoregressive channels. Moving average channels are discussed in section 2.2. In section 2.3 we present a rapidly converging version of the algorithm, based on minimizing time-average correlations.

2.1 Equalization of Autoregressive Channels

2.1.1 Problem Formulation

Consider an AR(n) channel driven by an equi-probable binary sequence $\{I_k\}$. The output X_k is given by

$$X_k = gI_k + \sum_{i=1}^{n} \alpha_i X_{k-i}, \tag{2.1}$$

where g is the channel gain and α's are the channel parameters.

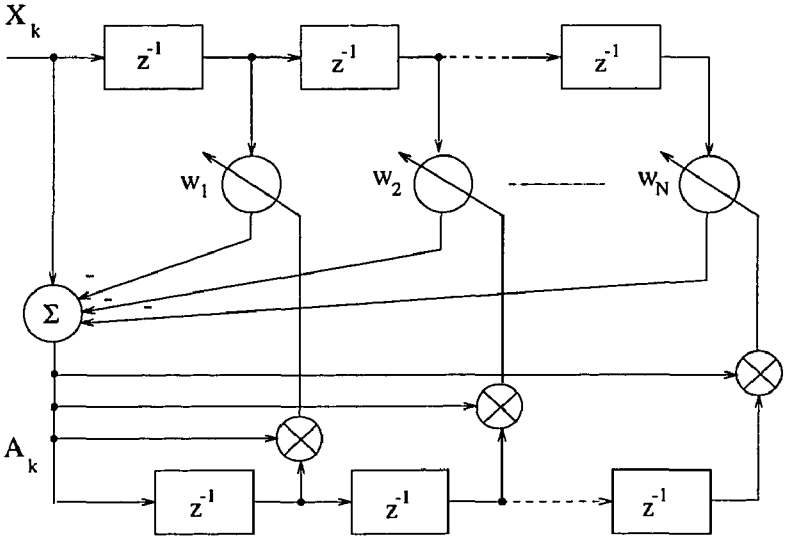

Figure 2.1 Blind Linear Equalizer

Figure 2.1 shows the anchored FIR blind equalizer. In Chapter 1, we mentioned the advantages of using an anchored equalizer [34]. The output of the equalizer, with weights $w_1, w_2, \cdots, w_n$, is given by

$$\begin{aligned} A_k &= X_k - \sum_{i=1}^{n} w_i X_{k-i} \\ &= gI_k + \sum_{i=1}^{n} (\alpha_i - w_i) X_{k-i} \end{aligned}$$

It can be shown that the above equation can be written as

$$A_k = g\left(I_k - \sum_{i=1}^{n} w_i I_{k-i}\right) + \sum_{i=1}^{n} \alpha_i A_{k-i}. \tag{2.2}$$

For the decorrelating algorithm, the weight update equation is given by

$$w_i^{k+1} = w_i^k + \mu A_k A_{k-i} \quad \text{for } i = 1, 2, \cdots, n. \tag{2.3}$$

The steady state equalizer weight is found by setting the error function function of the weight update equation (in this case $\overline{A_k A_{k-i}}$ for $i = 1, \cdots n.$) to zero. In what follows, we show that at steady state the equalizer will converge to the channel parameters and, hence, cancel ISI. Before proceeding with the proof, we derive some correlation relations.

Multiply equation (2.2) by $A_{k-(n+1)}$ and take the expectation, to get

$$\begin{aligned} \overline{A_k A_{k-(n+1)}} &= g\left(\overline{I_k A_{k-(n+1)}} - \sum_{i=1}^{n} w_i \overline{I_{k-i} A_{k-(n+1)}}\right) + \sum_{i=1}^{n} \alpha_i \overline{A_{k-i} A_{k-(n+1)}} \\ &= \sum_{i=1}^{n} \alpha_i \overline{A_{k-i} A_{k-(n+1)}}. \end{aligned} \tag{2.4}$$

The last step follows from the fact that $I_{k-i} A_{k-(n+1)}$ are independent for $i = 0, 1, 2, \cdots, n$. It further follows that

$$\begin{aligned} \overline{A_k A_{k-(n+1)}} &= \sum_{i=1}^{n} \alpha_{n-i+1} \overline{A_k A_{k-i}} \\ &= 0 \end{aligned}$$

since we require that $\overline{A_k A_{k-i}} = 0$ for $i = 1, 2, \cdots, n$ at the steady state. Similarly one can prove that $\overline{A_k A_{k-i}} = 0$ for $i > n$. Therefore, at steady state we have

$$\overline{A_k A_{k-i}} = \sigma_A^2 \delta(i), \tag{2.5}$$

where $\delta(\cdot)$ is the kronecker delta, where we assumed that A_k is wide sense stationery random process.

2.1.2 Steady State Analysis

In the Z-domain, one may write equation (2.2) as

$$\frac{1 - \alpha_1 z^{-1} - \cdots - \alpha_n z^{-n}}{1 - w_1 z^{-1} - \cdots - w_n z^{-n}} A(z) = g I(z),$$

where $A(z)$ and $I(z)$ are the Z-transforms of A_k and I_k, respectively. In the time domain, using long division, the above equation may be written as

$$\sum_{i=0}^{\infty} \gamma_i A_{k-i} = g I_k, \tag{2.6}$$

where

$$\begin{aligned}
\gamma_0 &= 1 \\
\gamma_1 &= w_1 - \alpha_1 \\
\gamma_2 &= (w_2 - \alpha_2) - w_1\gamma_1 \\
\gamma_3 &= (w_3 - \alpha_3) - w_2\gamma_1 - w_1\gamma_2 \\
&\vdots \\
\gamma_n &= (w_n - \alpha_n) - w_{n-1}\gamma_1 - w_1\gamma_{n-1} \\
&\vdots
\end{aligned}$$

Now multiply equation (2.6) by A_{k-1} and take expectation

$$\sum_{i=0}^{\infty} \gamma_i \overline{A_{k-i}A_{k-1}} = 0.$$

Using the result from equation (2.5), the above reduces to

$$\sigma_A^2 \gamma_1 = 0;$$

therefore, we get $\gamma_1 = 0$, *i.e.* $w_1 = \alpha_1$. Similarly, one can show that $\gamma_2 = 0$, which together with $\gamma_1 = 0$, gives $w_2 = \alpha_2$. Thus, showing that

$$w_i = \alpha_i \qquad \text{for } i = 1, 2, \cdots, n.$$

Therefore, at steady state, the decorrelation algorithm results in perfect ISI cancellation.

2.1.3 Simulation Results

Consider the output of an AR(1) channel given by

$$X_k = I_k + 0.9X_{k-1}.$$

The output of the anchored MA equalizer is

$$A_k = X_k - w_1 X_{k-1}.$$

The weight update equation is given by

$$w_1^{(k+1)} = w_1^{(k)} + 0.01 A_k A_{k-1}.$$

The figure below depicts the residual ISI and the equalizer tap weight. This is the result of the Monte Carlo average of 100 independent runs. The residual ISI is the mean square of the error $(I_k - \hat{A}_k)$.

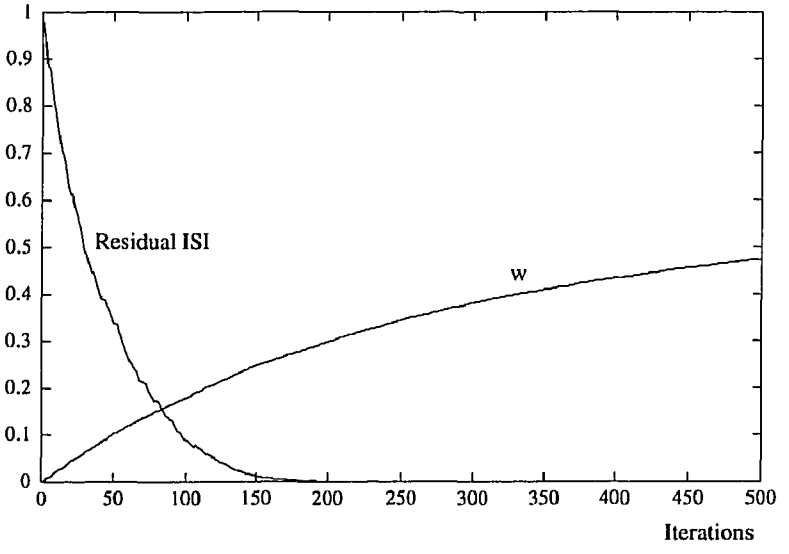

Figure 2.2 Residual ISI and Learning Curve for the Decorrelating Linear Equalizer

It is evident from the figure that the residual ISI vanishes after about 150 iterations. The convergence of the equalizer weight is also demonstrated.

2.2 Equalization of Moving Average Channels

2.2.1 Problem Formulation

The channel and equalizer model under consideration is shown in Figure 2.3. The cascade of transmit, channel and receive filters is modeled as an FIR filter with impulse response

$$h(n) = 1 + \sum_{i=1}^{N} h_i \delta(n-i),$$

where $\delta(\cdot)$ is the kronecker delta. In the above equation we normalized relative to the first cursor (h_0). We also assume that the input I_k is a binary white sequence with a zero mean. The output of the channel is thus given by

$$X_k = I_k + \sum_{i=1}^{N} h_i I_{k-i}.$$

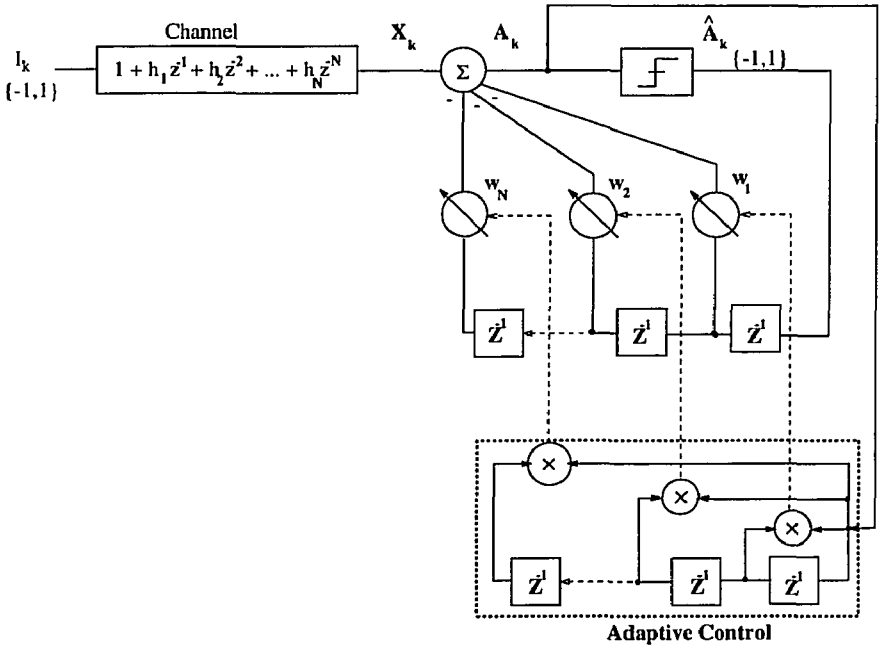

Figure 2.3 Decision Feedback Equalizer with Decorrelation Control

We assume that the channel is slowly time varying and the receiver has perfect carrier and timing recovery. The channel post-cursors $\{h_1, \cdots, h_N\}$ introduce inter-symbol interference on the current data symbol I_k. The estimated data $\hat{A}_k$ is produced by passing A_k through a slicer.

Referring to Figure 2.3, the input to the slicer of the decision feedback equalizer A_k is given by

$$\begin{aligned} A_k &= X_k - \hat{\mathbf{A}}'_{k-1}\mathbf{W} \\ &= I_k + \mathbf{I}'_{k-1}\mathbf{H} - \hat{\mathbf{A}}'_{k-1}\mathbf{W}, \end{aligned} \tag{2.7}$$

where $\hat{\mathbf{A}}_{k-1}$ is the vector of the past N decisions $\hat{\mathbf{A}}'_{k-1} = [\hat{A}_{k-1}, \hat{A}_{k-2}, \cdots, \hat{A}_{k-N}]$ (the prime stands for transpose) and $\mathbf{I}_{k-1}$ is the vector of past transmitted information bits $\mathbf{I}'_{k-1} = [I_{k-1}, I_{k-2}, \cdots, I_{k-N}]$, where $I_{k-i} \in \{-1, 1\}$ and $P\{I_{k-i} = 1\} = P\{I_{k-i} = -1\} = \frac{1}{2}$. $\mathbf{W}$ and $\mathbf{H}$ are the equalizer and channel parameter vectors, respectively;$\mathbf{W}' = [w_1, w_2, \cdots, w_N]$ and $\mathbf{H}' = [h_1, h_2, \cdots, h_N]$.

In this chapter we will assume a noiseless situation, *i.e.*, we consider an arbitrary high signal-to-noise ratio. Additive white noise will be considered in the next chapter. For ideal ISI cancellation, the slicer's input $A_k = I_k$ and therefore sequence $\{A_k\}$ will be decorrelated, *i.e.*, $\overline{A_k A_{k-n}} = 0$ for $n \neq 0$. In other words, decorrelation is a necessary condition for ideal cancellation of ISI. In order to be able to use the decorrelation of the slicer's input as a criterion for controlling the feedback weight vector $\mathbf{W}$, we must prove that decorrelation is also sufficient for cancelling ISI. This is what we intend to show in the next section.

2.2.2 Sufficiency

In order to prove sufficiency, we rewrite equation (2.7) as

$$A_k = I_k + \sum_{i=1}^{N} h_i I_{k-i} - \sum_{i=1}^{N} w_i \hat{A}_{k-i}. \tag{2.8}$$

If we denote the set of all correct decisions by A_1 and the set of all incorrect decisions by A_2, *i.e.*,

$$\begin{aligned} A_1 &= \{\hat{A}_i : \hat{A}_i = I_i\} \\ A_2 &= \{\hat{A}_i : \hat{A}_i = -I_i\}, \end{aligned}$$

then equation (2.8) can be written as

$$\begin{aligned} A_k &= I_k + \sum_{i:\hat{A}_{k-i} \in A_1} (h_i - w_i) I_{k-i} + \sum_{i:\hat{A}_{k-i} \in A_2} (h_i + w_i) I_{k-i} \\ &= I_k + \sum_{i=1}^{N} \gamma_i I_{k-i} \\ &= I_k + isi_k, \end{aligned} \tag{2.9}$$

where γ_i is given by

$$\gamma_i = \begin{cases} (h_i - w_i) & \text{for all } i : \hat{A}_{k-i} \in A_1 \\ (h_i + w_i) & \text{for all } i : \hat{A}_{k-i} \in A_2 \end{cases}$$

and

$$isi_k = \sum_{i=1}^{N} \gamma_i I_{k-i}.$$

We can now show that decorrelation is a sufficient condition for cancelling ISI. Multiply equation (2.9) by $\mathbf{A}_{k-1} = [A_{k-1}, A_{k-2}, \cdots, A_{k-N}]$, the vector of the past slicer's input, to obtain

$$A_k \mathbf{A}_{k-1} = I_k \mathbf{A}_{k-1} + \sum_{i=1}^{N} \gamma_i I_{k-i} \mathbf{A}_{k-1}.$$

Taking the expectation on both sides of the above equation, it can be shown that the above equation reduces to

$$\begin{pmatrix} \overline{A_k A_{k-1}} \\ \overline{A_k A_{k-2}} \\ \vdots \\ \overline{A_k A_{k-n}} \\ \vdots \\ \overline{A_k A_{k-N+1}} \\ \overline{A_k A_{k-N}} \end{pmatrix} = \begin{pmatrix} \gamma_1 + \sum_{i=1}^{N-1} \gamma_i \gamma_{i+1} \\ \gamma_2 + \sum_{i=1}^{N-2} \gamma_i \gamma_{i+2} \\ \vdots \\ \gamma_n + \sum_{i=1}^{N-n} \gamma_i \gamma_{i+n} \\ \vdots \\ \gamma_{N-1} + \gamma_1 \gamma_N \\ \gamma_N \end{pmatrix}. \tag{2.10}$$

It is clear from the last entry of the vectors in the above equation that $\gamma_N = 0$ iff $\overline{A_k A_{k-N}} = 0$. Similarly, it follows from the $(N-1)$th entry that if $\gamma_N = 0$, then $\gamma_{N-1} = 0$ iff $\overline{A_k A_{k-N+1}} = 0$. One would thus start from the bottom entry and use back substitution to show that $\gamma_i = 0$ for $i = 1, \cdots, N$ iff $\overline{A_k A_{k-i}} = 0$ for $i = 1, \cdots, N$. It thus follows from equation (2.9) that $isi_k = 0$ in the steady state iff $\overline{A_k A_{k-i}} = 0$ for $i = 1, \cdots, N$. This completes the proof that decorrelation is also sufficient for cancelling ISI. In the above analysis nothing was mentioned about the convergence of the equalizer weights. This point is investigated in the next section.

2.2.3 Steady State

We now consider the convergence of the weights. Multiply equation (2.7) by $\mathbf{A}_{k-1} = [A_{k-1}, A_{k-2}, \cdots, A_{k-N}]$, the vector of the past slicer's input, to obtain,

$$A_k \mathbf{A}_{k-1} = I_k \mathbf{A}_{k-1} + \mathbf{A}_{k-1}\mathbf{I}'_{k-1}\mathbf{H} - \mathbf{A}_{k-1}\hat{\mathbf{A}}'_{k-1}\mathbf{W}. \tag{2.11}$$

When taking the expectation of equation (2.11), the first term vanishes since

$$E\{A_{k-m} I_{k-n}\} = 0 \quad \text{for } m > n,$$

as A_k does not depend on the present or future data inputs. It can also be shown (Appendix A, Claim 2) that

$$E\{A_{k-m} \hat{A}_{k-n}\} = 0 \quad \text{for } m > n.$$

Therefore

$$\begin{pmatrix} \overline{A_k A_{k-1}} \\ \overline{A_k A_{k-2}} \\ \vdots \\ \overline{A_k A_{k-N}} \end{pmatrix} = \begin{pmatrix} \overline{A_{k-1} I_{k-1}} & \overline{A_{k-1} I_{k-2}} & \cdots & \overline{A_{k-1} I_{k-N}} \\ 0 & \overline{A_{k-2} I_{k-2}} & \cdots & \overline{A_{k-2} I_{k-N}} \\ \vdots & & & \\ 0 & & \cdots & \overline{A_{k-N} I_{k-N}} \end{pmatrix} \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_N \end{pmatrix}$$
$$- \begin{pmatrix} \overline{A_{k-1} \hat{A}_{k-1}} & \overline{A_{k-1} \hat{A}_{k-2}} & \cdots & \overline{A_{k-1} \hat{A}_{k-N}} \\ 0 & \overline{A_{k-2} \hat{A}_{k-2}} & \cdots & \overline{A_{k-2} \hat{A}_{k-N}} \\ \vdots & & & \\ 0 & 0 & \cdots & \overline{A_{k-N} \hat{A}_{k-N}} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{pmatrix} \tag{2.12}$$

The last entry of equation (2.12) can be written as

$$\overline{A_k A_{k-N}} = \overline{A_{k-N} I_{k-N}} h_N - \overline{|A_{k-N}|} w_N.$$

It can be shown from equation (2.7) for $(k - N)$, that

$$\overline{A_{k-N} I_{k-N}} = \overline{I^2_{k-N}} = \sigma_I^2 = 1.$$

Therefore,

$$\begin{aligned} \overline{A_k A_{k-N}} &= h_N - \overline{|A_{k-N}|} w_N \\ w_N &= \frac{h_N - \overline{A_k A_{k-N}}}{\overline{|A_{k-N}|}} \\ &= h_N - \overline{A_k A_{k-N}}, \end{aligned} \tag{2.13}$$

where we used the result of Claim 4. of Appendix A; that $\overline{|A_{k-n}|} = 1\ \forall n$. It is clear from equation (2.13) that $w_N = h_N$ iff $\overline{A_k A_{k-N}} = 0$ *i.e.* A_k and A_{k-N} are uncorrelated. Next, consider the $(N-1)$ th entry

$$\begin{aligned}\overline{A_k A_{k-N+1}} &= \overline{A_{k-N+1} I_{k-N+1}} h_{N-1} + \overline{A_{k-N+1} I_{k-N}} h_N - \overline{A_{k-N+1} \hat{A}_{k-N+1}} w_{N-1}. \\ &\quad - \overline{A_{k-N+1} \hat{A}_{k-N}} w_N\end{aligned}$$

But $\overline{A_{k-N+1} \hat{A}_{k-N+1}} = \overline{|A_{k-N+1}|} = 1$ and $\overline{A_{k-N+1} I_{k-N+1}} = \sigma_I^2 = 1$; therefore

$$\begin{aligned}\overline{A_k A_{k-N+1}} &= h_{N-1} - w_{N-1} + \overline{A_{k-N+1} I_{k-N}} h_N \\ &\quad - \overline{A_{k-N+1} \hat{A}_{k-N}} w_N \end{aligned} \tag{2.14}$$

Now, using equations (A.17) and (A.18) with $i = N$ and $m = N-1$ we get

$$\begin{aligned}\overline{A_k A_{k-N+1}} &= h_{N-1} - w_{N-1} + h_N(h_1 - w_1(1 - 2q_{k-N})) \\ &\quad + w_N(w_1 - h_1(1 - 2q_{k-N})).\end{aligned} \tag{2.15}$$

where $q_{k-i} = P\{\hat{A}_{k-i} = -I_{k-i}\}$ is the probability of error for $i = 1, \cdots, N$. In general, for the mth entry we have,

$$\overline{A_k A_{k-m}} = \overline{A_{k-m} I_{k-m}} h_m + \sum_{i=m+1}^{N} h_i \overline{A_{k-m} I_{k-i}} - w_m \overline{|A_{k-m}|} - \sum_{i=m+1}^{N} w_i \overline{A_{k-m} \hat{A}_{k-i}}.$$

Following the above argument, we get from (A.17) and (A.18)

$$\begin{aligned}\overline{A_k A_{k-m}} &= h_m - w_m + \sum_{i=m+1}^{N} h_i(h_{i-m} - w_{i-m}(1 - 2q_{k-i})) \\ &\quad + \sum_{i=m+1}^{N} w_i(w_{i-m} - h_{i-m}(1 - 2q_{k-i}))\end{aligned} \tag{2.16}$$

To summarize, we combine equations (2.13), (2.15) and (2.16) as

$$\begin{aligned}\overline{A_k A_{k-N}} &= h_N - w_N \\ \overline{A_k A_{k-N+1}} &= h_N(h_1 - w_1(1 - 2q_{k-N})) + w_N(w_1 - h_1(1 - 2q_{k-N})) + h_{N-1} - w_{N-1} \\ &\vdots \\ \overline{A_k A_{k-1}} &= \sum_{i=2}^{N} h_i(h_{i-1} - w_{i-1}(1 - 2q_{k-i})) + \sum_{i=2}^{N} w_i(w_{i-1} - h_{i-1}(1 - 2q_{k-i})) \\ &\quad + h_1 - w_1.\end{aligned} \tag{2.17}$$

In order to find the steady state equalizer weights, one would have to determine first the probability of error q_k. In section 2.2.5 we determine the probability of error q_k as a function of the equalizer weights. This is used with equation (2.17) to study the convergence of the weights and the transient behavior of the probability of error.

2.2.4 The Adaptive Control Algorithm

In order to decorrelate the A_k's, we control the equalizer weights using the steepest decent method. The weight update equation for an adaptive algorithm can be expressed as

$$w_i^{k+1} = w_i^k + \mu f(\cdot) \text{ for } i = 1, \cdots, N, \tag{2.18}$$

where μ is the constant of adaptation and $f(\cdot)$ is called the error function of the algorithm. The roots of the error function determine the steady state of the algorithm.

The previous discussion suggests using $\overline{A_k A_{k-i}}$ as a driving function for the algorithm. An appropriate error function in equation (2.18) would be $\overline{A_k A_{k-i}}$.

As a result one can write equation (2.18) as

$$w_i^{k+1} = w_i^k + \mu \overline{A_k A_{k-i}} \text{ for } i = 1, \cdots, N.$$

In a practical implementation one would replace the expectation by the current realization, leading to the stochastic difference equation,

$$w_i^{k+1} = w_i^k + \mu A_k A_{k-i} \text{ for } i = 1, \cdots, N. \tag{2.19}$$

The above analysis shows that the algorithm in equation described (2.19) will converge in the mean. That is, the mean value of w_i will converge to the channel parameter h_i.

2.2.5 Transient Analysis

In this section we examine the dynamic behavior of the proposed equalizer. We use the probability of symbol error q_k at instant k as a performance index. A difference

equation for q_k is developed, which can be solved together with the weight update equation to determine the probability of decision error as a function of time index k. The derivation given in this section can be extended to a general order N moving average type (MA) channel. However, for the sake of simplicity we consider an order 3 MA channel, which will be used in the simulation described in the next section.

The channel output X_k at the kth instant is given by

$$X_k = I_k + h_1 I_{k-1} + h_2 I_{k-2}, \tag{2.20}$$

where h_1 and h_2 are the channel parameters.

From equation (2.7) the slicer's input is given by

$$\begin{aligned} A_k &= X_k - w_1^{(k)} \hat{A}_{k-1} - w_2^{(k)} \hat{A}_{k-2} \\ &= I_k + h_1 I_{k-1} + h_2 I_{k-2} - w_1^{(k)} \hat{A}_{k-1} - w_2^{(k)} \hat{A}_{k-2}, \end{aligned} \tag{2.21}$$

where we have used the superscript k in the weights w_1 and w_2 to emphasize their dependence on time, since we are studying the transient response of the algorithm. Using equation (2.21) we will determine the probability of correct decision (q_k) as a function of the index k. Using the total probability theorem, one can write

$$\begin{aligned} q_k &= P\{\hat{A}_k \neq I_k\} \\ &= P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} \neq I_{k-1}, \hat{A}_{k-2} \neq I_{k-2}\} q_{k-1} q_{k-2} \\ &\quad + P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} \neq I_{k-1}, \hat{A}_{k-2} = I_{k-2}\} q_{k-1} p_{k-2} \\ &\quad + P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} = I_{k-1}, \hat{A}_{k-2} \neq I_{k-2}\} p_{k-1} q_{k-2} \\ &\quad + P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} = I_{k-1}, \hat{A}_{k-2} = I_{k-2}\} p_{k-1} p_{k-2}. \end{aligned} \tag{2.22}$$

Next, we evaluate each term in the right hand side of equation (2.22). The probability of an incorrect decision can be expressed as

$$P\{\hat{A}_k \neq I_k\} = \frac{1}{2}\left(P\{\hat{A}_k = 1 \mid I_k = -1\} + P\{\hat{A}_k = -1 \mid I_k = 1\}\right).$$

Expressing the first term in the RHS of equation (2.22), as above,

$$\begin{aligned} & P\left\{\hat{A}_k = 1 \mid I_k = -1, \hat{A}_{k-1} \neq I_{k-1}, \hat{A}_{k-2} \neq I_{k-2}\right\} \\ = \; & P\left\{\left(-1 + (h_1 + w_1^{(k)})I_{k-1} + (h_2 + w_2^{(k)})I_{k-2}\right) > 0\right\} \\ = \; & \frac{1}{2}\left(P\left\{I_{k-1} > \frac{1 - (h_2 + w_2^{(k)})}{|h_1 + w_1(k)|}\right\} + P\left\{I_{k-1} > \frac{1 + (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\}\right). \quad (2.23) \end{aligned}$$

In the last step of equation (2.23) we used Claim 7 of Appendix A:

$$\begin{aligned} & P\left\{\hat{A}_k = -1 \mid I_k = 1, \hat{A}_{k-1} \neq I_{k-1}, \hat{A}_{k-2} \neq I_{k-2}\right\} \\ = \; & P\left\{\left\{\left(1 + (h_1 + w_1^{(k)})I_{k-1} + (h_2 + w_2^{(k)})I_{k-2}\right) < 0\right\}\right. \\ = \; & \frac{1}{2}\left(P\left\{I_{k-1} < \frac{-1 - (h_2 + w_2^{(k)})}{|h_1 + w_1(k)|}\right\} + P\left\{I_{k-1} < \frac{-1 + (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\}\right) \quad (2.24) \end{aligned}$$

Therefore, from equations (2.23) and (2.24) one can write,

$$\begin{aligned} & P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} \neq I_{k-1}, \hat{A}_{k-2} \neq I_{k-2}\} = \frac{1}{4}\left(P\left\{I_{k-1} > \frac{1 - (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\}\right. \\ & + P\left\{I_{k-1} > \frac{1 + (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\} + P\left\{I_{k-1} < \frac{-1 + (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\} \\ & \left. + P\left\{I_{k-1} < -\frac{1 + (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\}\right). \end{aligned}$$

However, since the pdf of I_{k-1} is even, it follows that

$$P\{I_{k-1} > x\} = P\{I_{k-1} < -x\}.$$

Therefore,

$$\begin{aligned} & P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} \neq I_{k-1}, \hat{A}_{k-2} \neq I_{k-2}\} \\ = \; & \frac{1}{2}\left(P\left\{I_{k-1} > \frac{1 - (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\} + P\left\{I_{k-1} > \frac{1 + (h_2 + w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\}\right) \quad (2.25) \\ \triangleq \; & f_1(h_1, h_2, w_1^{(k)}, w_2^{(k)}). \end{aligned}$$

Similarly, for the other terms in equation (2.22) one can show:

$$P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} \neq I_{k-1}, \hat{A}_{k-2} = I_{k-2}\} = \frac{1}{2}\left(P\left\{I_{k-1} > \frac{1 - (h_2 - w_2^{(k)})}{|h_1 + w_1^{(k)}|}\right\}\right.$$

$$+ P\left\{ I_{k-1} > \frac{1+(h_2 - w_2^{(k)})}{|h_1 + w_1^{(k)}|} \right\} \Bigg)$$

$$\triangleq f_2(h_1, h_2, w_1^{(k)}, w_2^{(k)})$$

$$P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} = I_{k-1}, \hat{A}_{k-2} \neq I_{k-2}\} = \frac{1}{2}\left(P\left\{ I_{k-1} > \frac{1-(h_2 + w_2^{(k)})}{|h_1 - w_1^{(k)}|} \right\} \right.$$

$$\left. + P\left\{ I_{k-1} > \frac{1+(h_2 + w_2^{(k)})}{|h_1 - w_1^{(k)}|} \right\} \right)$$

$$\triangleq f_3(h_1, h_2, w_1^{(k)}, w_2^{(k)})$$

$$P\{\hat{A}_k \neq I_k \mid \hat{A}_{k-1} = I_{k-1}, \hat{A}_{k-2} = I_{k-2}\} = \frac{1}{2}\left(P\left\{ I_{k-1} > \frac{1-(h_2 - w_2^{(k)})}{|h_1 - w_1^{(k)}|} \right\} \right.$$

$$\left. + P\left\{ I_{k-1} > \frac{1+(h_2 - w_2^{(k)})}{|h_1 - w_1^{(k)}|} \right\} \right)$$

$$\triangleq f_4(h_1, h_2, w_1^{(k)}, w_2^{(k)})$$

Substituting the above in equation (2.22):

$$\begin{aligned} q_k &= q_{k-1}q_{k-2}f_1(h_1, h_2, w_1^{(k)}, w_2^{(k)}) + q_{k-1}p_{k-2}f_2(h_1, h_2, w_1^{(k)}, w_2^{(k)}) \\ &\quad + p_{k-1}q_{k-2}f_3(h_1, h_2, w_1^{(k)}, w_2^{(k)}) + p_{k-1}p_{k-2}f_4(h_1, h_2, w_1^{(k)}, w_2^{(k)}), \end{aligned} \quad (2.26)$$

where p_k is the probability of a correct decision. Equation (2.26) is a second-order difference equation which depends on the channel parameters h_1 and h_2 and on the current equalizer's weights $w_1^{(k)}$ and $w_2^{(k)}$. For the more general order N, the channel equation (2.26) will take the general form

$$q_k = f(q_{k-1}, \cdots, q_{k-N}, h_1, \cdots, h_N, w_1^{(k)}, \cdots, w_N^{(k)}). \quad (2.27)$$

The instantaneous probability of error may be computed recursively using equation (2.27), weights update equation (2.19), and the appropriate initial conditions for the probability of error. Equation (2.27) is highly nonlinear; therefore, only low-order channels are numerically tractable for showing the convergence of q_k to zero.

2.2.6 Illustrative Examples and Simulation

In studying the dynamic behavior of the blind decision feedback equalizer and examining the convergence of q_k in equation (2.26), we will use the mean of the weights, *i.e.*, the expected values of $w_1^{(k)}$ and $w_2^{(k)}$. Therefore the controlling algorithm:

$$\overline{w_i^{(k+1)}} = \overline{w_i^{(k)}} + \mu \overline{A_k A_{k-i}} \quad \text{for } i = 1, 2. \tag{2.28}$$

By substituting from equations (2.13) and (2.16) we get

$$\begin{aligned}
\overline{w_2^{(k+1)}} &= \overline{w_2^{(k)}} + \mu(h_2 - \overline{w_2^{(k)}}) \\
\overline{w_1^{(k+1)}} &= \overline{w_1^{(k)}} + \mu((h_1 - \overline{w_1^{(k)}}) + h_2(h_1 - (1 - 2q_{k-2})\overline{w_1^{(k)}}) \\
&\quad + \overline{w_2^{(k)}}(\overline{w_1^{(k)}} - (1 - 2q_{k-2})h_1))
\end{aligned}$$

Solving these two equations together with equation (2.26) recursively will give the transient behavior of the weights $w_i^{(k)}$ in the mean and the error probability q_k.

Example 1 Consider a minimum phase channel with transfer function

$$H(z^{-1}) = 1 + 0.8z^{-1} - 0.6z^{-2}.$$

Figure 2.4 depicts the probability of error q_k and the expected value of the weights $w_i^{(k)}, i = 1, 2$ as a function of k. The initial probability of error used was $q_{-1} = q_{-2} = \frac{1}{2}$, $q_0 = \frac{1}{4}$. It is clear from this figure that the weights converge to the channel parameters ($w_1 = 0.8$ and $w_2 = -0.6$). Notice that the weights converge to the right value after a certain number of iterations, and, that the error probability reaches zero after approximately 50 iterations. The error becomes zero and stays at that value before the weights converge to their final values. This is due to the decision making and the absence of noise.

Example 2 For the non-minimum phase channel given by

$$H(z^{-1}) = 1 + 0.5z^{-1} - 1.44z^{-2},$$

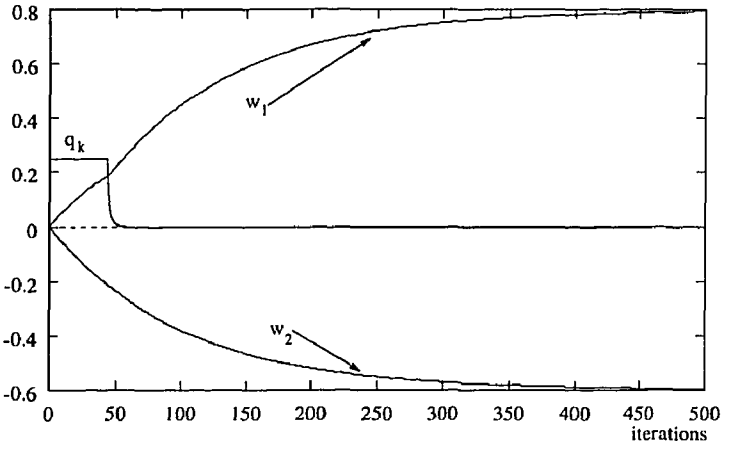

Figure 2.4 Probability of Error and Learning Curve for Example 1.

Figure 2.5 shows the probability of error q_k and the expected value of the weights $w_i^{(k)}$, $i = 1, 2$ as a function of k. Note that the weight as well as the probability of error converges slower than in Example 1 to the correct value of the channel parameters (in this case $w_1 = 0.5$ and $w_2 = -1.44$).

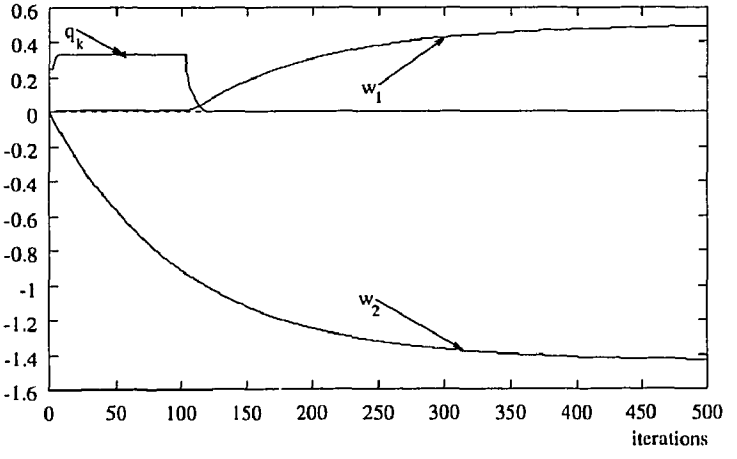

Figure 2.5 Probability of Error and Learning Curve for Example 2.

The channels considered in the above examples were also used in a computer simulation. Here we implement the stochastic control of equation (2.19) directly

to extract the value of $w_i^{(k)}$. The results are shown in Figures 2.6 and 2.7 for the minimum and non-minimum phase channels of Examples 1 and 2, respectively. At each iteration equation (2.26) was used to calculate the probability of error q_k which is also shown in Figures 2.6 and 2.7. Notice that these figures show the Monte Carlo averages of 200 experiments each. The adaptation constant used was $\mu = 0.01$ and the weights were first initialized to zero.

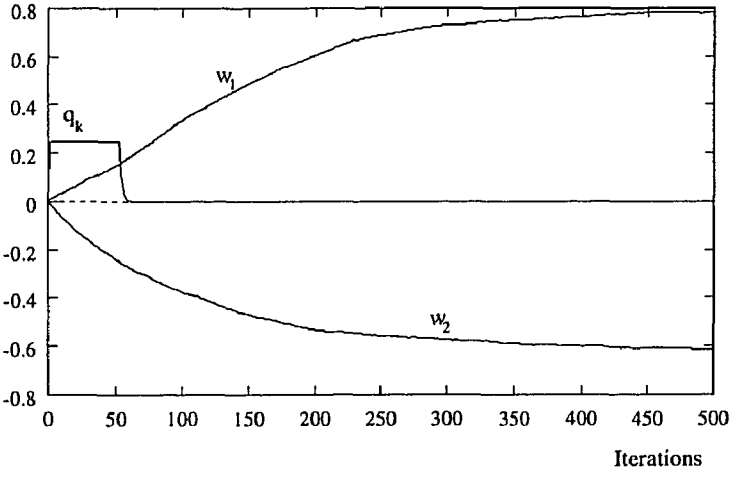

Figure 2.6 Probability of Error and Learning Curve using Simulation.

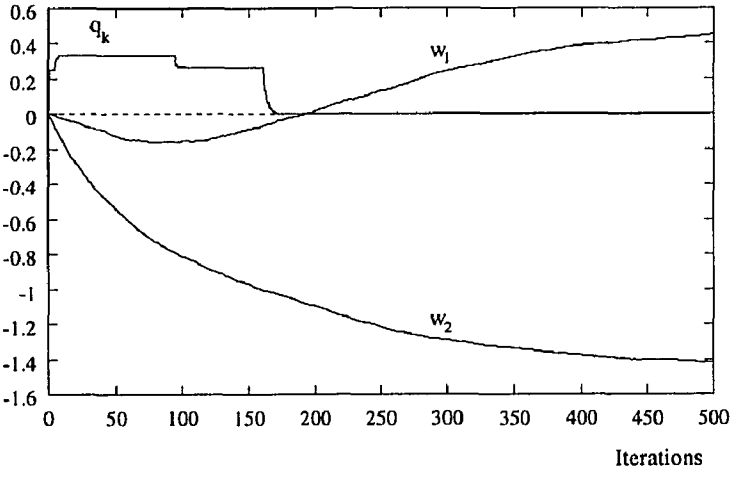

Figure 2.7 Probability of Error and Learning Curve using Simulation.

Next, by varying the initial settings of the equalizer weights, we show that with the channel of Example 2 the algorithm always converges to the correct point (0.5, -1.44) regardless of the initial condition. Figure 2.8 portrays the trajectories for different equalizer initializations. It is clearly showing that the decorrelation algorithm is globally convergent for the channel under consideration.

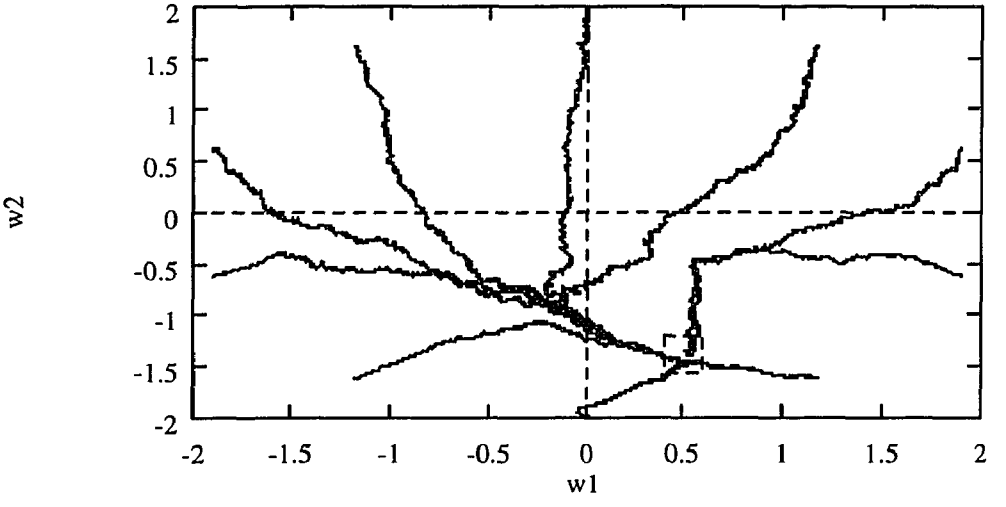

Figure 2.8 Admissibility of the BDFE

Analytical derivation of the probability of error for a higher-order channel using the techniques in the previous section is extremely complicated. Instead we present in Figure 2.9 a simulation performed in a manner similar to those in Figures 2.6 and 2.7 for equalizing the channel whose transfer function is given by

$$H(z^{-1}) = 1 + 0.8z^{-1} + 0.4z^{-2} - 0.6z^{-3} + 0.2z^{-4}.$$

The four weights $w_i, i = 1, 2, 3, 4$ converge to the correct channel parameters.

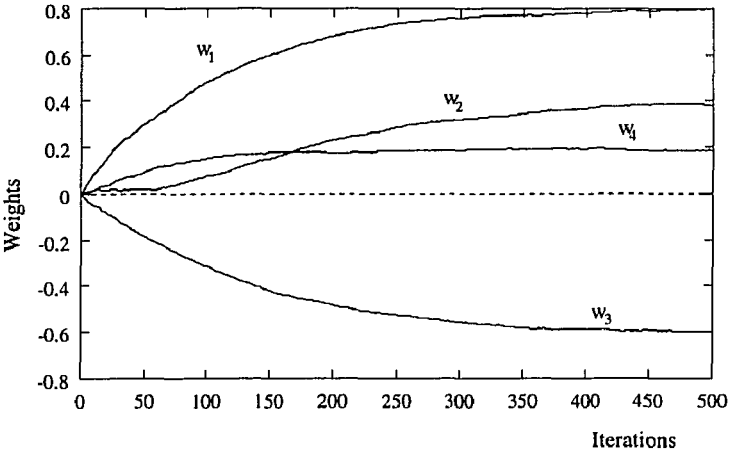

Figure 2.9 Learning Curve of the BDFE

2.3 Weighted Decorrelation Algorithm

A drawback of using a blind equalizer is the speed of convergence. It can take several hundred to several thousand symbols for the blind equalizer to converge. The speed of convergence for the conventional LMS equalizer was improved by using a weighted sum of the past squared errors [6]. The resulting algorithm, known as the Recursive Least Squares (RLS), improves the speed of convergence substantially. The penalty is an increase in computational complexity.

The decorrelation algorithm described in this chapter uses a simple error function, which makes it easy to extend to an RLS-like algorithm. In this section we show how one can improve the convergence of the algorithm by considering a weighted sum of past correlations.

Consider the Blind DFE. The input to the slicer is given by

$$A_k = X_k - \sum_{i=1}^{n} w_i^{(k)} \hat{A}_{k-i}, \tag{2.29}$$

where X_k is the input to the equalizer at time k and w_i's are the equalizer's weights. Equation (2.29) can also be written as

$$A_k = X_k - \hat{\mathbf{A}}_j' \mathbf{W}_k, \tag{2.30}$$

where $\hat{\mathbf{A}}'_j = [\hat{A}_{j-1}, \hat{A}_{j-2}, \cdots, \hat{A}_{j-n}]$ and $\mathbf{W}'_k = [w_1^{(k)}, w_2^{(k)}, \cdots, w_n^{(k)}]$.

To improve the convergence speed of the classical LMS equalizer, Godard [6] suggested the use of the Kalman algorithm for equalization. The Kalman algorithm, or Recursive Least Squares algorithm, minimizes the weighted sum squared error. Using the same approach, we can use the weighted sum of past correlations. We proceed as follows. We force

$$\sum_{j=0}^{k} \lambda^{k-j} A_j \mathbf{A}_j$$

to zero, where $\mathbf{A}'_j = [A_{j-1}, A_{j-2}, \cdots, A_{j-n}]$. Substituting for A_k from equation (2.30) and setting the weighted correlation time average to zero, we get

$$\sum_{j=0}^{k} \lambda^{k-j} \mathbf{A}_j \left(X_j - \hat{\mathbf{A}}'_j \mathbf{W}_k \right) = 0.$$

The above equation leads to

$$\mathbf{W}_k = \mathbf{R}^{-1}{}_k \mathbf{D}_k, \tag{2.31}$$

where

$$\mathbf{R}_k \triangleq \sum_{j=0}^{k} \lambda^{k-j} \mathbf{A}_j \hat{\mathbf{A}}'_j$$

and

$$\mathbf{D}_k \triangleq \sum_{j=0}^{k} \lambda^{k-j} X_j \mathbf{A}_j.$$

2.3.1 The Recursive Matrix Inversion

Equation (2.31) involves the inversion of an $n \times n$ matrix, $\mathbf{R}_k$, and the Kalman formulation involves a recursion formula for the evaluation of the inverse matrix. A similar one can be used here.

It is important to note that matrix $\mathbf{R}_k$ can be obtained recursively as

$$\mathbf{R}_k = \lambda \mathbf{R}_{k-1} + \mathbf{A}_k \hat{\mathbf{A}}'_k. \tag{2.32}$$

It is known that for any $\mathbf{A}$ nonsingular matrix, and $\mathbf{u}$ and $\mathbf{v}$ the following is true:

$$(\mathbf{A}+\mathbf{uv}')^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{uv}'\mathbf{A}^{-1}}{1+\mathbf{v}'\mathbf{A}^{-1}\mathbf{u}}. \tag{2.33}$$

Therefore, using equation (2.33) in equation (2.32), we get a recursive formula for $\mathbf{R^{-1}}_k$:

$$\mathbf{R^{-1}}_k = \frac{1}{\lambda}\left(\mathbf{R^{-1}}_{k-1} - \frac{\mathbf{R^{-1}}_{k-1}\mathbf{A}_k\hat{\mathbf{A}}_k'\mathbf{R^{-1}}_{k-1}}{\lambda+\hat{\mathbf{A}}_k'\mathbf{R^{-1}}_{k-1}\mathbf{A}_k}\right). \tag{2.34}$$

Next define

$$\mathbf{P}_k \triangleq \mathbf{R^{-1}}_k, \tag{2.35}$$

and further define the Kalman vector gain as

$$\mathbf{k}_k = \frac{1}{\lambda+\mu_k}\mathbf{P}_{k-1}\mathbf{A}_k, \tag{2.36}$$

where the scalar μ_k is given by

$$\mu_k = \hat{\mathbf{A}}_k'\mathbf{R^{-1}}_{k-1}\mathbf{A}_k.$$

Using the above definitions, one can write equation (2.34) as

$$\mathbf{P}_k = \frac{1}{\lambda}\left(\mathbf{P}_{k-1} - \mathbf{k}_k\hat{\mathbf{A}}_k'\mathbf{P}_{k-1}\right). \tag{2.37}$$

The vector $\mathbf{D}_k$ can also be obtained recursively as

$$\mathbf{D}_k = \lambda\mathbf{D}_{k-1} + X_k\mathbf{A}_k. \tag{2.38}$$

Using equations (2.31) and (2.35), we can write

$$\mathbf{W}_k = \mathbf{P}_k\mathbf{D}_k.$$

Therefore, using the recursive formulae for $\mathbf{P}_k$ and $\mathbf{D}_k$ from equations (2.37) and (2.38) respectively, we get

$$\mathbf{W}_k = \frac{1}{\lambda}\left(\mathbf{P}_{k-1} - \mathbf{k}_k\hat{\mathbf{A}}_k'\mathbf{P}_{k-1}\right)(\lambda\mathbf{D}_{k-1} + X_k\mathbf{A}_k)$$

$$
\begin{aligned}
&= \mathbf{P}_{k-1}\mathbf{D}_{k-1} + \frac{1}{\lambda}X_k\mathbf{P}_{k-1}\mathbf{A}_k \\
&\quad -\mathbf{k}_k\hat{\mathbf{A}}_k'\mathbf{P}_{k-1}\mathbf{D}_{k-1} - \frac{1}{\lambda}X_k\mathbf{k}_k\hat{\mathbf{A}}_k'\mathbf{P}_{k-1}\mathbf{A}_k \\
&= \mathbf{W}_{k-1} + \frac{1}{\lambda}X_k\left(\lambda + \mu_k\right)\mathbf{k}_k \\
&\quad -\mathbf{k}_k\hat{\mathbf{A}}_k'\mathbf{W}_{k-1} - \frac{1}{\lambda}X_k\mu_k\mathbf{k}_k \\
&= \mathbf{W}_{k-1} + \mathbf{k}_k\left(X_k - \hat{\mathbf{A}}_k'\mathbf{W}_{k-1}\right) \\
&= \mathbf{W}_{k-1} + z_k\mathbf{k}_k, \qquad (2.39)
\end{aligned}
$$

where z_k is given by

$$z_k = \left(X_k - \hat{\mathbf{A}}_k'\mathbf{W}_{k-1}\right).$$

The order that constitutes the weighted decorrelation algorithm is summarized below:

$$\mathbf{W}_k = \mathbf{W}_{k-1} + z_k\mathbf{k}_k$$

where

$$z_k = \left(X_k - \hat{\mathbf{A}}_k'\mathbf{W}_{k-1}\right)$$

The vector K_k is evaluated by the recursions

$$\mathbf{k}_k = \frac{1}{\lambda + \mu_k}\mathbf{P}_{k-1}\mathbf{A}_k$$

$$\mathbf{P}_k = \frac{1}{\lambda}\left(\mathbf{P}_{k-1} - \mathbf{k}_k\hat{\mathbf{A}}_k'\mathbf{P}_{k-1}\right).$$

where

$$\mu_k = \hat{\mathbf{A}}_k'\mathbf{P}_{k-1}\mathbf{A}_k$$

2.3.2 Simulation Results

A channel whose transfer function given by

$$H(z^{-1}) = 1 + 0.5z^{-1} - 1.44z^{-2}$$

was used to demonstrate the improvement in the convergence rate when using the weighted correlation approach. Figure 2.10 depicts the Monte-Carlo average of 100 experiments of the decorrelation DFE and the weighted decorrelation DFE.

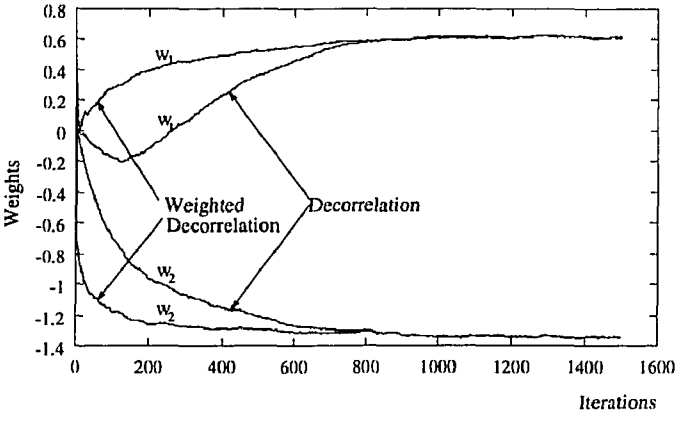

Figure 2.10 Learning Curves of the Decorrelating DFE and Weighted Decorrelating DFE

Figure 2.11 depicts the averaged squared error of the decorrelating DFE and the weighted decorrelating DFE. The increased speed of the latter is clearly shown.

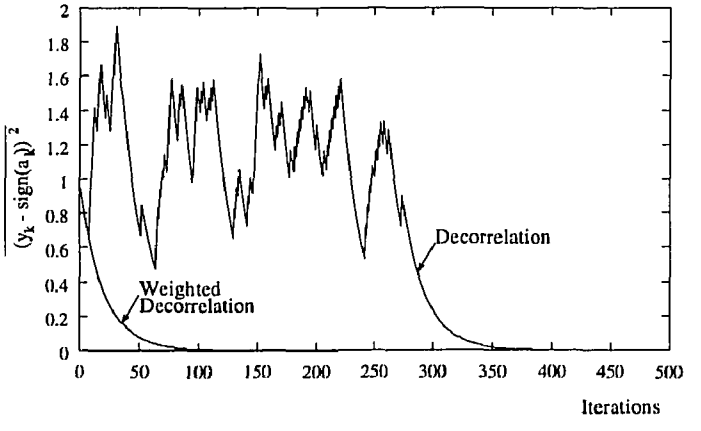

Figure 2.11 Residual ISI of the Decorrelating DFE and Weighted Decorrelating DFE

Other forms of fast blind equalizers based on decorrelation can also be derived. One can exploit the shifting properties of the correlation matrix $\mathbf{R}$ in the same way as [9] to derive fast versions. In this way, one can reduce the computational complexity while maintaining the speed of convergence.

CHAPTER 3

ERROR ANALYSIS OF THE BLIND DFE

In Chapter 2, we introduced a blind decision feedback equalizer based on the decorrelation of data samples at the input of the slicer. Assuming no additive noise, we showed that decorrelation of these samples is necessary and sufficient for obtaining zero ISI at the steady state. We also examined the dynamic behavior of the decorrelation algorithm and showed convergence to the steady state with zero ISI. However, due to the analytical complexity of the problem, we restricted our dynamic study only to the third-order moving average channel.

Following the technique developed in [15], we extend in this chapter our dynamic study of the blind decorrelation equalizer to include channels of any order. The steady state probability of error for additive white Gaussian noise (AWGN) is considered. We derive upper and lower bounds on the probability of error. The results of this chapter also appear in [48].

After stating the problem and giving the error model section 3.1, we derive an upper bound expression for the probability of error of the equalizer during the transient period in the absence of noise (The source of error during the transient period is the imperfect equalization). Section 3.3 presents lower and upper bounds on the probability of error in the steady state in the presence of AWGN. Numerical and simulation results are given in section 3.4. The chapter is concluded in section 3.5.

3.1 Problem Statement and Error Modeling

Consider the decision feedback equalizer shown in Figure 3.1. Assume the sampled impulse response at the input be given by

$$X_k = I_k + \sum_{i=1}^{N} h_i I_{k-i} + n_k, \tag{3.1}$$

where I_k is the data to be detected and n_k is an AWGN with a zero mean and variance of σ^2.

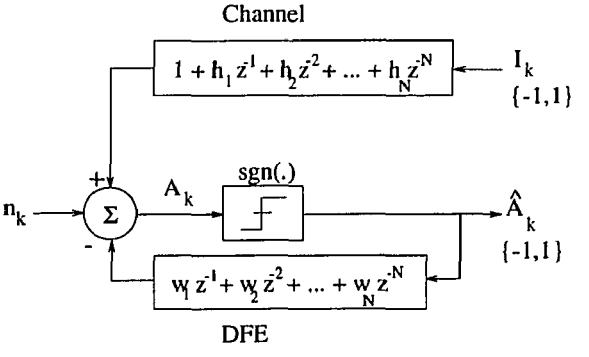

Figure 3.1 Decision Feedback Equalizer and Channel Model

At the input of the slicer, the sampled signal is given by

$$\begin{aligned} A_k &= X_k - \sum_{i=1}^{N} w_i \hat{A}_{k-i} \\ &= I_k + \sum_{i=1}^{N} h_i I_{k-i} - \sum_{i=1}^{N} w_i \hat{A}_{k-i} + n_k. \end{aligned} \tag{3.2}$$

Since we are interested in the probability of error due to noise and error propagation we will assume perfect equalization, *i.e.*, the weights w_i have converged to the channel parameters h_i. Extension to imperfect equalization can be addressed in a similar manner as in [15]. One may then write equation (3.2) as

$$A_k = I_k + \sum_{i=1}^{N} h_i E_{k-i} + n_k, \tag{3.3}$$

where $E_k = I_k - \hat{A}_k$. For binary transmission E_k takes the values $\{-2, 0, 2\}$.

Duttweiler, *et al.*, [15] considered the following reduced finite state machine for the error sequences $\{E_{k-i}, 1 \leq i \leq M\}$, where M, the number of states, is an arbitrary positive integer:

$$\begin{aligned}
\phi_0 &\triangleq \{E_{k-1}, E_{k-2}, \cdots | E_{k-1} \neq 0\} \\
\phi_m &\triangleq \{E_{k-1}, E_{k-2}, \cdots | E_{k-j} = 0, 1 \leq j \leq m, E_{k-m-1} \neq 0\}, \qquad 1 \leq m \leq M-1 \\
\phi_M &\triangleq \{E_{k-1}, E_{k-2}, \cdots | E_{k-j} = 0, 1 \leq j \leq M\}.
\end{aligned}$$

ϕ_m consists of the error sequences that start with m zeros (*i.e.*, all decisions made from $k-1$ to $k-m$ are correct) followed by a non-zero error (*i.e.*, the $k-m-1$ decision was erroneous). In other words, when the equalizer is in state ϕ_m, the first m delay elements (the first delay element is the one closest to the slicer) in the feedback filter contain correct decisions, and the $(m+1)$th element has an erroneous decision. For our analysis we let M equal N, the order of the channel.

The states are given in the table below.

Table 3.1 State Assignment

State	Error Sequence	
ϕ_0	E X X $\cdots$	
ϕ_1	0 E X X $\cdots$	
ϕ_2	0 0 E X X $\cdots$	
$\vdots$		
ϕ_{N-1}	0 0 $\cdots$ 0 E X X	(the first $N-1$ elements are 0's)
ϕ_N	0 0 $\cdots$ 0 0 X X	(the first N elements are 0's)

A discrete random process S_k is defined which takes an integer value from $\{0, 1, \cdots, N\}$. Specifically when the equalizer is in state ϕ_i then $S_k = i$. In other words, S_k equals the number of consecutive correct decisions starting from the first delay element. From Table 3.1 if at time k the equalizer is in state ϕ_i, $i = 1, \cdots, N-1$, *i.e.* $S_k = i$ then S_{k+1} could either be $i+1$ if the new decision is correct, or 0 if it is erroneous. Being in state ϕ_0 then $S_{k+1} = 1$ if the next decision is correct, or

$S_{k+1} = 0$ if the decision is erroneous. On the other hand, being in state ϕ_N, $S_k = N$ then $S_{k+1} = N$ if the next decision is correct or $S_{k+1} = 0$ if it is wrong. The state transition diagram is shown in Figure 3.2.

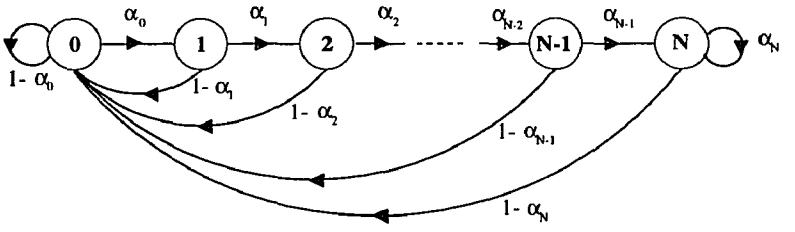

Figure 3.2 State Transition Diagram

In [15] it is assumed that the random sequence $\{S_k\}$ has reached a state of second-order stationarity, such that $P\{S_k = m\}$ and $P\{S_k = m|S_{k-1} = m - j\}$ are independent of the time k for all m and j. From the State Transition Diagram it clearly follows that

$$P\{S_k = m|S_{k-1} = m - j\} = 0, \quad \text{for } j \neq 1, \ m \neq 0, \tag{3.4}$$

and therefore,

$$\begin{aligned} P\{S_k = m\} &= \sum_{l=0}^{N} P\{S_k = m|S_{k-1} = l\}P\{S_{k-1} = l\} \\ &= P\{S_k = m|S_{k-1} = m-1\}P\{S_{k-1} = m-1\} \\ &\triangleq p_m \qquad m = 1, \cdots, N-1. \end{aligned}$$

It is also clear from the definition of S_k and the state diagram that

$$\begin{aligned} P\{S_k = m|S_{k-1} = m-1\} &= P\{E_k = 0|S_{k-1} = m-1\} \\ &\triangleq \alpha_{m-1} \qquad m = 1, \cdots, N-1, \end{aligned}$$

while

$$\begin{aligned} P\{S_k = 0|S_{k-1} = m-1\} &= P\{E_k \neq 0|S_{k-1} = m-1\} \\ &= 1 - \alpha_{m-1}. \end{aligned}$$

It can be seen that α_m is the probability of the correct decision when the equalizer is in state ϕ_m. The probability $P\{S_k = 0\}$ and $P\{S_k = N\}$ can be obtained from the above equations and the state diagram. In fact, one can show that

$$p_0 = \sum_{l=0}^{N} (1 - \alpha_l)\, p_l \tag{3.5}$$

$$p_m = \alpha_{m-1} p_{m-1}, \qquad 1 \leq m \leq N-1 \tag{3.6}$$

$$p_N = \frac{\alpha_{N-1}}{1 - \alpha_N} p_{N-1}. \tag{3.7}$$

Clearly, at any time k the probability of error is given by

$$\begin{aligned} q_k &= P\{E_k \neq 0\} \\ &= P\{S_k = 0\} \\ &= p_0. \end{aligned} \tag{3.8}$$

Now, from equations (3.6) and (3.7) we can write

$$\begin{aligned} p_m &= p_0 \prod_{i=0}^{m-1} \alpha_i, \qquad 1 \leq m \leq N-1 \\ p_N &= p_0 (1 - \alpha_N)^{-1} \prod_{i=0}^{N-1} \alpha_i. \end{aligned}$$

However,

$$\begin{aligned} \sum_{i=0}^{N} p_i &= p_0 + \sum_{i=1}^{N-1} p_i + p_N \\ &= p_0 \left(1 + \sum_{i=0}^{N-2} \prod_{m=0}^{i} \alpha_m + (1 - \alpha_N)^{-1} \prod_{m=0}^{N-1} \alpha_m \right) \\ &= 1. \end{aligned}$$

Therefore,

$$q = R_N^{-1},$$

where

$$R_N = 1 + \sum_{i=0}^{N-2} \prod_{m=0}^{i} \alpha_m + (1 - \alpha_N)^{-1} \prod_{m=0}^{N-1} \alpha_m. \tag{3.9}$$

In order to calculate the probability of error, one has to know all the state transition probabilities, α_i. This is not feasible for large N. Instead, Duttweiler, *et al.*, used a lower bound on the transition probability to derive an upper bound on the probability of error.

By deriving expressions for the lower and upper bounds on the transition probabilities, we obtain respectively an upper and lower bound on the probability of error of the blind decision feedback equalizer.

3.2 Transient Behavior of the Probability of Error

In Chapter 2 we considered the probability of error of the blind decision feedback equalizer in the transient state and showed that it converges to zero when noise is absent. In the noiseless case, the error is caused only by the residual ISI. Due to the analytical complexity of the problem, we were able to establish our results only for low-order channels. Using the error model of the previous section, we can extend the result to higher-order channels.

In this section, we first derive a lower bound on the state transition probabilities. Using this bound in equation (3.9), we can get an upper bound on the probability of error. It should be mentioned that in the transient period these bounds are functions of the time index k.

In the absence of noise, equation (3.2) becomes

$$A_k = I_k + \sum_{i=1}^{N} h_i I_{k-i} - \sum_{i=1}^{N} w_i^{(k)} \hat{A}_{k-i}. \tag{3.10}$$

3.2.1 Transition Probability α_N

Being at state N implies that $\hat{A}_{k-i} = I_{k-i}$ for $i = 1, \cdots, N$ so that equation (3.10) can be written as

$$A_k = I_k + \sum_{i=1}^{N} \left(h_i - w_i^{(k)}\right) I_{k-i}. \tag{3.11}$$

The state transition probability α_N (which is the probability of correct decision) is given by

$$\alpha_N = \frac{1}{2}P\{\hat{A}_k = 1|I_k = 1\} + \frac{1}{2}P\{\hat{A}_k = -1|I_k = -1\}. \tag{3.12}$$

It can be shown that $P\{\hat{A}_k = 1|I_k = 1\} = P\{\hat{A}_k = -1|I_k = -1\}$; therefore,

$$\begin{aligned}
\alpha_N &= P\{\hat{A}_k = 1|I_k = 1\} \\
&= P\{1 + \sum_{i=1}^{N} \left(h_i - w_i^{(k)}\right) I_{k-i} > 0\} \\
&= P\{\sum_{i=1}^{N} \left(h_i - w_i^{(k)}\right) I_{k-i} > -1\} \\
&= P\{\left(h_1 - w_1^{(k)}\right) I_{k-1} + Y > -1\},
\end{aligned} \tag{3.13}$$

where the random variable Y is given by

$$Y = \sum_{i=2}^{N} \left(h_i - w_i^{(k)}\right) I_{k-i}.$$

By conditioning on Y, we can obtain a lower and an upper bound on α_N. However, since we are interested in proving that the probability of error approaches zero, we will consider only the lower bound on α_N. In Appendix B we derive this bound:

$$\alpha_N \geq P\left\{I_{k-1} > -\frac{1 - \sum_{i=2}^{N} |h_i - w_i^{(k)}|}{|h_1 - w_1^{(k)}|}\right\}. \tag{3.14}$$

3.2.2 Transition Probability α_m

Being at state m implies that the previous m decisions were correct ($\hat{A}_{k-i} = I_{k-i}$ for $i = 1, \cdots, m$), the $(m+1)$th decision was incorrect ($\hat{A}_{k-m-1} = -I_{k-m-1}$) and one cannot specify the rest of the decisions. Therefore, equation (3.10) becomes

$$\begin{aligned}
A_k = \quad & I_k + \sum_{i=1}^{m} \left(h_i - w_i^{(k)}\right) I_{k-i} + \left(h_{m+1} + w_{m+1}^{(k)}\right) I_{k-m-1} + \sum_{i=m+1}^{N} h_i I_{k-i} \\
& - \sum_{i=m+1}^{N} w_i^{(k)} \hat{A}_{k-i}.
\end{aligned}$$

Following a similar technique as above, we derive a lower bound on the conditional probability $P\{\hat{A}_k = 1|I_k = 1\}$. In fact,

$$\alpha_m = P\left\{\hat{A}_k = 1|I_k = 1\right\}$$

$$
\begin{aligned}
&= P\left\{1 + \left(h_1 - w_1^{(k)}\right) I_{k-1} + \sum_{i=2}^{m} \left(h_i - w_i^{(k)}\right) I_{k-i} \right. \\
&\quad \left. + \left(h_{m+1} + w_{m+1}^{(k)}\right) I_{k-m-1} + \sum_{i=m+1}^{N} h_i I_{k-i} - \sum_{i=m+1}^{N} w_i^{(k)} \hat{A}_{k-i} > 0 \right\} \\
&= P\left\{1 + \left(h_1 - w_1^{(k)}\right) I_{k-1} + Y_1 + Y_2 + Y_3 - Y_4 > 0\right\}, \qquad (3.15)
\end{aligned}
$$

where

$$
\begin{aligned}
Y_1 &= \sum_{i=2}^{m} \left(h_i - w_i^{(k)}\right) I_{k-i} \\
Y_2 &= \left(h_{m+1} + w_{m+1}^{(k)}\right) I_{k-m-1} \\
Y_3 &= \sum_{i=m+1}^{N} h_i I_{k-i} \\
Y_4 &= \sum_{i=m+1}^{N} w_i^{(k)} \hat{A}_{k-i}.
\end{aligned}
$$

In Appendix B we derive the lower bound on α_m, which is given by

$$
\alpha_m \geq
P\left\{ I_{k-1} > -\frac{1 - \sum_{i=2}^{m} |h_i - w_i^{(k)}| - |h_{m+1} + w_{m+1}^{(k)}| - \sum_{i=m+2}^{N} |h_i| - \sum_{i=m+2}^{N} |w_i^{(k)}|}{|h_1 - w_1^{(k)}|} \right\}. \qquad (3.16)
$$

3.2.3 Transition Probability α_0

At state zero, the past decision was erroneous ($\hat{A}_{k-1} = -I_{k-1}$) and nothing can be said about the other decisions. Therefore, from equation (3.10) we have

$$
A_k = I_k + \left(h_1 + w_1^{(k)}\right) + \sum_{i=2}^{N} h_i I_{k-i} - \sum_{i=2}^{N} w_i^{(k)} \hat{A}_{k-i}
$$

$$
\begin{aligned}
\alpha_0 &= P\left\{\hat{A}_k \neq 1 | I_k = 1\right\}. \\
&= P\left\{1 + \left(h_1 + w_1^{(k)}\right) + \sum_{i=2}^{N} h_i I_{k-i} - \sum_{i=2}^{N} w_i^{(k)} \hat{A}_{k-i} < 0\right\}.
\end{aligned}
$$

Following a similar derivation, one can show that

$$
\alpha_0 \geq P\left\{ I_{k-1} > -\frac{1 - \sum_{i=2}^{N} |h_i| - \sum_{i=2}^{N} |w_i^{(k)}|}{|h_1 + w_1^{(k)}|} \right\}. \qquad (3.17)
$$

3.2.4 Summary of Lower Bounds on Transition Probabilities

In summary, we have

$$\alpha_0 \geq P\left\{ I_{k-1} > -\frac{1-\sum_{i=2}^{N}|h_i|-\sum_{i=2}^{N}|w_i^{(k)}|}{|h_1+w_1^{(k)}|} \right\}$$

$$\alpha_m \geq P\left\{ I_{k-1} > -\frac{1-\sum_{i=2}^{m}|h_i-w_i^{(k)}|-|h_{m+1}+w_{m+1}^{(k)}|-\sum_{i=m+2}^{N}|h_i|-\sum_{i=m+2}^{N}|w_i^{(k)}|}{|h_1-w_1^{(k)}|} \right\},$$
$$\text{for } 0 \leq m \leq N-1$$

$$\alpha_N \geq P\left\{ I_{k-1} > -\frac{1-\sum_{i=2}^{N}|h_i-w_i^{(k)}|}{|h_1-w_1^{(k)}|} \right\}.$$

Given $w_i^{(k)}$, one can determine an upper bound on the instantaneous probability of error q_k.

To evaluate the transient response of the equalizer, we can iteratively solve for the instantaneous probability of error together with the equalizer weights from the update equation given in [46] and summarized below.

$$\begin{aligned}
w_N^{(k+1)} &= w_N^{(k)} + \mu\left(h_N - w_N^{(k)}\right) \\
w_{N-1}^{(k+1)} &= w_{N-1}^{(k)} + \mu\left(h_N(h_1 - w_1^{(k)}(1-2q_{k-N})) + w_N(w_1^{(k)} - h_1(1-2q_{k-N}))\right. \\
&\quad \left. + h_{N-1} - w_{N-1}^{(k)}\right) \\
&\vdots \\
w_1^{(k+1)} &= w_1^{(k)} + \mu\left(\sum_{i=2}^{N} h_i(h_{i-1} - w_{i-1}^{(k)}(1-2q_{k-i})) + \sum_{i=2}^{N} w_i^{(k)}(w_{i-1}^{(k)} - h_{i-1}(1-2q_{k-i}))\right. \\
&\quad \left. + h_1 - w_1^{(k)}\right), \qquad (3.18)
\end{aligned}$$

where $q_k = P\{\hat{A}_k = -I_k\} = R_N^{-1}$, obtained from equation (3.9).

3.3 Steady State Probability of Error

In this section we consider the probability of error at the steady state. In this case we assume that the input to the equalizer is corrupted by AWGN. We consider

the state transition diagram in Figure 3.2. One can find exact expressions for the transition probabilities α_{N-1} and α_N. For the other transition probabilities α_m for $1 \leq m < N-1$; however, we will derive upper and lower bounds. Using these we can determine lower and upper bounds for the steady state probability of error as a function of the received bit energy to noise power ratio (E_b/N_0).

3.3.1 Transition Probability α_N

Rewriting equation (3.3),

$$A_k = I_k + \sum_{i=1}^{N} h_i E_{k-i} + n_k. \tag{3.19}$$

Since the equalizer was in state N, the last N decisions were correct ($E_{k-i} = 0$, $i = 1, \cdots, N$) and equation (3.19) becomes

$$A_k = I_k + n_k.$$

Now the state transition probability, α_N, is

$$\alpha_N = \frac{1}{2} P\left\{\hat{A}_k = 1 | I_k = 1\right\} + \frac{1}{2} P\left\{\hat{A}_k = -1 | I_k = -1\right\},$$

where

$$\begin{aligned} P\left\{\hat{A}_k = 1 | I_k = 1\right\} &= P\{1 + n_k > 0\} \\ &= P\{n_k > -1\}, \end{aligned}$$

and

$$\begin{aligned} P\left\{\hat{A}_k = -1 | I_k = -1\right\} &= P\{-1 + n_k < 0\} \\ &= P\{n_k < 1\}. \end{aligned}$$

Therefore,

$$\begin{aligned} \alpha_N &= \frac{1}{2}\left(P\{n_k > -1\} + P\{n_k < 1\}\right) \\ &= P\{n_k > -1\} \\ &= Q\left(-\frac{1}{\sigma}\right), \end{aligned} \tag{3.20}$$

where

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{y^2}{2}} dy.$$

3.3.2 Transition Probability α_{N-1}

Being in state $N-1$ means that the past $N-1$ were correct and the Nth decision was erroneous. Therefore, equation (3.19) becomes

$$A_k = I_k + h_N E_{k-N} + n_k. \tag{3.21}$$

Now $\hat{A}_{k-N}$ in error implies that E_{k-N} assumes the value ± 2. Following a similar procedure as the one above, we derive an expression for α_{N-1} in Appendix 2. The transition probability α_{N-1} is given by

$$\alpha_{N-1} = \frac{1}{2}\left(Q\left(\frac{-1-2h_N}{\sigma}\right) + Q\left(\frac{-1+2h_N}{\sigma}\right)\right). \tag{3.22}$$

3.3.3 Transition Probability α_m

Being in state m implies that the past m were correct and the decision on $\hat{A}_{k-m-1}$ was erroneous. Nothing can be said about $\hat{A}_{k-i}$ for $i = m+2, \cdots, N$. Thus, rewriting equation (3.3) as

$$\begin{aligned} A_k &= I_k + h_{m+1} E_{k-m-1} + \sum_{m+2}^{N} h_i E_{k-i} + n_k \\ &= I_k + h_{m+1} E_{k-m-1} + Y_{m+1} + n_k, \end{aligned} \tag{3.23}$$

where Y_m is defined as

$$Y_m \triangleq \sum_{i=m+1}^{N} h_i E_{k-i}.$$

Based on equation (3.23), in Appendix 2 we derive the lower and upper bounds on α_m:

$$\alpha_m \geq \frac{1}{2}\left(Q\left(\frac{-1-2h_{m+1}+\beta_{m+1}}{\sigma}\right) + Q\left(\frac{-1+2h_{m+1}+\beta_{m+1}}{\sigma}\right)\right) \tag{3.24}$$

$$\alpha_m \leq \frac{1}{2}\left(Q\left(\frac{-1-2h_{m+1}-\beta_{m+1}}{\sigma}\right) + Q\left(\frac{-1+2h_{m+1}-\beta_{m+1}}{\sigma}\right)\right), \tag{3.25}$$

where

$$\beta_m = 2 \sum_{i=m+1}^{N} |h_i|. \tag{3.26}$$

3.3.4 Summary of Results

To summarize, we have

$$\begin{aligned}
\alpha_m &\geq \frac{1}{2}\left(Q\left(\frac{-1-2h_{m+1}+\beta_{m+1}}{\sigma}\right)+Q\left(\frac{-1+2h_{m+1}+\beta_{m+1}}{\sigma}\right)\right), \\
&\qquad \text{for } 0 \leq m \leq N-2 \\
\alpha_m &\leq \frac{1}{2}\left(Q\left(\frac{-1-2h_{m+1}-\beta_{m+1}}{\sigma}\right)+Q\left(\frac{-1+2h_{m+1}-\beta_{m+1}}{\sigma}\right)\right), \\
&\qquad \text{for } 0 \leq m \leq N-2 \\
\alpha_{N-1} &= \frac{1}{2}\left(Q\left(\frac{-1-2h_N}{\sigma}\right)+Q\left(\frac{-1+2h_N}{\sigma}\right)\right) \\
\alpha_N &= Q\left(-\frac{1}{\sigma}\right).
\end{aligned}$$

Using these results in equation (3.9), we can obtain the lower and upper bounds on the probability of error in the steady state.

3.4 Numerical Examples and Simulation Results

As an example, we consider the channel whose transfer function is given by

$$H(z^{-1}) = 1 + 0.8z^{-1} + 0.6z^{-2} + 0.4z^{-3}.$$

In this case, the feedback section contains three delay elements, $N = 3$. We also have four states *viz*, ϕ_0, ϕ_1, ϕ_2 and ϕ_3.

The lower bounds on the transition probabilities for the instantaneous probability of error, in the absence of noise, are given by

$$\alpha_0 \geq P\left\{I_{k-1} > -\frac{1-\sum_{i=2}^{3}|h_i|-\sum_{i=2}^{3}|w_i^{(k)}|}{|h_1+w_1^{(k)}|}\right\}$$

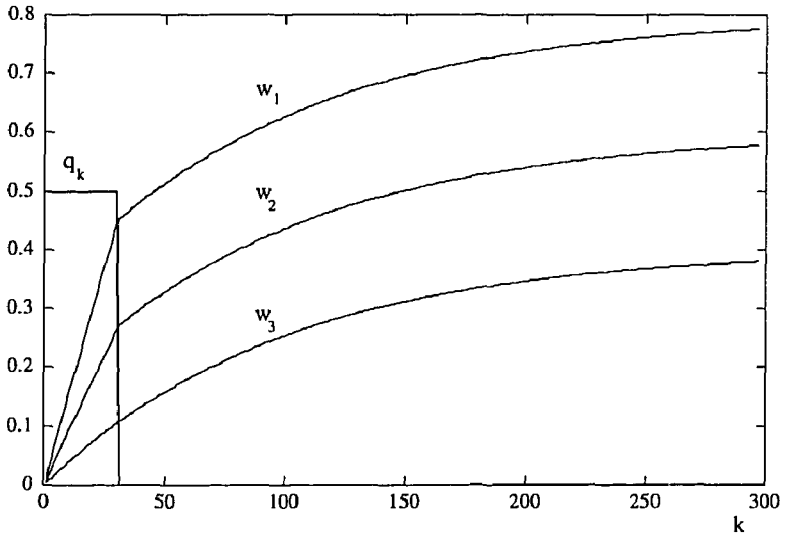

Figure 3.3 Probability of Error q_k in the Absence of Noise (Transient Period)

$$\begin{aligned}
\alpha_1 &\geq P\left\{ I_{k-1} > -\frac{1 - |h_i - w_i^{(k)}| - |h_2 + w_2^{(k)}| - |h_3| - |w_3^{(k)}|}{|h_1 - w_1^{(k)}|} \right\} \\
\alpha_2 &\geq P\left\{ I_{k-1} > -\frac{1 - \sum_{i=1}^{2} |h_i - w_i^{(k)}| - |h_3 + w_3^{(k)}|}{|h_1 - w_1^{(k)}|} \right\} \\
\alpha_3 &\geq P\left\{ I_{k-N} > -\frac{1 - \sum_{i=2}^{3} |h_i - w_i^{(k)}|}{|h_1 - w_1^{(k)}|} \right\}.
\end{aligned}$$

Substituting the above transition probabilities in equation (3.9), one can obtain an expression for the probability of error in terms of the equalizer weights. The upper bound on the probability of error can be obtained by solving the resulting expression and the weight update equation given in equation (3.18). The figure below portrays the upper bound of the probability of error against the time index k.

For the steady state probability of error, we assume perfect equalization and AWGN. The transition probabilities are given by

$$\begin{aligned}
\alpha_0 &\geq \frac{1}{2}\left(Q\left(\frac{-1 - 2h_1 + \beta_1}{\sigma} \right) + Q\left(\frac{-1 + 2h_1 + \beta_1}{\sigma} \right) \right) \\
\alpha_0 &\leq \frac{1}{2}\left(Q\left(\frac{-1 - 2h_1 - \beta_1}{\sigma} \right) + Q\left(\frac{-1 + 2h_1 - \beta_1}{\sigma} \right) \right)
\end{aligned}$$

$$
\begin{aligned}
\alpha_1 &\geq \frac{1}{2}\left(Q\left(\frac{-1-2h_2+\beta_2}{\sigma}\right)+Q\left(\frac{-1+2h_2+\beta_2}{\sigma}\right)\right)\\
\alpha_1 &\leq \frac{1}{2}\left(Q\left(\frac{-1-2h_2-\beta_2}{\sigma}\right)+Q\left(\frac{-1+2h_2-\beta_2}{\sigma}\right)\right)\\
\alpha_2 &= \frac{1}{2}\left(Q\left(\frac{-1-2h_3}{\sigma}\right)+Q\left(\frac{-1+2h_3}{\sigma}\right)\right)\\
\alpha_2 &= Q\left(-\frac{1}{\sigma}\right),
\end{aligned}
$$

where $\beta_1 = 2\left(|h_2| + |h_3|\right)$ and $\beta_2 = 2|h_3|$. Figure 3.4, below, shows the lower and the upper bound of the probability of error. Also shown is the simulation result, which did not assume perfect equalization. The upper and lower bound are also applicable to the conventional decision feedback equalizer. The "no ISI" lower bound is shown, and one can see that the lower bound derived here is tighter than the no ISI bound. We also have

$$\left(\frac{E_b}{N_0}\right)_{\text{dB}} = 10\log_{10}\frac{\sum_{i=1}^{4} h_i}{2\sigma^2}$$

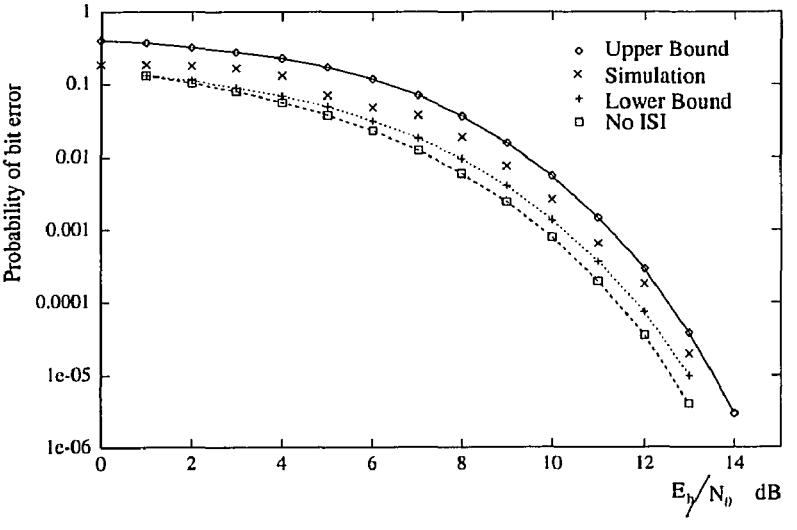

Figure 3.4 Probability of Error

3.5 Conclusion

In this chapter we used the error model proposed in [15] to obtain upper and lower bounds on the steady state probability of error for a blind decision feedback equalizer. Despite using a different technique, the upper bound obtained was the same as that derived in [15]. The lower bound is tighter than the commonly used "no ISI" lower bound.

The same error model was also used to study the behavior of the equalizer in the transient mode when the only source of error is the residual ISI. It was shown through a numerical example that the equalizer converges to the zero ISI case. It was also shown that the no noise probability of error vanishes after less than 50 iterations.

In the steady state case, we assumed perfect equalization to determine the probability of error, while in the transient case we assumed zero noise to study the convergence of the algorithm.

CHAPTER 4

ANCHORED CONSTANT MODULUS ALGORITHM

Among the first known blind equalization algorithms is the constant modulus algorithm (CMA). This algorithm is of the property restoral type. That is, it exploits the constant modulus property of the transmitted signal constellation to adapt the blind equalizer. Its cost function is non-convex, and has local minima, at some of which the equalizer is incapable of canceling ISI. The existence and stability of these minima were discussed in [31]. Due to these undesirable minima, the equalizer initialization becomes an important issue. One would need to initialize the tap weights away from the neighborhood of these minima.

Verdu, *et al.* [34], developed a technique that insures global convergence of blind equalizers. Their key observation was that overparameterizing the blind equalizer is the prime cause of ill-convergence. Hence, they proposed to anchor (set the first coefficient to one) the blind equalizer. Anchoring the blind equalizer together with using a convex function guarantees convergence. In [34], the minimum energy is used as a cost function and, hence, we will refer to this algorithm as the "anchored minimum energy algorithm" (AMEA).

In this chapter we will consider anchoring the constant modulus algorithm (CMA). Anchoring the CMA will improve the performance of the convergence property of the original CMA. It is shown that the anchored blind equalizer with the CMA (ACMA) converges to the channel parameters rendering zero ISI provided the channel gain exceeds a certain critical value. If the gain drops below this critical point, the algorithm will converge to a local minimum. This problem can be alleviated by introducing a gain in the equalizer. The speed of convergence of this equalizer will be compared to that of AMEA [34].

This chapter is organized as follows. First we consider using the ACMA for blind equalization of autoregressive and moving average type channels in sections 4.1 and 4.2, respectively. In section 4.3, we present simulation results. We draw conclusions in section 4.4.

4.1 Equalization of Autoregressive Channels

Consider a real AR channel of order n ($AR(n)$), the received signal is given by

$$r_k = g a_k + \sum_{i=1}^{n} \alpha_i r_{k-i},$$

where g is an arbitrary gain, and α_i's are the $AR(n)$ channel parameters. The information symbols (a_k's) are binary, independent and identically distributed with zero mean and unit variance. The moving average (MA) anchored equalizer output has its first tap set to 1, and, therefore, its output is given by

$$\begin{aligned} y_k &= r_k + \sum_{i=1}^{n} w_i r_{k-i} \\ &= g a_k + \sum_{i=1}^{n} (\alpha_i + w_i) r_{k-i} \\ &\triangleq g a_k + isi_k \end{aligned} \tag{4.1}$$

where w_i's are the equalizer's coefficients and isi_k is given by

$$isi_k = \sum_{i=1}^{n} (\alpha_i + w_i) r_{k-i}. \tag{4.2}$$

The CMA exploits the fact that the original constellation has a constant envelope, that is, $E\{|a_k|\} = 1$ for all k. Therefore an appropriate cost function would be

$$J(y_k) = \left(|y_k|^2 - 1\right)^2, \tag{4.3}$$

which is minimized when the equalizer output has a constant modulus ($E\{|y_k|\} = 1$). Using stochastic gradient descent to minimize the above cost function, the update equation for the CMA is given by

$$w_i^{k+1} = w_i^k - \mu r_{k-i} \left(y_k^2 - 1\right) y_k \quad \text{for } i = 1, \cdots, n. \tag{4.4}$$

Figure 4.1 shows the anchored linear equalizer and the CMA control.

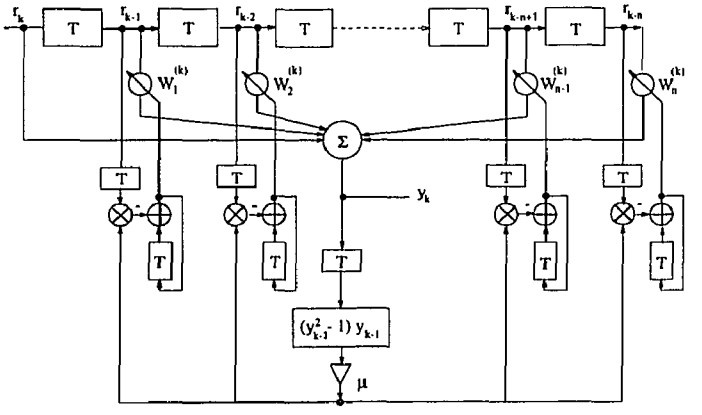

Figure 4.1 Anchored Linear Equalizer with the CMA

Multiply the above equation by $\alpha_i + w_i^k$:

$$w_i^{k+1}\left(\alpha_i + w_i^k\right) = w_i^k\left(\alpha_i + w_i^k\right) - \mu\left(\alpha_i + w_i^k\right) r_{k-i} y_k \left(y_k^2 - 1\right).$$

Now take the expectation of the above conditioned on w_i^k:

$$\alpha_i E\{w_i^{k+1}|w_i^k\} + w_i^k E\{w_i^{k+1}|w_i^k\} = \alpha_i w_i^k + \left(w_i^k\right)^2 - \mu\left(\alpha_i + w_i^k\right) E\{r_{k-i} y_k \left(y_k^2 - 1\right)|w_i^k\}.$$

Steady state is reached when $E\{w_i^{k+1}|w_i^k\} = w_i^k$, and, therefore, we have

$$\left(\alpha_i + w_i^k\right) E\{r_{k-i} y_k \left(y_k^2 - 1\right)|w_i^k\} = 0 \quad \text{for} \quad i = 1, \cdots, n \text{ and all } k. \tag{4.5}$$

The above n equations determine the points of equilibrium of the algorithm. One would have to solve the above equations in order to determine whether the algorithm would converge to the desired values ($w_i = \alpha_i$) and, hence, cancel the ISI completely. Instead we show directly that, under certain conditions for the gain g, equation (4.5) implies complete cancellation of ISI.

Adding the above n equation we get

$$\sum_{i=1}^{n}\left(\alpha_i + w_i^k\right) E\{r_{k-i} y_k \left(y_k^2 - 1\right)|w_i^k\} = 0.$$

Substituting equations (4.1) and (4.2) in the above equation results in

$$\begin{aligned} & E\{isi_k\left((ga_k+isi_k)^2-1\right)(ga_k+isi_k)\,|w_i^k\} \\ = E\{isi_k & \left(g^3a_k^3+\left(3g^2a_k^2-1\right)isi_k+\left(3isi_k^2-1\right)ga_k+isi_k^3\right)|w_i^k\}\}. \\ & = 0 \end{aligned}$$

Now, with the definition (4.2), isi_k depends on the previous data and w_i^k (which itself depends only on the previous data $a_{k-i}, i \geq 1$), therefore isi_k is independent of the current data a_k. Using this together with the fact that both have a zero mean, we get

$$\begin{aligned} E\left\{\left(y_k^2-1\right)y_k\sum_{i=1}^{n}(\alpha_i+w_i)\,r_{k-i}|w_i^k\right\} &= E\left\{isis_k(y_k^2-1)y_k|w_1^k\right\} \\ &= \left(3g^2\sigma_a^2-1\right)E\{isi_k^2|w_i^k\}+E\{isi_k^4|w_i^k\} \\ &= \left(3g^2-1\right)E\{isi_k^2|w_i^k\}+E\{isi_k^4|w_i^k\} \qquad (4.6) \\ &= 0. \end{aligned}$$

If $3g^2-1$ in equation (4.6) is a positive quantity then it can be written as

$$E\{isi_k^4|w_i^k\} = -K^2E\{isi_k^2|w_i^k\} \qquad (4.7)$$

with K positive. However, both $E\{isi_k^4|w_i^k\}$ and $E\{isi_k^2|w_i^k\}$ being positive quantities implies

$$E\{isi_k^4|w_i^k\} = E\{isi_k^2|w_i^k\} = 0,$$

and together with the fact that the expected value of isi_k is 0, we conclude that $isi_k = 0$ with a probability of 1. In sum, if the algorithm reaches a steady state then equation (4.5) is satisfied for $i = 1, \cdots, n$, and from equation (4.1)

$$r_k = ga_k \qquad \text{for all } k.$$

If, however, $3g^2-1$ is negative then equation (4.6) can be written as

$$E\{isi_k^4|w_i^k\} = \Lambda^2E\{isi_k^2|w_i^k\}, \qquad (4.8)$$

where $\Lambda^2 = 1 - 3g^2$, in which case the ISI power is not necessarily zero. This corresponds to a case wherein the algorithm converges to a local minimum, which could be undesirable. We show the existence of such undesirable equilibria by using a simple example.

4.1.1 Undesirable Equilibria

Consider an $AR(n)$ channel with one feedback tap, given by

$$r_k = ga_k + \alpha r_{k-n}.$$

The equalizer output is then given by

$$y_k = ga_k + (w + \alpha)\, r_{k-n}, \tag{4.9}$$

and the ISI term by,

$$isi_k = (w + \alpha) r_{k-n}. \tag{4.10}$$

It is then easy to show that

$$\frac{E\{r_k^4\}}{E\{r_k^2\}} = g^2 \frac{1 + 5\alpha^2}{1 - \alpha^4},$$

and by substituting equation (4.10),

$$\frac{E\{isi_k^4 | w\}}{E\{isi_k^2 | w\}} = g^2 (w + \alpha)^2 \frac{1 + 5\alpha^2}{1 - \alpha^4}. \tag{4.11}$$

Combining equation (4.11) with equation (4.8), we get

$$\begin{aligned} (w + \alpha)^2 &= \Lambda^2 \frac{1 - \alpha^4}{g^2(1 + 5\alpha^2)} \\ &= \left(1 - 3g^2\right) \frac{1 - \alpha^4}{g^2(1 + 5\alpha^2)}, \end{aligned} \tag{4.12}$$

or

$$w = -\alpha \pm \frac{\sqrt{1 - 3g^2}}{g} \sqrt{\frac{1 - \alpha^4}{(1 + 5a^2)}}. \tag{4.13}$$

This clearly shows that the weight w will not converge to the correct channel parameter α.

In particular, following a procedure similar to that of [31], we predict the condition on g and α which results in $w = 0$, and in no ISI cancellation (see equation (4.9)). Setting $w = 0$, in equation (4.13) with $w = 0$, we get

$$g^2 = \frac{1 - \alpha^4}{3 + \alpha^2 + 2\alpha^4}.$$

Therefore if the gain satisfies the above equation, the equalizer will not remove ISI. In conclusion, if the condition $3g^2 - 1 > 0$ is guaranteed, one would ensure that the algorithm would always converge to the correct channel parameters. In other words should the channel gain g be less than $\frac{1}{\sqrt{3}}$, the algorithm will not converge to the no ISI case.

The actual dependence of steady state and ill convergence of the ACMA on the channel gain g is examined in the following example. For the $AR(1)$

$$r_k = ga_k + 0.6r_{k-1}. \tag{4.14}$$

The ACMA equalizer is given by

$$y_k = r_k + w^{(k)}r_{k-1}. \tag{4.15}$$

Using these two equations, we plot in Figure 4.2 the cost function $J(y_k) = E\{(y_k^2 - 1)^2\}$ as a function of w and g. From this figure it is clear that if the gain $g > \frac{1}{\sqrt{3}} = 0.577$ then the cost function has a unique minimum at $w = \alpha = 0.6$. If, however, the gain g drops below $\frac{1}{\sqrt{3}}$, the function will have two minima and a maximum at $w = \alpha$ and therefore the equalizer will not converge to the channel parameter.

One can alleviate this problem by introducing an arbitrary gain, G, in the equalizer. The output of the equalizer is then given by

$$\begin{aligned} y_k &= Gr_k + G\sum_{i=1}^{n} w_i r_{k-i} \\ &= Gga_k + G\sum_{i=1}^{n} (\alpha_i + w_i). \end{aligned} \tag{4.16}$$

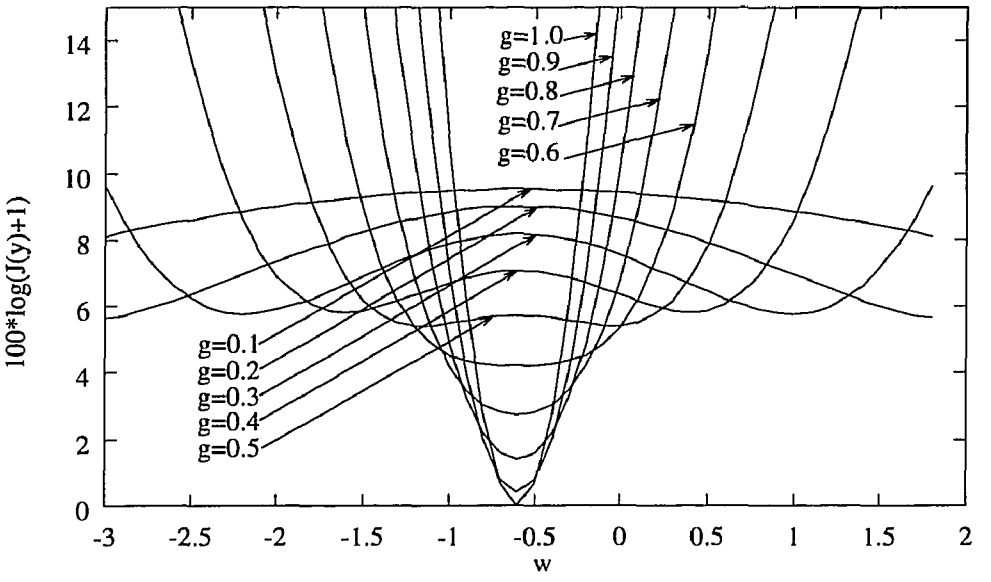

Figure 4.2 Cost Function for Different Gain g

Following a similar procedure as the one above, one can show that at the algorithm's steady state, ISI is cancelled (with a probability of 1) if and only if

$$g > \frac{1}{G\sqrt{3}}. \tag{4.17}$$

Thus, one can choose G appropriately such that condition (4.17) is satisfied. In other words, one would choose G such that for the worst case g condition (4.17) is met. If the worst case g is 0.01, for example, one would choose $G > \frac{100}{\sqrt{3}}$.

4.2 Equalization of Moving Average Type Channels

Consider a real MA type of channel of order n, $MA(n)$, the received signal, is given by

$$r_k = ga_k + \sum_{i=1}^{n} h_i a_{k-i},$$

where g is an arbitrary gain and h_i's are the MA(n) channel parameters. The input to the slicer of the decision feedback equalizer is given by

$$\begin{aligned} y_k &= r_k - \sum_{i=1}^{n} w_i \hat{y}_{k-i} \\ &= g a_k + \sum_{i=1}^{n} h_i a_{k-i} - \sum_{i=1}^{n} w_i \hat{y}_{k-i}, \end{aligned} \tag{4.18}$$

where w_i's are the equalizer's coefficients. Figure 4.3 shows the decision feedback equalization, with the control section.

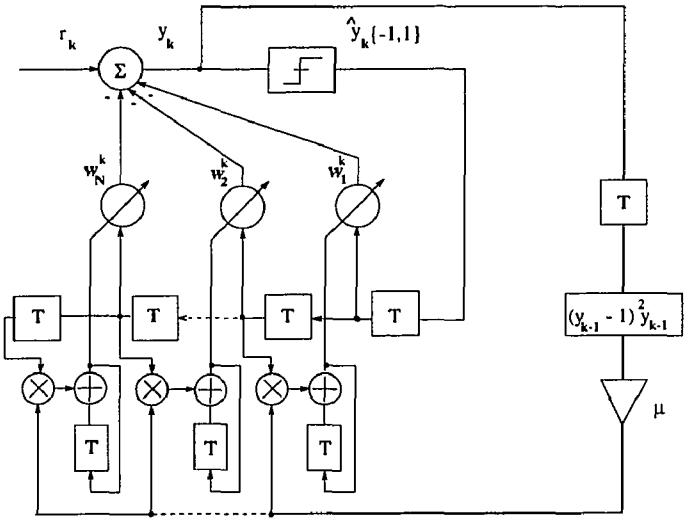

Figure 4.3 The Anchored DFE using the CMA

Now, if we denote the set of all correct decisions by Y_1 and the set of all incorrect decisions by Y'', *i.e.*,

$$Y' = \{\hat{y}_i : \hat{y}_i = a_i\}$$

$$Y'' = \{\hat{y}_i : \hat{y}_i = -a_i\},$$

then equation (4.18) can be written as

$$\begin{aligned} y_k &= g a_k + \sum_{i:\hat{y}_{k-i} \epsilon Y'} (h_i - w_i^k)\hat{y}_{k-i} - \sum_{i:\hat{y}_{k-i} \epsilon Y''} (h_i + w_i^k)\hat{y}_{k-i} \\ &= g a_k + \sum_{i=1}^{n} \gamma_i \hat{y}_{k-i} \\ &\triangleq g a_k + isi_k, \end{aligned} \tag{4.19}$$

where

$$isi_k = \sum_{i=1}^{n} \gamma_i \hat{y}_{k-i},$$

and the γ_i's are given by

$$\begin{aligned} \gamma_i &= (h_i - w_i^k) \qquad \text{for } i : \hat{y}_{k-i} \epsilon Y_1 \\ \gamma_i &= -(h_i + w_i^k) \qquad \text{for } i : \hat{y}_{k-i} \epsilon Y_2. \end{aligned}$$

Applying the constant modulus cost function in equation (4.3) for the DFE, and using equation (4.18) for dy_k/dw_i, we get the following update equation

$$w_i^{k+1} = w_i^k + \mu \hat{y}_{k-i} \left(y_k^2 - 1\right) y_k \text{ for } i = 1, 2, \cdots, n. \tag{4.20}$$

Multiplying equation (4.20) by γ_i,

$$w_i^{k+1} \gamma_i = w_i^k \gamma_i + \mu \gamma_i \hat{y}_{k-i} \left(y_k^2 - 1\right) y_k. \tag{4.21}$$

As before, take expectation conditioned on w_i^k to obtain

$$E\{w_i^{k+1} \gamma_i | w_i^k\} = w_i^k E\{\gamma_i | w_i^k\} + \mu E\{\gamma_i \hat{y}_{k-i} \left(y_k^2 - 1\right) y_k | w_i^k\}. \tag{4.22}$$

It is simple to show that steady state is reached (that is $E\{w_i^{k+1} | w_i^k\} = w_i^k$) if and only if

$$E\{\gamma_i \hat{y}_{k-i} \left(y_k^2 - 1\right) y_k\} = 0 \text{ for all } i \text{ for which } \hat{y}_{k-i} \epsilon Y_1 \text{ or } \hat{y}_{k-i} \epsilon Y_2.$$

Summing on i we have

$$\begin{aligned} E\left\{\sum_{i=1}^{n} \gamma_i \hat{y}_{k-i} \left(y_k^2 - 1\right) y_k | w_i^k\right\} &= E\{isi_k \left(y_k^2 - 1\right) y_k | w_i^k\} \\ &= 0. \end{aligned}$$

This is exactly the same equation we have for the AR(n) channel case. In Figure 4.3 we present a digital implementation of the DFE-ACMA for an MA type channel.

4.3 Simulation

In this section we present simulation results of the anchored constant modulus algorithm as it is applied to the linear equalization for autoregressive channels and to the decision feedback equalization of moving average type channels. In particular, we show the effect of the gain g on the performance.

4.3.1 Linear Equalization

Consider the $AR(1)$ channel whose output signal r_k is given by

$$r_k = ga_k + 0.6r_{k-1}$$

The linear equalizer taps weights are updates using

$$w^{(k+1)} = w^{(k)} - \mu r_{k-1} y_k \left(y_k^2 - 1\right).$$

The averaged squared error of this equalizer is given in Figure 4.4

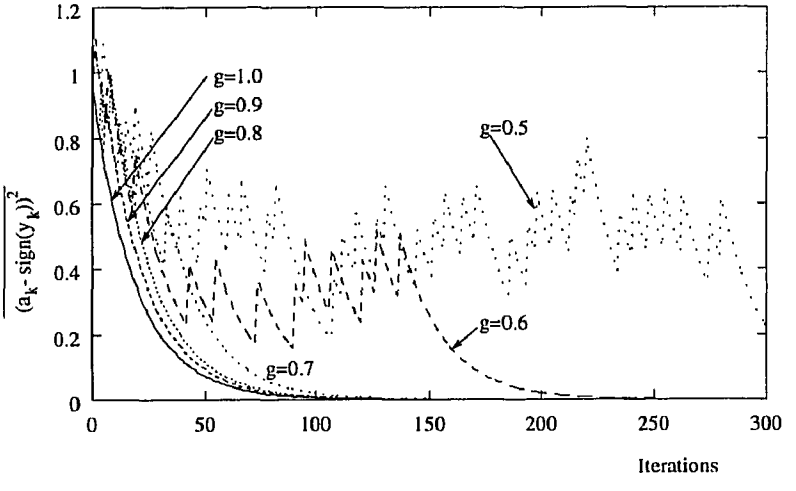

Figure 4.4 Mean of the Squared Error of ACMA for Different Gain g.

4.3.2 Decision Feedback Equalization

To examine the performance of the DFE we consider a channel whose transfer function is given by

$$H(z^{-1}) = g + 0.5z^{-1} - 1.44z^{-2} \tag{4.23}$$

with the corresponding adaptation rule of equation (4.20). The averaged squared error is depicted in Figure 4.5.

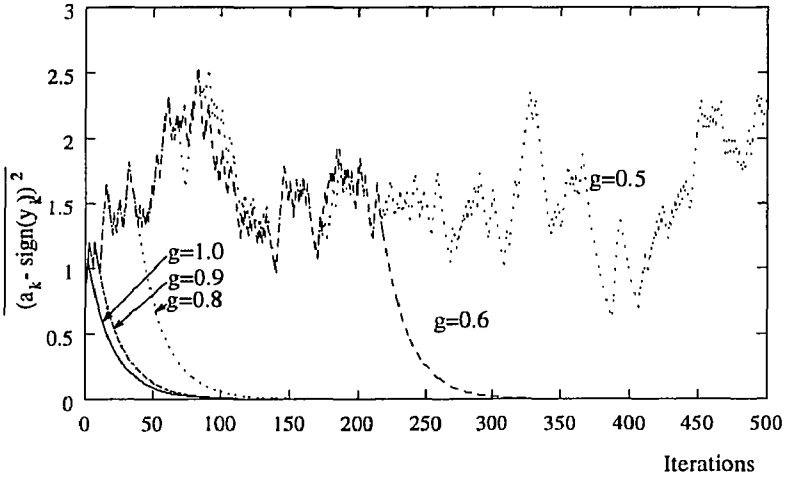

Figure 4.5 Mean of the Squared Error of ACMA for Different Gain g.

In these figures, the estimate of the residual ISI power is obtained by passing the sequence of the squared error$((a_k - \text{sgn}\ (y_k))^2)$ through a smoothing filter whose transfer function is given by $0.05/\left(1 - 0.95z^{-1}\right)$. These figures show clearly that the speed of convergence of the ACMA, for the linear and decision feedback equalizers, depends on the channel gain g. As the gain g the approaches breakpoint, $g = \frac{1}{\sqrt{3}}$, the algorithm takes a longer time to converge. We notice that for $1 \leq g \leq 0.7$, the speed of convergence is nearly constant, reaching approximately zero after 130 iterations, while for $g = 0.6$ the algorithm converges after 250 iterations. The ill convergence of the algorithm is also evident for the gain of $g = 0.5 < \frac{1}{\sqrt{3}}$.

The performance of the ACMA is also compared with the anchored minimum energy algorithm described in [34]. In Figure 4.6, we depict the convergence of this algorithm when used with the $AR(1)$ channel used in equation (4.9). The adaptation rule for the AMEA is given by

$$w^{(k+1)} = w^{(k)} + \mu y_k r_{k-1}$$

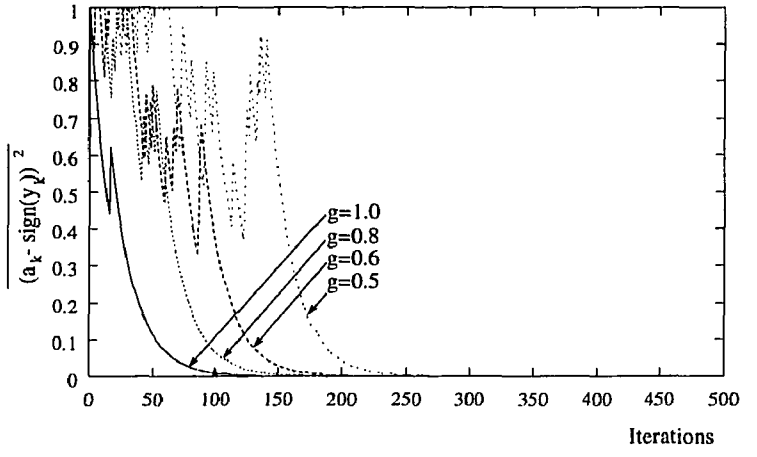

Figure 4.6 Mean of the Squared Error of AMEA for Different Gain g.

As in ACMA, the speed of convergence depends on the channel gain g. However, being globally convergent, the AMEA shows no ill convergence for a small g.

Finally, in Figure 4.7 we compare the rate of convergence of the ACMA with that of the AMEA for $g = 1.0$. The ACMA converges faster than the AMEA. In fact, the ACMA converges to the exact channel parameters such that after approximately 130 iterations, the mean squared error vanishes.

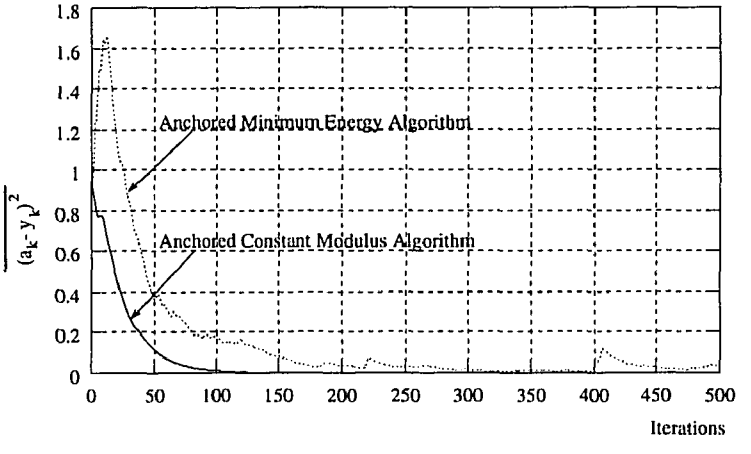

Figure 4.7 Mean of Error Square of the ACMA and AMEA

4.4 Conclusion

In this chapter we used the concept of anchoring the blind equalizer [34] with the constant modulus algorithm for AR and MA channels. We showed analytically and through simulation that the algorithm converges successfully if the unknown channel gain exceeds a certain value ($\frac{1}{\sqrt{3}}$). The algorithm will fail to converge to the desired value if the channel gain drops below this value. This problem can be minimized if we introduce a gain in front of the equalizer. Introducing a gain at the equalizer will not eliminate the problem, but it will lower the critical point below which ill convergence appears.

Compared to the algorithm described in [34], the ACMA converges faster and achieves less mean squared error at the steady state.

CHAPTER 5

BLIND MAXIMUM LIKELIHOOD SEQUENCE ESTIMATION

In chapters 2-4 we considered blind equalizers based on symbol-by-symbol detection. This includes linear and decision feedback equalizers. In this and next chapter we use the maximum likelihood sequence estimation approach, which is based on the entire received sequence [18]. For severely distorted channels, linear equalizers enhance noise, resulting in unsatisfactory performance. The performance of the decision feedback equalizer is, on the other hand, limited by error propagation. The maximum likelihood technique is efficiently implemented using the VA. The MLSE thus offers improved performance over the linear and decision feedback equalizers, but not without an increase in complexity. This point will be addressed in the next chapter, where we describe a techniques for reducing the complexity of VA.

Only recently did blind maximum likelihood sequence estimation (MLSE) start receiving attention [40] [41] [42] [50].

In this chapter we consider a blind maximum likelihood sequence estimation algorithm that has lower complexity compared with existing ones. It also offers a good complexity/speed-of-convergence tradeoff.

This chapter is organized as follows. In section 5.1, we describe the channel model and present the formulation of the problem. The new technique for blind Viterbi equalization is proposed in section 5.2. An illustrative example is given in section 5.3. We derive an upper bound on the probability of bit error, in section 5.4. In section 5.5, we present the simulation results and the computation of the upper bound. Conclusions are drawn in section 5.6.

5.1 Problem Statement

In this section we first introduce the discrete channel model and then the blind maximum likelihood sequence estimation problem. Consider the discrete channel model given below.

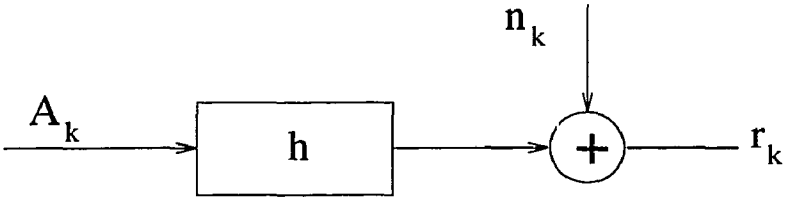

Figure 5.1 Discrete Channel Model

The sampled output of the channel, r_k, at instant k is given by

$$r_k = \mathbf{A}_k'\mathbf{h} + n_k, \tag{5.1}$$

where $\mathbf{A}_k = [a_k, a_{k-1}, \cdots, a_{k-L}]'$ and $\mathbf{h} = [h_0, h_1, \cdots, h_L]'$.

$\{h_i\}_{i=0}^{L}$ is the sampled impulse response of the cascaded transmit, channel and receive filters, assumed slowly time varying, $\{a_{k-i}\}$ is the sequence of transmitted symbols, which are assumed identically distributed independent random variables and $\{n_k\}$ is an additive white noise sequence with Gaussian distribution. At each instant the data takes one of the M possible levels $\{\pm 1, \pm 3, \cdots, \pm(M-1)\}$ with equal probability.

First we consider the problem of estimating a sequence of N transmitted data symbols from a sequence of channel outputs $\mathbf{r}_k = [r_1, r_2, \cdots r_N]'$ for a known channel. There are M^N equi-probable sequences denoted by $\{\mathbf{A}(1), \cdots, \mathbf{A}(M^N)\}$. The ML estimator chooses the most likely sequence $\mathbf{A}_{ML}$ according to

$$\mathbf{A}_{ML} = \arg\max_{\mathbf{A}} f_{\mathbf{r}|\mathbf{A}}(\mathbf{r}|\mathbf{A}), \tag{5.2}$$

where $f_{\mathbf{r}|\mathbf{A}}(\cdot|\cdot)$ is the conditional probability density function (pdf). Since $\{n_k\}$ are iid random variables, one can write

$$\begin{aligned} f_{\mathbf{r}|\mathbf{A}}(\mathbf{r}|\mathbf{A}) &= \prod_{k=1}^{N} f_{r_k|\mathbf{A}}(r_k|\mathbf{A}_k), \\ &= \prod_{k=1}^{N} f_n(r_k - \mathbf{A}_k'\mathbf{h}), \end{aligned} \tag{5.3}$$

where $f_n(\cdot)$ is the Gaussian pdf. In principle the maximization in equation (5.2) should take place through exhaustive search over the M^N sequences, which can be carried out efficiently using the VA [18].

For the blind equalization problem at hand, one might consider the conditional probability of the received sequence $\mathbf{r}_k$ conditioned on both the transmitted sequence and the channel impulse response. Assuming all channel realizations are equally probable, the joint ML estimate for the transmitted data sequence and the channel parameters is given by

$$(\mathbf{A}_{ML}, \mathbf{h}_{ML}) = \arg \max_{\mathbf{A},\mathbf{h}} f_{\mathbf{r}|\mathbf{A},\mathbf{h}}(\mathbf{r}|\mathbf{A}, \mathbf{h}), \tag{5.4}$$

wherein the maximization is carried over all possible channel realizations and transmitted data sequences. Such a problem is not trivial since $\mathbf{h}$ is continuous and $\mathbf{A}$ is discrete.

In [40], Seshadri proposed to solve equation (5.4) by finding the least square estimate of the channel for each possible sequence and then choosing the data sequence with the lowest least square error. This means that one will have to retain all the possible sequences and as a result the complexity will increase exponentially with the message length. Realizing this, it was also proposed in [40] to use a suboptimal search algorithm. In the suboptimum algorithm, one would retain at each node the M ($M \geq 1$) best sequences, as opposed to the VA which retains only the surviving path. The major drawback of such an approach is its complexity.

Ghosh and Weber [41], on the other hand, developed an iterative procedure, whereby one would start with an initial guess of the channel parameters. Given the

initial channel parameters, the VA is used on a frame of observed data to determine the maximum likelihood estimate of the transmitted data. This is then used to obtain a better estimate of the channel. The procedure is iterated until the channel estimate converges. The length of the data frame is an important parameter of the algorithm and has to be large enough to obtain a good channel estimate. In [41] a frame length of 1000 symbols was used.

An iterative approach was also proposed in [42]; however, channel estimation was based on the Expectation Maximization (EM) algorithm [43]. In [50] it was suggested to use numerical techniques to carry out the maximization over the channel parameters. The technique proposed in [50] was based on processing a frame of data and iterating between the VA and the maximizer with respect to the channel.

In [50], it was shown that the estimator given in equation (5.4) would lead to a biased estimate of the channel parameter $\mathbf{h}$. In particular, equation (5.4) can be written as

$$\mathbf{A_{ML}h_{ML}} = \arg \min_{\mathbf{A}} \min_{\mathbf{h}} \frac{1}{K} ||\mathbf{r} - \underline{\mathbf{A}}\mathbf{h}||^2, \tag{5.5}$$

where $||\cdot||^2$ is the l_2-norm, K is the sample size over which the search of $\mathbf{A}$ takes place, and

$$\mathbf{r} = \begin{pmatrix} r_1 \\ \vdots \\ r_{K+L} \end{pmatrix}$$

$$\underline{\mathbf{A}} = \begin{pmatrix} a_1 & 0 & \cdots & 0 \\ a_2 & a_1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & a_K & a_{K-1} \\ 0 & \cdots & 0 & a_K \end{pmatrix}.$$

It was shown that as K approaches infinity, the estimate $\mathbf{h_{ML}}$ will be a biased estimate of the channel parameter vector $\mathbf{h}$. Similar to [41] and [42] an iterative procedure was used, where the maximization with respect to the input was performed using the VA, and the maximization over the continuous channel was performed using

numerical techniques. The latter maximization is performed on a frame of data. It was demonstrated that with a frame length of $K = 1000$, the error performance of blind VA closely matches that of the conventional VA (with a known channel) and for $K = 100$ the blind VA departs from the conventional one.

Since [50] deals with the general problem given in equation (5.4), the results are applicable to the techniques given in [41], [42] and the technique developed in this chapter. The major drawback with the techniques reported in [40] [41] [42] [50] is complexity.

5.2 The Proposed Technique

We assume a quantized channel (this is justified in practice, since finite precision processors are used to implement the algorithm) and develop two trellises: one for the channel and the other for the data. The resulting scheme offers a considerable reduction in the computational complexity compared with [41] [42]. It also prevails over existing techniques with a good complexity/performance tradeoff.

The key observation is that if the channel is discrete, one could interchange the roles of data and channel parameters in the VA branch metric. That is, if the data is known one would search a channel trellis for the ML channel parameters and vice versa. Therefore, we propose to use two trellises, one for data and the other for channel. Two VAs are used to search two trellises in parallel, one for the data and the other for the channel. The output of one is fed into the metric calculator of the other. This joint maximization eventually converges to the estimate given in equation (5.4). The resulting scheme has a considerably lower complexity compared with existing techniques. It also offers a good complexity/performance tradeoff.

The channel parameters are approximated by discrete values from the infinite alphabet $\{0, \pm c, \pm 2c, \cdots\}$, where c can be chosen to be arbitrarily small. With such a channel alphabet, the corresponding channel trellis will have an infinite number of

states. However, since the channel vector $\mathbf{h}$ does not vary much from one signaling interval to the other, as the data vector $\mathbf{A}$, we need not consider all possible levels of the channel parameters at a given instant. In order to reduce complexity, we propose a simple state assignment scheme for the channel trellis. The next channel estimate, $\mathbf{h}^{i+1}$, in the proposed scheme is given by

$$\mathbf{h}^{i+1} = \mathbf{h}^{i} \quad \text{for state } 0$$

and

$$\mathbf{h}^{i+1} = \mathbf{h}^{i} \pm c \cdot \mathbf{1}_n \quad \text{for state } n = 1, 2, \cdots,$$

where $\mathbf{1}_n$ is a vector of length $L+1$ whose elements are either zeros or ones. For the special case when $\mathbf{1}_n = \mathbf{0}$, the degenerate state 0 results. Clearly the number of states (the maximum number of states is 2^{L+1}) does not depend on the parameter c but on the channel memory L.

A smaller number of states can be used if the vector $\mathbf{1}_n$ is restricted to be all zero except for the unity at the nth location to unity. It is clear that the above state assignment results in $L+2$ states. Therefore the number of states increases linearly with L. The branches emerging from all states, except for state 0, have two parallel transitions, one corresponding to an increment $(+c)$ and the other to a decrement $(-c)$. There are other state assignment schemes with less than 2^{L+1} states, but the above assignment will result in a simple trellis.

The algorithm will proceed as follows:

1. Start with an initial channel estimate, $\hat{\mathbf{h}}_{ML} = \mathbf{h}^0$.

2. Use the VA to solve for

$$\hat{\mathbf{A}}_{\mathrm{ML}} = \arg\max_{\mathbf{A}} f(\mathbf{r}|\mathbf{A}, \hat{\mathbf{h}}_{ML}),$$

with the branch metric $\left(r_k - \sum_{i=1}^{L} \hat{h}_i a_{k-i}\right)^2$.

3. Use the VA to solve for

$$\hat{\mathbf{h}}_{\mathbf{ML}} = \arg\max_{\mathbf{h}} f(\mathbf{r}|\hat{\mathbf{A}}_{ML}, \mathbf{h}_{\mathbf{ML}}),$$

with the branch metric $\left(r_k - \sum_{i=1}^{L} h_i \hat{a}_{k-i}\right)^2$.

4. Iterate 2 and 3.

It can be noticed that the algorithm achieves the ML estimate of the channel by adaptively incrementing or decrementing the previous estimate. Using the channel state table above, we change one channel parameter per transition. To improve the speed of convergence one can add more states to the channel trellis, which allows one to change two or more parameters at a time. This will significantly improve the rate of convergence at the expense of complexity. Thus, one can compromise rate of convergence to complexity.

Another parameter that affects the performance is the step parameter c. Choosing a smaller c will reduce the rate of convergence, but will improve the error rate. This point is demonstrated in the following example.

5.3 An Illustrative Example

The algorithm described above was used to equalize the channel (assumed unknown) whose sampled impulse response is given by

$$h(n) = 0.407 \cdot \delta(n) + 0.815 \cdot \delta(n-1) + 0.407 \cdot \delta(n-2),$$

where $\delta(\cdot)$ is the Kronecker delta function. For simplicity, binary transmission is assumed, and therefore with $L = 2$, the channel and data trellises will have 4 states each. The states of the channel trellis are given by

$$\begin{aligned} \mathbf{h}^{i+1} &= \mathbf{h}^{i} \quad \text{for state } 0 \\ \mathbf{h}^{i+1} &= \mathbf{h}^{i} \pm c \cdot \mathbf{1}_n \quad \text{for state } n = 1, 2, 3. \end{aligned}$$

The vector $\mathbf{1}_n$ is in this case has all it elements zero except the nth element equals one. The trellises are shown in Figure 5.2

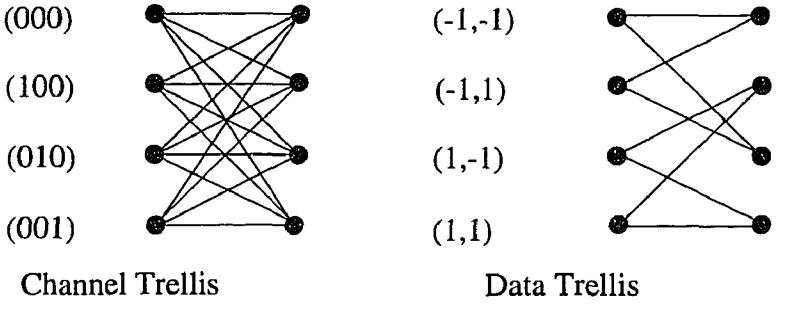

Figure 5.2 Channel and Data Trellises

The channel was initialized to $\mathbf{h}^0 = (0\ 0\ 0)$. The estimation error for three different values of c were determined by simulation at an SNR of 10dB. The estimation error, defined as the mean squared difference between the true and estimated channel parameters, is depicted in Figure 5.3.

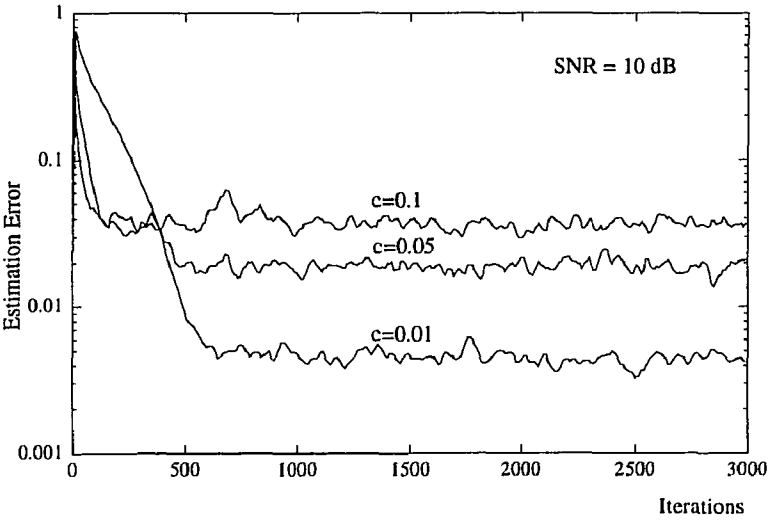

Figure 5.3 Estimation Error for Different Values of c

5.4 Probability of Error

In this section we investigate the error performance of the blind sequence estimation scheme developed in section 5.2. We will use Forney's approach [18].

Define the state of the channel $s(k)$ at time k by

$$s(k) = (a_{k-1}, a_{k-2}, \cdots, a_{k-i}),$$

where i is the memory of the channel, and denote the corresponding state estimated by the VA by $\hat{s}(k)$, where

$$\hat{s}(k) = (\hat{a}_{k-1}, \hat{a}_{k-2}, \cdots, \hat{a}_{k-i}).$$

The sequence $\{\hat{a}_{k-1}, \hat{a}_{k-2}, \cdots, \hat{a}_{k-i}\}$ is the sequence estimated by the VA. Following Forney's approach [18], an error event $\mathcal{E}$ is said to occur between $k = k_1$ and $k = k_2$, if $\hat{s}(k_1) = s(k_1)$, $\hat{s}(k_2) = s(k_2)$ and $\hat{s}(k) \neq s(k)$ for all k, $k_1 < k < k_2$. Since $\hat{s}(k) = s(k)$ for $k = k_1, k_2$, it follows that

$$(\hat{a}_{k_1-1}, \hat{a}_{k_1-2}, \cdots, \hat{a}_{k_1-i}) = (a_{k_1-1}, a_{k_1-2}, \cdots, a_{k_1-i})$$

and

$$(\hat{a}_{k_2-1}, \hat{a}_{k_2-2}, \cdots, \hat{a}_{k_2-i}) = (a_{k_2-1}, a_{k_2-2}, \cdots, a_{k_2-i}).$$

Now, define the error sequence associated with the event $\mathcal{E}$ as

$$\mathbf{e} \triangleq \{e_{k_1}, e_{k_1+1}, \cdots, e_{k_2-i-1}\},$$

where $e_k \triangleq a_k - \hat{a}_k$. The Euclidean distance $d_i^2(\mathcal{E})$ of the error event is given by

$$d_i^2(\mathcal{E}) = \sum_{k=k_1}^{k_2} \left(\sum_{j=0}^{\min(k-k_1, i)} h_j e_{k-j} \right)^2. \tag{5.6}$$

5.4.1 Probability of an Error Event

The probability of an event error associated with an error sequence **e**, is now derived. Following Forney's approach for an error event to occur, three sub-events must take place:

$\mathcal{E}_1$: At time k_1 we must have $s(k_1) = \hat{s}(k_1)$.

$\mathcal{E}_2$: The input sequence $\mathbf{A}$ between k_1 and $k_2 - i - 1$

$$\mathbf{A} \triangleq \{a_{k_1}, a_{k_1+1}, \cdots a_{k_2-i-1}\}.$$

must be such that $\mathbf{A} + \mathbf{e}$ is an allowable sequence. For the binary transmission of ± 1, if $e_{k_1} = 2$ then the corresponding input symbol must be $a_{k_1} = 1$.

$\mathcal{E}_3$: Between k_1 and k_2, the noise terms must be such that the estimated sequence $\{\hat{a}_j : k_1 \leq j < k_2\}$ accumulates greater likelihood than the transmitted sequence $\{a_j : k_1 \leq j < k_2\}$.

Event $\mathcal{E}_2$ is independent of $\mathcal{E}_1$ and $\mathcal{E}_3$; therefore, we can write

$$P\{\mathcal{E}\} = P\{\mathcal{E}_2\}P\{\mathcal{E}_3\}P\{\mathcal{E}_1|\mathcal{E}_3\}.$$

5.4.2 Probability of $\mathcal{E}_2$

For binary transmission considered in this chapter, $P\{\mathcal{E}_2\}$ given by

$$\begin{aligned} P\{\mathcal{E}_2\} &= \prod_{k=k_1}^{k_2-i-1} \frac{2 - \frac{|e_k|}{2}}{2} \\ &= 2^{-w(\mathbf{e})}, \end{aligned}$$

where $w(\mathbf{e})$ is the number of non zero elements in $\mathbf{e}$.

5.4.3 Probability of $\mathcal{E}_3$

Define the received signal sample by

$$r_k = x_k + n_k,$$

where $x_k = \mathbf{h}'\mathbf{A}$. Now consider the blind scheme introduced in the previous chapter. We assume that at steady state, the estimated channel parameters are given by $\hat{\mathbf{h}}$,

and further define

$$\begin{aligned} y_k &\triangleq \hat{\mathbf{h}}'\mathbf{A} \\ \hat{y}_k &\triangleq \hat{\mathbf{h}}'\hat{\mathbf{A}} \text{ where } \hat{\mathbf{h}}' \text{ is the estimated parameter vector of the channel} \\ \epsilon_k &\triangleq x_k - y_k \\ &= (\mathbf{h}' - \hat{\mathbf{h}}')\mathbf{A} \\ &= \Delta\mathbf{h}'\mathbf{A}. \end{aligned} \tag{5.7}$$

Then we have

$$\begin{aligned} P\{\mathcal{E}_3\} &= P\left\{\sum_{k_1}^{k_2}(r_k - y_k)^2 - (r_k - \hat{y}_k)^2 > 0\right\} \\ &= P\left\{\sum_{k_1}^{k_2}(x_k + n_k - y_k)^2 - (x_k + n_k - \hat{y}_k)^2 > 0\right\} \\ &= P\left\{\sum_{k_1}^{k_2}(\epsilon_k + n_k)^2 - (y_k - \hat{y}_k + \epsilon_k + n_k)^2 > 0\right\} \\ &= P\left\{2\sum_{k_1}^{k_2}(\hat{y}_k - y_k)(n_k + \epsilon_k) > \sum_{k_1}^{k_2}(y_k - \hat{y}_k)^2\right\} \\ &= P\left\{\langle\hat{\mathbf{y}} - \mathbf{y}, \mathbf{n} + \epsilon\rangle \;\; > \frac{||\mathbf{y} - \hat{\mathbf{y}}||^2}{2}\right\}, \end{aligned}$$

where the $k_2 - k_1$ dimensional vectors $\mathbf{y}$, $\hat{\mathbf{y}}$, $\mathbf{n}$ and ϵ are defined as

$$\begin{aligned} \mathbf{y}' &\triangleq (y_{k_1}, y_{k_1+1}, \cdots y_{k_2}) \\ \hat{\mathbf{y}}' &\triangleq (\hat{y}_{k_1}, \hat{y}_{k_1+1}, \cdots \hat{y}_{k_2}) \\ \mathbf{n}' &\triangleq (n_{k_1}, n_{k_1+1}, \cdots, n_{k_2}) \\ \epsilon &\triangleq (\epsilon_{k_1}, \epsilon_{k_1+1}, \cdots, \epsilon_{k_2}). \end{aligned}$$

$\langle\mathbf{a}, \mathbf{b}\rangle$ is the inner product of the vectors $\mathbf{a}$ and $\mathbf{b}$ and $||\cdot||$ is the l_2 norm. The probability of event $\mathcal{E}_3$ is thus given by

$$P\{\mathcal{E}_3\} = P\left\{\tilde{n} > \frac{||\mathbf{y} - \hat{\mathbf{y}}||}{2} - \tilde{\epsilon}\right\}, \tag{5.8}$$

where

$$\begin{aligned} \tilde{n} &= \frac{< \hat{\mathbf{y}} - \mathbf{y}, \mathbf{n} >}{||\mathbf{y} - \hat{\mathbf{y}}||} \\ \tilde{\epsilon} &= \frac{< \hat{\mathbf{y}} - \mathbf{y}, \epsilon >}{||\mathbf{y} - \hat{\mathbf{y}}||}. \end{aligned} \tag{5.9}$$

The scalar quantities $\tilde{n}$ and $\tilde{\epsilon}$ are the projections of the vectors $\mathbf{n}$ and ϵ on $\mathbf{y} - \hat{\mathbf{y}}$, respectively. The quantity $\tilde{\epsilon}$ indicates the amount of mismatch between the estimated and actual channel parameters: It is proportional to the step c and the length of the error event $k_2 - k_1$. A two-dimensional representation of the different vectors and scalars described so far is shown in the following figure.

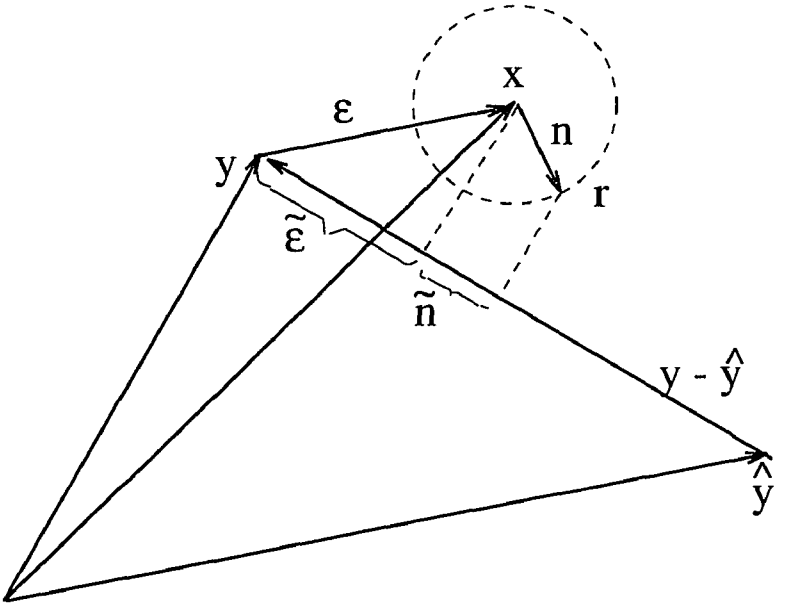

Figure 5.4 Representation of Different Vectors

Since $\mathbf{n}$ is a vector of iid Gaussian random variables, each with a mean of zero and a variance of σ^2 , it can be shown that $\tilde{n}$ is a Gaussian random variable with a zero mean and variance of σ^2. Therefore,

$$\begin{aligned} P\{\mathcal{E}_3\} &= Q\left(\frac{||\hat{\mathbf{y}} - \mathbf{y}|| - 2\tilde{\epsilon}}{2\sigma}\right) \\ &= Q\left(\frac{d(\mathcal{E}) - 2\tilde{\epsilon}}{2\sigma}\right) \end{aligned}$$

$$= Q\left(\frac{\tilde{d}(\mathcal{E})}{2\sigma}\right), \tag{5.10}$$

where

$$\begin{aligned} Q(u) &= \frac{1}{\sqrt{2\pi}}\int_u^\infty e^{-v^2/2}dv \\ \tilde{d}(\mathcal{E}) &= d(\mathcal{E}) - 2\tilde{\epsilon}. \end{aligned}$$

5.4.4 Probability of $\mathcal{E}_1$ Given $\mathcal{E}_3$

Following Forney's argument, the probability, $P\{\mathcal{E}_1|\mathcal{E}_3\}$, is closely overbounded by 1, for moderate SNR. Therefore, the probability of occurrence of $\mathcal{E}$ is given by

$$\begin{aligned} P\{\mathcal{E}\} &= P\{\mathcal{E}_2\}P\{\mathcal{E}_3\}P\{\mathcal{E}_1|\mathcal{E}_3\} \\ &\leq 2^{-w(\mathbf{e})}Q\left(\frac{\tilde{d}(\mathcal{E})}{2\sigma}\right). \end{aligned} \tag{5.11}$$

Further, denote the set of all possible error events starting at k_1 by E, then summing over all events starting at k_1, we get

$$P\{E\} \leq \sum_{\mathcal{E}\in E} P\{\mathcal{E}\}.$$

where $P\{E\}$ is the probability that any error event starts at k_1. The above upper bound can alternatively written as

$$P\{E\} \leq \sum_{\tilde{d}\in D} Q\left(\frac{\tilde{d}}{2\sigma}\right), \sum_{\mathcal{E}\in E_{\tilde{d}}} 2^{-w(\mathbf{e})} \tag{5.12}$$

where D is the set of all possible Euclidean distances and $E_{\tilde{d}}$ is the set of all error events with Euclidean distance $\tilde{d}$. For moderate SNR the term involving the minimum distance will dominate the above summation; therefore, at moderate to high SNR, we have

$$P\{E\} \leq Q\left(\frac{\tilde{d}_{min}}{2\sigma}\right) \sum_{\mathcal{E}\in E_{\tilde{d}}} 2^{-w(\mathbf{e})}. \tag{5.13}$$

For binary transmission, the number of bit errors associated with an error event is equal to the number of non-zero elements of the error vector $\mathbf{e}$, which is given by

$w(\mathbf{e})$. Thus the probability of error, P_e, is upper- bounded by

$$\begin{aligned} P_e &\leq Q\left(\frac{\tilde{d}_{min}}{2\sigma}\right) \sum_{\mathcal{E} \in E_{\tilde{d}}} 2^{-w(\mathbf{e})} w(\mathbf{e}) \\ &= J \cdot Q\left(\frac{\tilde{d}_{min}}{2\sigma}\right) \end{aligned} \tag{5.14}$$

at moderate to high SNR. The coefficient J is given by

$$J = \sum_{\mathcal{E} \in E_{\tilde{d}}} 2^{-w(\mathbf{e})} w(\mathbf{e}).$$

The essential difference between the above derivation and the one given in [18], is that here we considered the effect of channel estimation error. This error will reduce the effective minimum distance and hence degrade the performance.

The degradation in performance depends on the quantity $\tilde{\epsilon}$ given in equation (5.9). It can be shown that

$$\tilde{\epsilon} \leq ||\epsilon||.$$

By approximating $||\epsilon||$ by $(k_2 - k_1) \cdot c$, we will show in the next section that the resulting bound is a valid one for different values of c.

5.5 Simulation Results and Upper Bounds

5.5.1 Simulation Results

For evaluating the upper bounds derived in the previous section, we used the same channel as in section 5.3. The impulse response of the channel is given by

$$h(n) = 0.407 \cdot \delta(n) + 0.815 \cdot \delta(n-1) + 0.407 \cdot \delta(n-2).$$

The estimated probability of bit error was found through simulation for different values of c and for the known channel case. Figure 5.5 shows the probability of bit error versus the received bit energy-to-noise power ratio, which is given by

$$\left(\frac{E_b}{N_0}\right)_{\text{dB}} = 10 \log_{10} \frac{1}{2\sigma^2}.$$

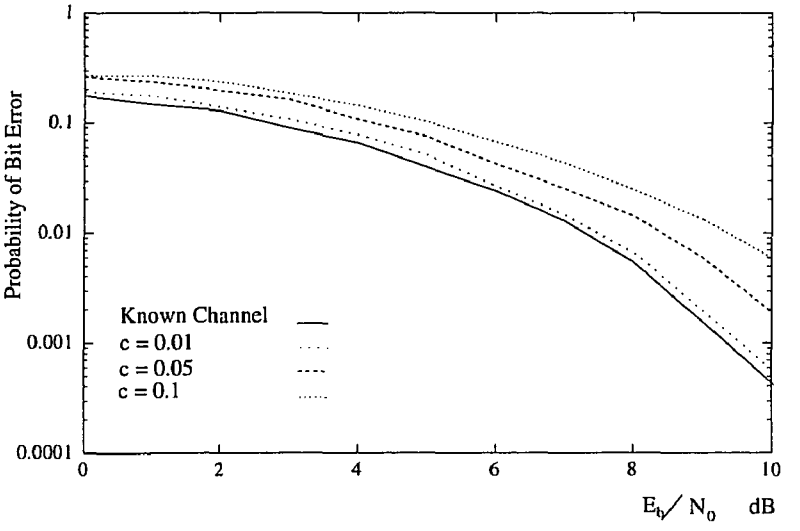

Figure 5.5 Probability of Bit Error (Simulation)

With $c = 0.01$ the probability of error closely follows that of the ideal VA. For the $c = 0.05$ curve it can be seen that there is a loss of less than 1 dB. However as the value c increases beyond 0.05, the degradation from the ideal case becomes more pronounced.

5.5.2 Upper Bound

We consider first the ideal case, MLSE with a known channel. For the channel under consideration, there are an infinite number of error events of the form $\mathbf{e} = \pm\{2, -2, 2, -2, \cdots, 0, 0\}$ (*i.e.* error symbols have alternating signs), all achieving the minimum distance of $\sqrt{2.67}$. Referring to equation (5.14), the error coefficient J is given by

$$\begin{aligned} J &= \sum_{n=1}^{\infty} 2(n+1)\frac{1}{2^{n+1}} \\ &= 3. \end{aligned}$$

Therefore, the probability of bit error is over-bounded by

$$P_e \leq 3Q\left(\frac{\sqrt{2.67}}{2\sigma}\right),$$

for moderate SNR.

For the blind case, as mentioned at the end of the previous section, we take $||\epsilon|| = n \cdot c$, where n is the length of the error event. This means that we assume all channel parameters were estimated correctly except one, which deviates from the correct one by a factor of c. It will be demonstrated that with this assumption, one will get a valid upper bound. The upper bound in this case is given by

$$P_e \leq \sum_{n=1}^{\infty} 2(n+1)\frac{1}{2^{n+1}} \cdot Q\left(\frac{\sqrt{2.67} - 2\sqrt{n} \cdot c}{2\sigma}\right).$$

The figure below shows the upper bound for $c = 0.01, 0.05$ and 0.1.

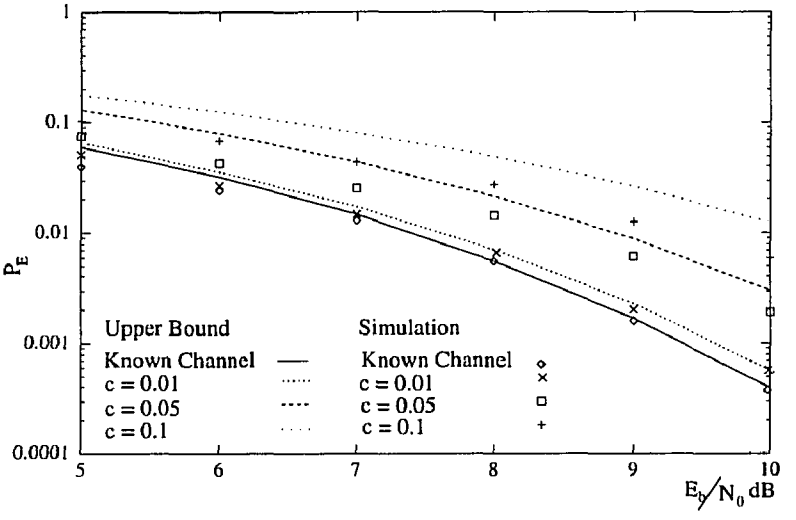

Figure 5.6 Upper Bounds on the Probability of Bit Error

The above figure shows the upper bounds for different values of c and for the ideal MLSE. It can be seen that the bound developed in the previous section is a valid one. It is also evident that the assumption we made at the end of the previous section, $||\epsilon|| = n \cdot c$ is also valid. As expected, the simulation results lie below the

upper bound which demonstrates that on the average

$$|\mathbf{h} - \hat{\mathbf{h}}| \leq c \cdot \mathbf{1}.$$

where $\mathbf{1}$ a vector of length L with only one non-zero element which is equal to one.

5.6 Conclusion

A new algorithm for blind Viterbi equalization was proposed. It approximates the continuous level channel model by a discrete one. A channel state assignment scheme was presented that leads to a simple channel trellis. The number of states of the channel trellis increase linearly with the channel memory. The channel and data trellises are used to find the joint maximum likelihood channel and data estimates.

The algorithm offers a good complexity/performance tradeoff. It also compromises complexity for faster convergence and lower error rates. The rate of convergence depends directly on the parameter c. For $c = 0.01$, the probability of error of the blind scheme is very close to that for the conventional one. With $c = 0.05$, the probability of error degraded by less than 0.5 dB.

This blind scheme could also be used with reduced complexity trellises discussed in Chapter 6. In this way the overall complexity can be varied.

CHAPTER 6

REDUCED STATE VITERBI EQUALIZATION

The type of equalization used to mitigate ISI caused by noisy linear channels can be divided into two classes. The first, symbol-by-symbol equalization, encompasses linear and decision feedback equalization. The second involves maximum likelihood sequence estimation (MLSE) [18], where the Viterbi algorithm (VA) is used to solve the estimation problem.

While the first class has low complexity and a high error rate, the second has a lower error rate at the expense of complexity. The complexity of the VA grows exponentially with the length of the channel impulse response. When the impulse response becomes larger, the VA becomes impractical, and methods for complexity reduction are needed.

Research has been directed toward obtaining reduced-complexity equalizers, while maintaining MLSE performance as close as possible. To reduce the complexity, a number of authors have proposed incorporating a linear or decision feedback preprocessor so that the MLSE will be deal with an equivalent channel with a shorter impulse response [53] [54]. In [53], a linear equalizer was used to shorten the impulse response of the channel, while in [54] a DFE was used to truncate the length of the channel. Such approaches were found to limit the performance of the combined system.

Recently, Eyuboğlu and Qureshi [20] and Duel-Hallen and Heegard [21] have proposed sequence estimators which provide a good performance/complexity tradeoff. The technique in [20] is useful for systems utilizing a large signal constellation, while that in [21], which is a special case of [20], is suitable for channels with a long impulse response.

In [21] the complexity of the VA is reduced by considering a few states of the channel. The ISI due to the rest of the states is estimated using a feedback detector analogous to that of the decision feedback equalizer (DFE). The estimated ISI is then used in the branch metric computation. As in the DFE, error propagation affects the performance of the algorithm. The degradation due to error propagation was found to be less than that of the DFE.

In this chapter, a new technique is presented for reducing the complexity of the VA for channels with long memory. The technique offers more flexibility in the choice between performance and complexity than the one in [21]. It can generate trellises with any number of states rather than only power of 2 states as in [20] [21].

The motivation to this chapter is the work on error propagation for the DFE given in [15]. The error sequences for the DFE can be modeled as a Markov chain, whose number of states is exponential in the number of distinct magnitudes of error and the number of past decisions that influence the current decisions. The complexity of the resulting systems is extremely high. To reduce systems complexity Duttweiler, *et al.*, [15] proposed a reduced state machine. This can be viewed as grouping error sequences in order to reduce the complexity. This grouping can also be envisioned as partitioning the set of all possible error sequences in a unique manner. This led to a technique for reducing complexity, which is presented in this chapter.

This chapter is organized as follows. In section 6.1, we present the channel model and introduce the partitioning approach for reducing the complexity of the VA. In section 6.2, we describe a general procedure to do the partitioning. An example is given in section 6.3. In section 6.4, we discuss the probability of error for the reduced complexity scheme. Conclusions are given in section 6.5. A summary of the results presented in this chapter can be found in [22], and more detailed results in [52].

6.1 Channel Model and the Proposed Technique

In this section we will consider channels with finite impulse response. The discrete time channel model considered here is given in Figure 6.1. This model arises in a pulse amplitude modulation (PAM) system at the output of a sampled, whitened matched filter [18]. The channel $h(D)$ is modeled as a finite response filter (FIR), and $n(D)$ is a white Gaussian noise source with a zero mean and variance of σ^2. The data sequence $a(D)$ consists of symbols a_k, which are independent and identically distributed.

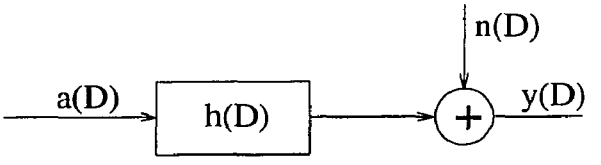

Figure 6.1 The Discrete Channel Model

We will assume binary transmission in this chapter. Therefore, data symbols a_k take values of ± 1 with equal probability. Referring to Figure 6.1, the output $y(D)$ is given by

$$y(D) = a(D)h(D) + n(D),$$

where $h(D)$, given by $h(D) = h_0 + h_1 D + \cdots + h_n D^n$, defines the channel impulse response, whose degree is determined by the channel memory. The state of the channel, $s(k)$, at time k is defined by the binary n tuple given by $(a_{k-1}, a_{k-2}, \cdots, a_{k-n})$, the previous input data. Therefore, at any time k there are 2^n possible states. We denote the set of channel states by Ω; then

$$\Omega = \{s_i : s_i \text{ is a state of the system }, i = 0, 1, \cdots, 2^n - 1\}.$$

In the proposed technique, the set Ω is divided into N, S_i, subsets, where N is $2 \leq N \leq 2^n$, such that

1. $\bigcup_{i=0}^{N-1} S_i = \Omega$

2. $S_i \cap S_j = \emptyset$; the empty set, for $i \neq j$ and $0 \leq i, j \leq N-1$

3. The subsets S_i are chosen such that for all $s_n(k) \in S_i$, the corresponding next state $s_n(k+1)$ must belong to one subset.

The first two conditions specify a partition on the set Ω, and hence one could also specify an equivalence relation on Ω. The third condition is a constraint on the partitions that enables a trellis to be defined. Thus, not every partition on Ω could be a candidate; only those that result in a trellis are suitable. A procedure is devised for defining partitions that satisfy the third condition. This is detailed in section 6.2. The resulting trellis will have N states.

At this point one should emphasize the difference between the partitioning considered in this chapter and that in [20]. In [20] the signal constellation (signal set) is partitioned into different levels so that each element $a_{k-i}, 1 \leq i \leq m$, of state vector $s(k)$, is assigned to a subset. A subset trellis having a smaller number of states than the original trellis is then defined. In this chapter we are partitioning the set of channel states. Only when the number of states per trellis is a power of 2, will our technique result in trellises similar to those reported in [20].

The branch metric for the MLSE is given, by $(y_k - \sum_{i=1}^{n} h_i a_{k-i} - a_k)^2$. Since each state in the reduced trellis is a union of two or more channel states, an ambiguity will result in the branch metric calculation. That is, the branch metric is no longer uniquely determined by the previous/present trellis states' pair. Similar to [20] [21], a feedback mechanism is introduced to resolve this ambiguity. The branch metric associated with the reduced trellis is given by $(y_k - \sum_{i=1}^{l} h_i a_{k-i} - \sum_{l+1}^{n} h_i \hat{a}_{k-i} - a_k)^2$, where $l < m$ is determined by the reduced trellis. The previous state estimate $(\hat{a}_{k-l-1}, \cdots, \hat{a}_{k-n})$ is stored in the path history associated with the present state.

6.2 The Partitioning Procedure

A partition of a set Ω is a pairwise disjoint collection of non empty subsets of Ω, whose union is Ω. It is known that an equivalence relation in Ω defines a partition of Ω, and, conversely, a partition in Ω yields an equivalence relation. Given an equivalence relation R in Ω, let $R(a) \triangleq \{x \in \Omega : aRx\}$ for each $a \in \Omega$. $R(a)$ is known as an equivalence class of R and is a subset of Ω. The collection of subsets, $\{R(a) : a \in \Omega\}$, is a partition of Ω. A collection of equivalence relations $\{R_1, R_2, \cdots, R_n\}$ is known as an equivalence sequence iff for all i,j $1 \leq i \leq j \leq n$, and all $x, y \in \Omega$ we have $xR_jy \Longrightarrow xR_iy$. That is, $R_n(x) \subset R_{n-1}(x) \subset \cdots \subset R_1(x) \subset \Omega$.

For the channel model described in the previous section, the states of the channel are given by binary n tuples. Consider the equivalence relation R_i given by: xR_iy iff the first i components of the n-tuples x and y are identical, for any states x and $y \in \Omega$. It can be shown that R_i is an equivalence relation. It can also be shown that the sequence $\{R_1, R_2, \cdots, R_n\}$ is an equivalence sequence.

Figure 6.2 shows the different levels of partitioning and the corresp onding subsets. Label the subsets at the ith level, with binary i tuples. It should be noted that there are 2^i subsets at level i, each with cardinality 2^{n-i}. The number of subsets will determine the number of states of the trellis.

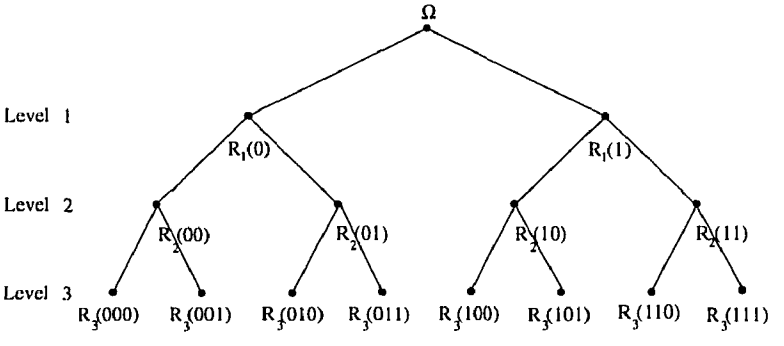

Figure 6.2 The Partitioning Tree

We will now show that the equivalence sequence $\{R_1, R_2, \cdots, R_n\}$ can define partitions that will result in a reduced complexity trellis.

In order to show that the partitions associated with the above equivalence sequence satisfy the third constraint, we first present it in a mathematical form.

Define two functions $F_1(x)$ and $F_{-1}(x)$ as the next state of the channel, when the present state is x, and inputs are 1 and -1 respectively. That is, if $x = (x_1, x_2, \cdots, x_n)$, then

$$\begin{aligned} F_1(x) &= (1, x_1, \cdots, x_{n-1}) \\ \text{and} \qquad F_{-1}(x) &= (-1, x_1, \cdots, x_{n-1}). \end{aligned}$$

To meet the third condition, the equivalence relation R must satisfy

$$\begin{aligned} xRy &\implies F_1(x)RF_1(y) \text{ and} \\ &\implies F_{-1}(x)RF_{-1}(y) \quad \text{for all } x \text{ and } y \in \Omega. \end{aligned} \tag{6.1}$$

When the above statements are satisfied, the functions $F_1(\cdot)$ and $F_{-1}(\cdot)$ are said to be compatible with R. Equation (6.1) can also be written as,

$$\begin{aligned} \text{for all } x \text{ and } y \in \Omega \quad y \in R(x) &\implies F_1(y) \in R(F_1(x)) \text{ and} \\ &\implies F_{-1}(y) \in R(F_{-1}(x)). \end{aligned} \tag{6.2}$$

6.2.1 Trellises with 2^m States

Now we are ready to show that the partition obtained from the different relations satisfies the third constraint, *i.e.*, equations. (6.1) and (6.2). Consider the equivalence relation R_i; the set $\{F_1(y) : y \in R_i(x)\}$ for some $x \in \Omega$ is the set of all channel states that have the first $i+1$ components identical. The first element being 1 and the consecutive i elements are identical, since the previous state $y \in R_i(x)$. Therefore, one can write:

$$\begin{aligned} \{F_1(y) : y \in R_i(x)\} &= R_{i+1}(F_1(x)) \\ &\subset R_i(F_1(x)). \end{aligned}$$

The second step follows from the definition of equivalence sequence. Therefore, it can be concluded that

$$xR_iy \implies F_1(x)R_{i+1}F_1(y) \text{ and} \tag{6.3}$$

$$\implies F_1(x)R_iF_1(y) \text{ for all } x \text{ and } y \in \Omega. \tag{6.4}$$

A similar argument holds for $F_{-1}(\cdot)$; therefore,

$$xR_iy \implies F_1(x)R_iF_1(y) \text{ and} \tag{6.5}$$

$$\implies F_{-1}(x)R_iF_{-1}(y) \quad \text{for all } x \text{ and } y \in \Omega.$$

Comparing with equation (6.1), we conclude that the equivalence relation R_i defines a partition that would result in a trellis. Using an equivalence relation at a given level will result in a power of two-state trellis. It should be mentioned that these trellises are the same as those found by Duel-Hallen, *et al.* [21] On the other hand, using our state partitioning technique, one can find trellises with any number of states.

6.2.2 Trellises with Number of States not 2^m

This is accomplished by considering partitions formed by subsets taken from adjacent levels.

To show that the partition so formed would result in a trellis, one has to satisfy two conditions:

$$xR_iy \implies F_1(x)R_{i+1}F_1(y) \text{ and}$$

$$\implies F_{-1}(x)R_{i+1}F_{-1}(y) \quad \text{for all } x \text{ and } y \in \Omega \tag{6.6}$$

and

$$xR_{i+1}y \implies F_1(x)R_iF_1(y) \text{ and}$$

$$\implies F_{-1}(x)R_iF_{-1}(y) \quad \text{for all } x \text{ and } y \in \Omega. \tag{6.7}$$

Condition (6.6) follows from equation (6.3). To prove condition (6.7); we note that since $R_{i+1}(x) \subset R_i(x)$, then

$$F_1(x)R_{i+1}F_1(y) \Longrightarrow F_1(x)R_iF_1(y).$$

Also from equation (6.4) we have

$$xR_{i+1}y \Longrightarrow F_1(x)R_{i+1}F_1(y).$$

Therefore, it follows that

$$xR_{i+1}y \Longrightarrow F_1(x)R_iF_1(y).$$

A similar argument holds for $F_{-1}(x)$. Therefore, the partition formed by considering subsets from two adjacent levels results in a trellis.

It is worth mentioning that subsets from non adjacent levels will not form a trellis since (6.6) will not be satisfied. In fact,

$$\begin{aligned} xR_iy &\Longrightarrow F_1(x)R_jF_1(y) \text{ and} \\ &\Longrightarrow F_{-1}(x)R_jF_{-1}(y) \quad \text{for all } x \text{ and } y \in \Omega \end{aligned}$$

is true only for $j = i + 1$.

6.3 An Example

Consider the channel given by

$$h(D) = h_0 + h_1D + h_2D^2 + h_3D^3. \tag{6.8}$$

The above channel has memory $n = 3$; therefore, the state can be represented by binary three tuples $x = (x_1, x_2, x_3)$. One can use up to three levels of partitioning, or the equivalence sequence $\{R_1, R_2, R_3\}$. We will consider the trellises formed by the equivalence sequence. Using the notation in section 6.2, the table below gives different partitioning schemes.

Table 6.1 Different Partitioning Schemes for Channel with Memory $n = 3$.

Number of States	Subsets
2	$R_1(0), R_1(1)$
3	$R_1(0), R_2(10), R_2(11)$
4	$R_2(00), R_2(01), R_2(10), R_2(11)$
5	$R_3(000), R_3(001), R_2(01)$ $R_2(10), R_2(11)$
6	$R_3(000), R_3(001), R_2(01)$ $R_2(10), R_3(110), R_3(111)$
7	$R_3(000), R_3(001), R_2(01)$ $R_3(100), R_3(101), R_3(110), R_3(111)$
8	$R_3(000), R_3(001), R_3(010), R_3(011)$ $R_3(100), R_3(101), R_3(110), R_3(111)$

The last entry in the table is the degenerate case of 8 states. The branch metric depends on the originating node of a given branch. If the originating node of a branch corresponds to a subset from level l ($l < n$), then the branch metric γ_k is given by

$$\gamma_k = (y_k - \sum_{i=0}^{l} h_i a_{k-i} - \sum_{i=l+1}^{n} h_i \hat{a}_{k-i})^2. \tag{6.9}$$

The trellises for the partitioning schemes considered in the above table are given below.

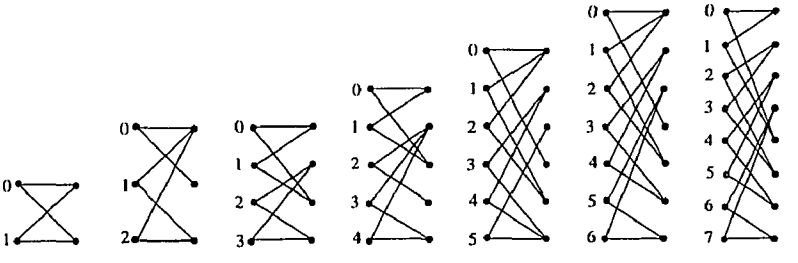

Figure 6.3 Trellises for the schemes given in Table 6.1

6.4 Probability of Error

In this section we will investigate the error performance of the partitioning schemes developed in section 6.2. It was noted in [20] [21] [55] that the effect of the error propagation is minimal for moderate to high SNR. Therefore, we will assume in our derivation that the effect of error propagation is negligible. In the sequel we will consider trellises with powers of two states, from which other trellises will be derived.

6.4.1 Trellises with 2^m States

These trellises are the same as those derived in [21], and hence the analysis given in [21] applies here. Nevertheless, we will relate the probability of error to different partitioning levels. This will be vital for the analysis of trellises with an arbitrary number of states. We will use Forney's approach [18].

Consider the trellis formed by the subsets from level i. As noted earlier, the resulting trellises will have 2^i states, which are represented by the binary i tuple. Define the state of the channel $s(k)$ at time k by

$$s(k) = (a_{k-1}, a_{k-2}, \cdots, a_{k-i}),$$

and denote the corresponding state estimated by the VA by $\hat{s}(k)$, where

$$\hat{s}(k) = (\hat{a}_{k-1}, \hat{a}_{k-2}, \cdots, \hat{a}_{k-i}).$$

The sequence $\{\hat{a}_{k-1}, \hat{a}_{k-2}, \cdots, \hat{a}_{k-i}\}$ is the sequence estimated by the VA. Following Forney's approach [18], an error event $\mathcal{E}$ is said to occur between $k = k_1$ and $k = k_2$, if $\hat{s}(k_1) = s(k_1)$, $\hat{s}(k_2) = s(k_2)$ and $\hat{s}(k) \neq s(k)$ for $k_1 < k < k_2$. Since $\hat{s}(k) = s(k)$ for $k = k_1, k_2$, it follows that

$$(\hat{a}_{k_1-1}, \hat{a}_{k_1-2}, \cdots, \hat{a}_{k_1-i}) = (a_{k_1-1}, a_{k_1-2}, \cdots, a_{k_1-i})$$

and

$$(\hat{a}_{k_2-1}, \hat{a}_{k_2-2}, \cdots, \hat{a}_{k_2-i}) = (a_{k_2-1}, a_{k_2-2}, \cdots, a_{k_2-i}).$$

Now define the input error sequence associated with the event $\mathcal{E}$, by

$$\mathbf{e} \triangleq \{e_{k_1}, e_{k_1+1}, \cdots, e_{k_2-i-1}\},$$

where $e_k \triangleq a_k - \hat{a}_k$. The Euclidean distance $d_i^2(\mathcal{E})$ of the error event is given by

$$d_i^2(\mathcal{E}) = \sum_{k=k_1}^{k_2} \left(\sum_{j=0}^{\min(k-k_1,i)} h_j e_{k-j} \right)^2. \tag{6.10}$$

In the case of binary transmission, the probability of error is upper bounded by [18]

$$P_e \leq \sum_{d \in D} Q\left(\frac{d_i}{2\sigma}\right) \sum_{\mathcal{E} \in E_{d_i}} w(\mathbf{e}) 2^{-w(\mathbf{e})},$$

where E_{d_i} is the set of all error events having a Euclidean distance of d_i^2 and D is the set of square roots of Euclidean distances attained by error events. The factor $w(\mathbf{e})$ is the number of bit errors a given error event entails, and $Q(\cdot)$ is given by

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-y^2/2} dy.$$

For moderate to high SNR the upper bound of the probability of error is dominated by events attaining the minimum distance, *i.e.*,

$$P_e \leq K_i Q\left(\frac{d_{i_{min}}}{2\sigma}\right), \tag{6.11}$$

where K_i is given by

$$K_i = \sum_{\mathcal{E} \in E_{d_{i_{min}}}} w(\mathbf{e}) 2^{-w(\mathbf{e})}. \tag{6.12}$$

Note that we used the subscript i throughout to emphasize the dependence of terms like $d_{i_{min}}$ and K_i on the level of partitioning i. Therefore, to evaluate the upper bound on the probability of error for a given level, one has to determine $d_{i_{min}}$ and K_i.

At lower SNR one can get better bounds by considering the stack algorithm given in [55]. However, with the stack algorithm one has to first find the error state diagram [55]. The complexity of such a diagram becomes intractable for channels with a long impulse response. Therefore, we will only consider events with minimum distances.

6.4.2 Trellises with Number of States not 2^m

We showed in section 6.2 how to form trellises by considering subsets from adjacent levels rather than from one level. Examples of these trellises were given in section 6.3.

To find an upper bound on the probability of error for these trellises, without loss of generality we consider the trellis formed by subsets from levels i and $i+1$. Further assume that the trellises are formed by considering p subsets from level i and q from level $i+1$ in such a way that the third constraint given in section 6.1 is satisfied. At moderate to high SNR, using the total probability theorem, the probability of error of such a scheme can be upper bounded by

$$P_e \leq \frac{p}{2^i} K_i Q\left(\frac{d_{i_{min}}}{2\sigma}\right) + \frac{q}{2^{i+1}} K_{i+1} Q\left(\frac{d_{i+1_{min}}}{2\sigma}\right). \tag{6.13}$$

It can be easily shown that for $q = 0$ the upper bound for level i results, while for $p = 0$ that of level $i+1$ results. Note that in equation (6.13) the first term dominates asymptotically, since $d_{i_{min}} < d_{i+1_{min}}$. That is, at high SNR the first term in equation (6.13) is more dominant than the second. Therefore, the performance of such a trellis would be the same as that of level i at sufficiently high SNR. However, at moderate SNR, the performance of these trellises is better than those with 2^i states, *i.e.*, trellises formed by considering subsets from level i only. This point is demonstrated in the following example. The improvement in performance becomes more pronounced for longer channels.

6.4.3 Simulation and Upper Bounds

As an example, consider the channel whose impulse response is given by

$$h(n) = 0.7107{\cdot}\delta(n)+0.1421{\cdot}\delta(n-1)+0.2132{\cdot}\delta(n-2)+0.1421{\cdot}\delta(n-3)+0.6396{\cdot}\delta(n-4). \tag{6.14}$$

The above channel has memory $n = 4$; therefore, the states can be represented by binary four tuples $x = (x_1, x_2, x_3, x_4)$. One can use up to four levels of partitioning,

or the equivalence sequence $\{R_1, R_2, R_3, R_4\}$. We will consider the trellises formed by the equivalence relations R_3 and R_4. Using the same notation as above, the table below gives some schemes we considered in the simulation and computation of the upper bounds.

Table 6.2 Selected Partitioning Schemes for a Channel with Memory $n = 4$

Number of States	Subsets
8	$R_3(000), R_3(001), R_3(010), R_3(011)$ $R_3(100), R_3(101), R_3(110), R_3(111)$
12	$R_3(000), R_3(001)$ $R_4(0100), R_4(0101), R_4(0110), R_4(0111)$ $R_4(1000), R_4(1001), R_4(1010), R_4(1011)$ $R_3(110), R_3(111)$
14	$R_3(000), R_4(0010), R_4(0011)$ $R_4(0100), R_4(0101), R_4(0110), R_4(0111)$ $R_4(1000), R_4(1001), R_4(1010), R_4(1011)$ $R_4(1100), R_4(1101), R_3(111)$
16	$R_4(0000), R_4(0001), R_4(0010), R_4(0011)$ $R_4(0100), R_4(0101), R_4(0110), R_4(0111)$ $R_4(1000), R_4(1001), R_4(1010), R_4(1011)$ $R_4(1100), R_4(1101), R_4(1110), R_4(1111)$

For level 4, the error sequences that have minimum distance are $\pm(2,0,0,0,0)$, and for level 3 these sequences are $\pm(2,0,0,0)$. The simulation results together with the upper bounds derived in the previous subsections are shown below.

Figure 6.4 shows that the upper bound is in agreement with the simulation results. For the example at hand, there is less than a 2dB loss when considering 8-state instead of the 16-state trellis. The improvement in the error performance obtained when using 12- and 14-state trellises over the 8-state trellis decreases with increasing SNR. That is, at moderate SNR the 12- and 14-state trellises have better performance over the 8-state, but at higher SNR the improvement will be insignificant.

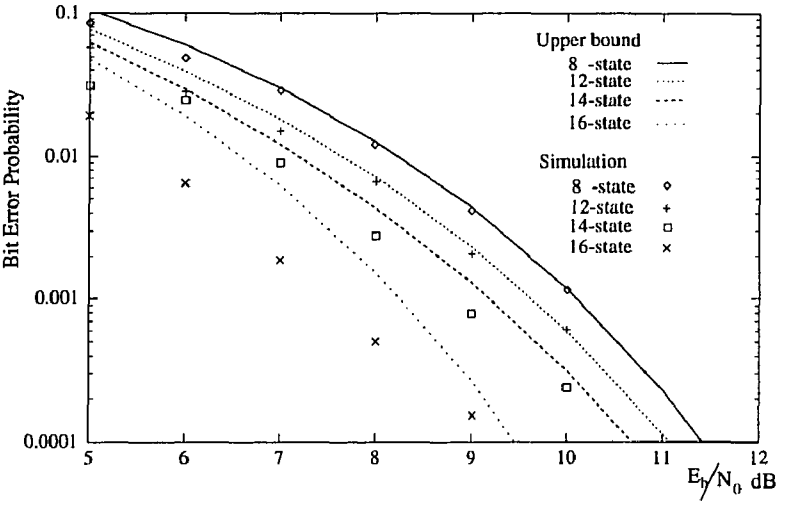

Figure 6.4 Probability of Error for Different Partitioning Schemes

6.5 Conclusion

In this chapter we introduced a new approach to reduce the complexity of the VA. This approach is based on partitioning the set of channel states. It also offers good complexity versus performance tradeoff. It was shown that the trellises obtained in [21] are special cases of those described in this chapter.

The state partitioning approach results in trellises with an arbitrary number of states which are not necessarily powers of 2 states, as in [21]. Depending on the length of the channel and the operating SNR, trellises with non-power of 2 states can attain a considerably lower probability of error than the powers of 2. However, at high SNR the improvement of the former over the latter is insignificant.

CHAPTER 7

CONCLUSIONS AND FUTURE DIRECTIONS

In this work we reported a number of new approaches to blind equalization. These varied from symbol-by-symbol detection to the sequence estimation.

In Chapter 2 we presented the decorrelation algorithm for decision feedback equalization and we showed convergence both analytically and through simulation. We also presented a rapidly converging version of the decorrelation algorithm. A natural extension would be to apply the decorrelation algorithm to the linear equalizer and study the effect of finite parameterization on convergence of the algorithm. In this work we assumed that the original source emits a white, noise-like sequence, *i.e.*, with zero correlation. An interesting point would be to investigate the effect of a non-white source on the decorrelation algorithm. A possible modification on the algorithm would be to match the output and input correlation. If such a match is achieved, correct convergence would be guaranteed [56].

In Chapter 3, we derived lower and upper bounds for the steady state probability of error. The lower bound was found to be tighter than the "no ISI" bound. We assumed perfect equalization for the probability of error calculation. A possible direction to follow is to relax such an assumption.

In Chapter 4, we introduced the concept of anchoring to the constant modulus algorithm. We showed that such an approach will improve the convergence of the algorithm. As a matter of fact, we showed that as long as the channel gain exceeds a certain critical value, the algorithm will be globally convergent. The anchored constant modulus algorithm was applied to the linear equalizer for autoregressive channels and decision feedback equalizers for moving average channels. An extension of the anchored constant modulus algorithm of Chapter 4, to include the linear

equalizer, is also a possible future step. The effect of anchoring on the finitely parameterized equalizer is a point worth investigating.

In Chapter 5 we introduced a scheme for blind Viterbi equalization, using a fixed step size c. It was found that for small values of c (less than 0.05), the probability of error of the blind Viterbi approaches the ideal one. The speed of convergence, on the other hand, was found to decrease with the value of c. A possible way to enhance the speed of convergence is to use a variable step size.

In Chapter 6, the concept of state partitioning was successfully applied in the binary case. We showed that this approach will generate trellises with an arbitrary number of states, not necessarly powers of two, which offers a better complexity/performance trade-off than other techniques [20][21]. Extending this to include non-binary and two-dimensional modulation schemes is essential. A possible path to follow might be to incorporate Ungerboeck-type partitioning [57] on the constellation level [20] and state partitioning on the channel level. Preliminary results showed the effectiveness of this method.

Equalization is one of the possible fields of application of reduced state sequence estimation. Applications to other fields should be addressed. These include decoding of convolutional and trellis codes and multi-user detection.

APPENDIX A

DERIVATION OF DENSITY FUNCTIONS

Claim 1 **The probability density function $f_{A_k}(\cdot)$ of the random variable A_k defined in equation (2.7) is an even function.**

Proof

The input to the slicer in equation (2.7) A_k is given by

$$\begin{aligned} A_k &= X_k - \sum_{i=1}^{N} w_i \hat{A}_{k-i} \\ &= I_k + \sum_{i=1}^{N} \left(h_i I_{k-i} - w_i \hat{A}_{k-i} \right). \end{aligned} \tag{A.1}$$

If we denote the set of all correct decisions by A' and the set of all incorrect decisions by A'', *i.e.*,

$$A' = \{\hat{A}_i : \hat{A}_i = I_i\}$$

$$A'' = \{\hat{A}_i : \hat{A}_i = -I_i\},$$

then the input of the slicer in (A.1) can be written as

$$A_k = I_k + \sum_{i:\hat{A}_{k-i} \epsilon A'} (h_i - w_i) I_{k-i} + \sum_{i:\hat{A}_{k-i} \epsilon A''} (h_i + w_i) I_{k-i}. \tag{A.2}$$

From the above equation one can see that A_k can be expressed as a sum of independent random variables. Therefore, the probability density function (pdf) of A_k is the convolution of the individual pdfs, thus,

$$f_{A_k} = f_{I_k} * \mathrm{Conv}_{i:\hat{A}_{k-i} \epsilon A'} f_{(h_i - w_i) I_{k-i}} * \mathrm{Conv}_{i:\hat{A}_{k-i} \epsilon A''} f_{(h_i + w_i) I_{k-i}}, \tag{A.3}$$

where $\mathrm{Conv}_{i:\hat{A}_{k-i} \epsilon A'}$ and $\mathrm{Conv}_{i:\hat{A}_{k-i} \epsilon A''}$ are the convolution of the probability density functions of the corresponding random variables in the summations of equation (A.3).

Since I_k's are random variables taking values of -1 and 1 with equal probabilities we have

$$\begin{aligned} f_{I_k}(x) &= \frac{1}{2}(\delta(x+1)+\delta(x-1)) \\ f_{(h_i-w_i)I_{k-i}}(x) &= \frac{1}{2}(\delta(x+h_i-w_i)+\delta(x-h_i+w_i)) \\ f_{(h_i+w_i)I_{k-i}}(x) &= \frac{1}{2}(\delta(x+h_i+w_i)+\delta(x-h_i-w_i)). \end{aligned}$$

The convolution equation in (A.3) can be transformed into a product form by using the Fourier Transform

$$\mathcal{F}_{A_k} = \mathcal{F}_{I_k} \cdot \prod_{i:\hat{A}_{k-i}\epsilon A'} \mathcal{F}_{(h_i-w_i)I_{k-i}} \cdot \prod_{i:\hat{A}_{k-i}\epsilon A''} \mathcal{F}_{(h_i+w_i)I_{k-i}}, \tag{A.4}$$

where $\mathcal{F}_X$ is the Fourier Transform of the pdf of the random variable X. Therefore, we have

$$\begin{aligned} \mathcal{F}_{I_k}(\omega) &= \cos(\omega) \\ \mathcal{F}_{(h_i-w_i)I_{k-i}}(\omega) &= \cos((h_i-w_i)\omega) \\ \mathcal{F}_{(h_i+w_i)I_{k-i}}(\omega) &= \cos((h_i+w_i)\omega). \end{aligned}$$

Now we consider the product terms in equation (A.4). The first term,

$$\prod_{i:\hat{A}_{k-i}\epsilon A'} \mathcal{F}_{(h_i-w_i)I_{k-i}}(\omega) = \frac{1}{2^{|A'|-1}} \sum_{a_i} \cos(a_i\omega),$$

where $|A'|$ is the cardinality of the set A' and a_i's represent all the possible sums and differences among all $(h_i - w_i)$ such that $\hat{A}_{k-i}\epsilon A'$.

Similarly, for the other product term in equation (A.3) one can write,

$$\prod_{i:\hat{A}_{k-i}\epsilon A''} \mathcal{F}_{(h_i+w_i)I_{k-i}}(\omega) = \frac{1}{2^{|A''|-1}} \sum_{b_i} \cos(b_i\omega),$$

where $|A''|$ is the cardinality of the set A'' and b_i's represent all the possible sums and differences among all $(h_i + w_i)$ such that $\hat{A}_{k-i}\epsilon A''$.

As a result equation (A.4) can be written as

$$\begin{aligned}
\mathcal{F}_{A_k}(\omega) &= \cos(\omega) \cdot \frac{1}{2^{|A'|-1}} \sum_{a_i} \cos(a_i\omega) \cdot \frac{1}{2^{|A''|-1}} \sum_{b_i} \cos(b_i\omega) \\
&= \cos(\omega) \cdot \frac{1}{2^{N-2}} \sum_{a_i} \sum_{b_i} \cos(a_i\omega)\cos(b_i\omega) \text{ since } |A'| + |A''| = N \\
&= \cos(\omega) \cdot \frac{1}{2^{N-1}} \sum_{a_i} \sum_{b_i} \cos((a_i + b_i)\omega) + \cos((a_i - b_i)\omega) \\
&= \cos(\omega) \cdot \frac{1}{2^{N-1}} \sum_{c_i} \cos(c_i\omega),
\end{aligned}$$

where c_i represents all the possible pairwise sums and differences of a_is and b_is. Further, one can write

$$\mathcal{F}_{A_k} = \frac{1}{2^N} \sum_{c_i} \cos((c_i + 1)\omega) + \cos((c_i - 1)\omega). \tag{A.5}$$

Taking the inverse transform of equation (A.5), we can write the pdf of A_k as

$$f_{A_k}(x) = \frac{1}{2^{N+1}} \sum_{c_i} \left(\delta(x - c_i - 1) + \delta(x + c_i + 1) + \delta(x - c_i + 1) + \delta(x + c_i - 1)\right). \tag{A.6}$$

Therefore, the pdf of A_k is an even function, and it also exhibits half symmetry about ± 1.

Claim 2

$$\mathbf{E\{A_{k-m}\hat{A}_{k-n}\} = 0 \quad for \quad m > n}$$

Proof:

Consider the joint cumulative distribution function (CDF) of $A_{k-m}\hat{A}_{k-n}$ *viz* $F_{A_{k-m}\hat{A}_{k-n}}(x, y)$,

$$\begin{aligned}
F_{A_{k-m}\hat{A}_{k-n}}(x, y) &= P\{A_{k-m} \le x, \hat{A}_{k-n} \le y\} \\
&= P\{A_{k-m} \le x, \hat{A}_{k-n} \le y \mid \hat{A}_{k-n} = I_{k-n}\}p \\
&\quad + P\{A_{k-m} \le x, \hat{A}_{k-n} \le y \mid \hat{A}_{k-n} = -I_{k-n}\}q \\
&= P\{A_{k-m} \le x, I_{k-n} \le y \mid \hat{A}_{k-n} = I_{k-n}\}p \\
&\quad + P\{A_{k-m} \le x, -I_{k-n} \le y \mid \hat{A}_{k-n} = -I_{k-n}\}q,
\end{aligned}$$

where p is the probability of a correct decision and q is the probability of an incorrect decision.

$$\begin{aligned} F_{A_{k-m}\hat{A}_{k-n}}(x,y) &= P\{A_{k-m} \le x \mid \hat{A}_{k-n} = I_{k-n}\}P\{I_{k-n} \le y \mid \hat{A}_{k-n} = I_{k-n}\}p \\ &\quad + P\{A_{k-m} \le x \mid \hat{A}_{k-n} = -I_{k-n}\}P\{-I_{k-n} \le y \mid \hat{A}_{k-n} = -I_{k-n}\}q \\ &= P\{A_{k-m} \le x, \hat{A}_{k-n} = I_{k-n}\}P\{I_{k-n} \le y \mid \hat{A}_{k-n} = I_{k-n}\} \\ &\quad + P\{A_{k-m} \le x, \hat{A}_{k-n} = -I_{k-n}\}P\{-I_{k-n} \le y \mid \hat{A}_{k-n} = -I_{k-n}\} \end{aligned} \tag{A.7}$$

since A_{k-m} is independent of I_{k-n} for $m > n$. By definition

$$\begin{aligned} P\{I_{k-n} \le y \mid \hat{A}_{k-n} = I_{k-n}\} &= \int_{-\infty}^{y} \delta(\mu - 1)P\{I_{k-n} = 1 \mid \hat{A}_{k-n} = I_{k-n}\} \\ &\quad + \delta(\mu + 1)P\{I_{k-n} = -1 \mid \hat{A}_{k-n} = I_{k-n}\}d\mu. \end{aligned} \tag{A.8}$$

Now, from equation (2.7) we write

$$A_{k-n} = I_{k-n} + Y_{k-n},$$

where

$$Y_{k-n} \triangleq \sum_{i=1}^{N} \left(h_i I_{k-n-i} - w_i \hat{A}_{k-n-i}\right).$$

From the definition of $\hat{A}_k$,

$$\begin{aligned} P\{\hat{A}_{k-n} = I_{k-n} \mid I_{k-n} = 1) &= P\{\text{sgn}(A_{k-n}) = I_{k-n} \mid I_{k-n} = 1\} \\ &= P\{\text{sgn}(1 + Y_{k-n}) = 1) \\ &= P\{Y_{k-n} > -1). \end{aligned}$$

Similarly,

$$P\{\hat{A}_{k-n} = I_{k-n} \mid I_{k-n} = -1\} = P\{Y_{k-n} < 1\}.$$

However from Claim 1 of this appendix the pdf of A_k and, hence, of A_{k-n} and Y_{k-n} is even. This leads to

$$P\{\hat{A}_{k-n} = I_{k-n} \mid I_{k-n} = 1\} = P\{\hat{A}_{k-n} = I_{k-n} \mid I_{k-n} = -1\}. \tag{A.9}$$

Now, using Bayes' law,

$$P\{I_{k-n}=1 \mid \hat{A}_{k-n}=I_{k-n}\} = \frac{P\{\hat{A}_{k-n}=I_{k-n} \mid I_{k-n}=1\}P\{I_{k-n}=1\}}{P\{\hat{A}_{k-n}=I_{k-n}}$$

$$P\{I_{k-n}=-1 \mid \hat{A}_{k-n}=I_{k-n}\} = \frac{P\{\hat{A}_{k-n}=I_{k-n} \mid I_{k-n}=-1\}P\{I_{k-n}=1\}}{P\{\hat{A}_{k-n}=I_{k-n}}.$$

Therefore, by using equation (A.9) we get

$$P\{I_{k-n}=1 \mid \hat{A}_{k-n}=I_{k-n}\} = P\{I_{k-n}=-1 \mid \hat{A}_{k-n}=I_{k-n}\} = \frac{1}{2}.$$

Hence, we can write

$$\begin{aligned} P\{I_{k-n} \le y \mid \hat{A}_{k-n}=I_{k-n}\} &= \frac{1}{2}\int_{-\infty}^{y}(\delta(\mu-1)+\delta(\mu+1))\,d\mu \\ &= F_{I_{k-n}}(y). \end{aligned} \tag{A.10}$$

Similarly, it can be shown that

$$\begin{aligned} P\{-I_{k-n} \le y \mid \hat{A}_{k-n}=-I_{k-n}\} &= \frac{1}{2}\int_{-\infty}^{y}(\delta(\mu-1)+\delta(\mu+1))\,d\mu \\ &= F_{I_{k-n}}(y). \end{aligned} \tag{A.11}$$

Substituting equations (A.10) and (A.11) in (A.7), we get

$$\begin{aligned} F_{A_{k-m}\hat{A}_{k-n}}(x,y) &= P\{A_{k-m} \le x, \hat{A}_{k-n}=I_{k-n}\}F_{I_{k-n}}(y) \\ &\quad + P\{A_{k-m} \le x\hat{A}_{k-n}=-I_{k-n}\}F_{I_{k-n}}(y) \\ &= F_{A_{k-m}}(x)F_{I_{k-n}}(y). \end{aligned}$$

Therefore, the joint pdf of $A_{k-m}\hat{A}_{k-n}$ is given by

$$f_{A_{k-m}\hat{A}_{k-n}} = f_{A_{k-m}}(x)f_{I_{k-n}}(y).$$

Hence,

$$\begin{aligned} E\{A_{k-m}\hat{A}_{k-n}\} &= E\{A_{k-m}\}E\{I_{k-n}\} \\ &= 0. \end{aligned}$$

Claim 3

$$\mathbf{E\{I_{k-m}\hat{A}_{k-n}\} = 0 \quad n \neq m}$$

Proof:

We have

$$\begin{aligned} F_{\hat{A}_{k-n}I_{k-m}}(x,y) &= P\{\hat{A}_{k-n} \leq x, I_{k-m} \leq y\} \\ &= P\{I_{k-m} \leq y, \hat{A}_{k-n} \leq x \mid \hat{A}_{k-n} = I_{k-n}\}P\{\hat{A}_{k-n} = I_{k-n}\} \\ &\quad + P\{I_{k-m} \leq y, \hat{A}_{k-n} \leq x \mid \hat{A}_{k-n} = -I_{k-n}\}P\{\hat{A}_{k-n} = -I_{k-n}\} \\ &= P\{I_{k-m} \leq y, I_{k-n} \leq x \mid \hat{A}_{k-n} = I_{k-n}\}p_{k-n} \\ &\quad + P\{I_{k-m} \leq y, I_{k-n} \leq x \mid \hat{A}_{k-n} = -I_{k-n}\}q_{k-n}. \end{aligned}$$

I_{k-m} independent of I_{k-n} for $m \neq n$, hence

$$\begin{aligned} F_{\hat{A}_{k-n}I_{k-m}}(x,y) &= P\{I_{k-m} \leq y \mid \hat{A}_{k-n} = I_{k-n}\}P\{I_{k-n} \leq x \mid \hat{A}_{k-n} = I_{k-n}\}p_{k-n} \\ &\quad + P\{I_{k-m} \leq y \mid \hat{A}_{k-n} = -I_{k-n}\}P\{-I_{k-n} \leq x \mid \hat{A}_{k-n} = -I_{k-n}\}q_{k-n}. \end{aligned}$$

Using equations (A.9) and (A.10), we get

$$\begin{aligned} F_{\hat{A}_{k-n}I_{k-m}}(x,y) &= P\{I_{k-m} \leq y, \hat{A}_{k-n} = I_{k-n-1}\}F_{I_{k-n-1}}(x) \\ &\quad + P\{I_{k-m} \leq y, \hat{A}_{k-n} = -I_{k-n-1}\}F_{I_{k-n-1}}(x) \\ &= F_{I_{k-m}}(y)F_{I_{k-n}}(x). \end{aligned} \tag{A.12}$$

Therefore, from equation (A.12), we can conclude that

$$\begin{aligned} E\{\hat{A}_{k-n}I_{k-m}\} &= E\{I_{k-m}\overline{I_{k-n}}\}. \\ &= 0 \end{aligned} \tag{A.13}$$

Claim 4

$$\mathbf{E\{|A_k|\} = 1 \quad for\ every\ n} \tag{A.14}$$

Proof:

The pdf $f_{|A_k|}$ of the random variable $|A_k|$ can be expressed as

$$\begin{aligned} f_{|A_k|}(x) &= \begin{cases} f_{A_k}(x) + f_{A_k}(-x) & x \geq 0 \\ 0 & x < 0 \end{cases} \\ &= \begin{cases} 2f_{A_k}(x) & x \geq 0 \\ 0 & x < 0 \end{cases} \end{aligned}$$

since $f_{A_k}(x)$ is an even function.

Substituting from equation (A.6)

$$f_{|A_k|} = \begin{cases} \frac{1}{2^{N+1}} \sum_{c_i} \left(\delta(x - |1 + c_i|) + \delta(x - |1 - c_i|)\right) & x \geq 0 \\ 0 & x < 0. \end{cases}$$

The above equation is symmetric about $x = 1$, therefore the mean

$$E\{|A_k|\} = 1.$$

Claim 5

$$\textbf{For } \mathbf{m \neq n} \quad \overline{\hat{\mathbf{A}}_{\mathbf{k-m}} \hat{\mathbf{A}}_{\mathbf{k-n}}} = \mathbf{0}$$

$$\mathbf{m = n} \quad \overline{\hat{\mathbf{A}}^{\mathbf{2}}_{\mathbf{k-n}}} = \mathbf{1}$$

Proof:

Assume $m < n$, then

$$\begin{aligned} F_{\hat{A}_{k-m}\hat{A}_{k-n}} &= P\{\hat{A}_{k-n} \leq x, \hat{A}_{k-m} \leq y\} \\ &= P\{\hat{A}_{k-n} \leq x, \hat{A}_{k-m} \leq y \mid \hat{A}_{k-m} = I_{k-m}\} p_{k-m} \\ &\quad + P\{\hat{A}_{k-n} \leq x, \hat{A}_{k-m} \leq y \mid \hat{A}_{k-m} = -I_{k-m}\} q_{k-m} \\ &= P\{\hat{A}_{k-n} \leq x, I_{k-m} \leq y \mid \hat{A}_{k-m} = I_{k-m}\} p_{k-m} \\ &\quad + P\{\hat{A}_{k-n} \leq x, -I_{k-m} \leq y \mid \hat{A}_{k-m} = -I_{k-m}\} q_{k-m}. \end{aligned}$$

$\hat{A}_{k-n}$ depends only on I_{k-m} for $m \geq n$, hence, it is independent of all I_{k-m} with $m < n$. Therefore,

$$\begin{aligned} F_{\hat{A}_{k-m}\hat{A}_{k-n}} &= P\{\hat{A}_{k-n} \leq x \mid \hat{A}_{k-m} = I_{k-m}\}P\{\hat{I}_{k-m-1} \leq y \mid \hat{A}_{k-m} = I_{k-m}\}p_{k-m} \\ &\quad + P\{\hat{A}_{k-n} \leq x, \mid \hat{A}_{k-m} = -I_{k-m}\}P\{-\hat{I}_{k-m} \leq y, \mid \hat{A}_{k-m} = -I_{k-m}\}q_{k-m} \\ &= P\{\hat{A}_{k-n} \leq x, \hat{A}_{k-m} = I_{k-m}\}P\{\hat{I}_{k-m} \leq y \mid \hat{A}_{k-m} = I_{k-m}\} \\ &\quad + P\{\hat{A}_{k-n} \leq x, \hat{A}_{k-m} = -I_{k-m}\}P\{-\hat{I}_{k-m} \leq y, \mid \hat{A}_{k-m} = -I_{k-m}\}. \end{aligned}$$

By using equations (A.9) and (A.10), we get

$$\begin{aligned} F_{\hat{A}_{k-m}\hat{A}_{k-n}} &= P\{\hat{A}_{k-n} \leq x, \hat{A}_{k-m} = I_{k-m}\}F_{I_{k-m}}(y) \\ &\quad + P\{\hat{A}_{k-n} \leq x, \hat{A}_{k-m} = -I_{k-m}\}F_{I_{k-m}}(y) \\ &= F_{\hat{A}_{k-n}}(x)F_{I_{k-m}}(y). \end{aligned} \tag{A.15}$$

Therefore,

$$\begin{aligned} \overline{\hat{A}_{k-m}\hat{A}_{k-n}} &= \overline{\hat{A}_{k-n}}\,\overline{I_{k-m-1}} \\ &= 0. \end{aligned} \tag{A.16}$$

For $m > n$, a similar proof can be shown by conditioning on $\hat{A}_{k-n}$ instead.

For $m = n$, since the pdf of A_{k-n} is even, it follows that

$$P\{\hat{A}_{k-n} = 1\} = P\{\hat{A}_{k-n} = -1\} = \frac{1}{2}.$$

From the above, it is straightforward to show that, for $m = n$,

$$\overline{\hat{A}^2_{k-n}} = 1.$$

Claim 6

For $m + 1 < i < N$

$$\begin{aligned} \overline{\mathbf{A_{k-m}I_{k-i}}} &= \mathbf{h_{i-m} - w_{i-m}\,(1 - 2q_{k-i})} \\ \overline{\mathbf{A_{k-m}\hat{A}_{k-i}}} &= \mathbf{h_{i-m}\,(1 - 2q_{k-i}) - w_{i-m}} \end{aligned}$$

where q_{k-i} is the probability that $\hat{A}_{k-i} \neq I_{k-i}$.

Proof:

Note that

$$\overline{A_{k-m}e_{k-i}} = \overline{A_{k-m}I_{k-i}} - \overline{A_{k-m}\hat{A}_{k-i}}.$$

From equation (2.7),

$$A_{k-m} = I_{k-m} + \sum_{j=1}^{N} h_j I_{k-m-j} - \sum_{j=1}^{N} w_j \hat{A}_{k-m-j}.$$

We consider each term separately:

$$\overline{A_{k-m}I_{k-i}} = \overline{I_{k-m}I_{k-i}} + \sum_{j=1}^{N} h_j \overline{I_{k-m-j}I_{k-i}} - \sum_{j=1}^{N} w_j \overline{\hat{A}_{k-m-j}I_{k-i}}.$$

The first term in the RHS is zero, since $i > m$. Similarly, the summation in the second term is $j = i - m$. Using the result in Claim 3 the terms in the second summation are all zero except for $j = i - m$. Therefore,

$$\begin{aligned} \overline{A_{k-m}I_{k-i}} &= h_{i-m} - w_{i-m}\overline{\hat{A}_{k-i}I_{k-i}}, \\ &= h_{i-m} - w_{i-m}(1 - 2q_{k-i}) \end{aligned} \tag{A.17}$$

Now, consider

$$\overline{A_{k-m}\hat{A}_{k-i}} = \overline{I_{k-m}\hat{A}_{k-i}} + \sum_{j=1}^{N} h_j \overline{I_{k-m-j}\hat{A}_{k-i}} - \sum_{j=1}^{N} w_j \overline{\hat{A}_{k-m-j}\hat{A}_{k-i}}.$$

Using Claim 3, the first term is zero. Furthermore, using the same claim, the only non-zero term in the first summation is $j = i - m$. On the other hand, using Claim 5 the only non-zero term in the second sum is $j = i - m$.

$$\begin{aligned} \overline{\hat{A}_{k-m}\hat{A}_{k-i}} &= h_{i-m}\overline{\hat{A}_{k-i}I_{k-i}} - w_{i-m}\overline{\hat{A}_{k-i}^2} \\ &= h_{i-m}\overline{\hat{A}_{k-i}I_{k-i}} - w_{i-m} \\ &= h_{i-m}(1 - 2q_{k-i}) - w_{i-m} \end{aligned} \tag{A.18}$$

Claim 7

$$\begin{aligned} & \mathbf{P}\{\left(-\mathbf{1}+(\mathbf{h_1}+\mathbf{w_1^{(k)}})\mathbf{I_{k-1}}+(\mathbf{h_2}+\mathbf{w_2^{(k)}})\mathbf{I_{k-2}}\right)>\mathbf{0}\} \\ = & \frac{\mathbf{1}}{\mathbf{2}}\left(\mathbf{P}\{\mathbf{I_{k-2}}>\frac{\mathbf{1}-(\mathbf{h_2}+\mathbf{w_2^{(k)}})}{|\mathbf{h_1}+\mathbf{w_1^{(k)}}|}\}+\mathbf{P}\{\mathbf{I_{k-1}}>\frac{\mathbf{1}+(\mathbf{h_2}+\mathbf{w_2^{(k)}})}{|\mathbf{h_1}+\mathbf{w_1^{(k)}}|}\right) \quad \text{(A.19)} \end{aligned}$$

Proof:

Define $\mathcal{P}$ as

$$\mathcal{P} \triangleq P\{\left(-1+(h_1+w_1^{(k)})I_{k-1}+(h_2+w_2^{(k)})I_{k-2}\right)>0\}.$$

Then

$$\begin{aligned} \mathcal{P} = & \frac{1}{2}\left(P\{\left(-1+(h_1+w_1^{(k)})I_{k-1}+(h_2+w_2^{(k)})\right)>0\}\right. \\ & \left.+P\{\left(-1+(h_1+w_1^{(k)})I_{k-1}-(h_2+w_2^{(k)})\right)>0\}\right). \end{aligned}$$

If $(h_1+w_1^{(k)})>0$, then

$$\mathcal{P}=\frac{1}{2}\left(P\{I_{k-1}>\frac{1-(h_2+w_2^{(k)}}{h_1+w_1^{(k)}}\}+(P\{I_{k-1}>\frac{1-(h_2+w_2^{(k)}}{h_1+w_1^{(k)}}\}\right).$$

If, on the other hand, $(h_1+w_1^{(k)})<0$, then

$$\begin{aligned} \mathcal{P} = & \frac{1}{2}\left(P\{I_{k-1}<\frac{1-(h_2+w_2^{(k)}}{h_1+w_1^{(k)}}\}+(P\{I_{k-1}<\frac{1-(h_2+w_2^{(k)}}{h_1+w_1^{(k)}}\}\right) \\ = & \frac{1}{2}\left(P\{I_{k-1}>-\frac{1-(h_2+w_2^{(k)}}{h_1+w_1^{(k)}}\}+(P\{I_{k-1}>-\frac{1-(h_2+w_2^{(k)}}{h_1+w_1^{(k)}}\}\right). \end{aligned}$$

The last step follows since the pdf of I_{k-1} is an even function. Therefore, combining the above

$$\begin{aligned} & P\{\left(-1+(h_1+w_1^{(k)})I_{k-1}+(h_2+w_2^{(k)})I_{k-2}\right)>0\} \\ = & \frac{1}{2}\left(P\{I_{k-1}>\frac{1-(h_2+w_2^{(k)})}{|h_1+w_1^{(k)}|}\}+P\{I_{k-1}>\frac{1+(h_2+w_2^{(k)})}{|h_1+w_1^{(k)}|}\right). \end{aligned}$$

APPENDIX B

DERIVATION OF TRANSITION PROBABILITIES

B.1 Derivation of Equation (3.14)

From equation (3.13),

$$P\{\hat{A}_k = 1 | I_k = 1\} = \sum_{y(N,k)} P\left\{ I_{k-1} > -\frac{1 + y(N,k)}{|h_1 - w_1^{(k)}|} | Y = y(N,k) \right\} P\{Y = y(N,k)\}.$$

Since Y is independent of I_{k-1}, we get

$$\begin{aligned} P\{\hat{A}_k = 1 | I_k = 1\} &= \sum_{y(N,k)} P\left\{ I_{k-1} > -\frac{1 + y(N,k)}{|h_1 - w_1^{(k)}|} \right\} P\{Y = y(N,k)\} \\ P\{\hat{A}_k = 1 | I_k = 1\} &\geq P\left\{ I_{k-1} > -\frac{1 - \sum_{i=2}^{N} |h_i - w_i^{(k)}|}{|h_1 - w_1^{(k)}|} \right\}. \end{aligned}$$

It can be shown that

$$P\{\hat{A}_k = 1 | I_k = 1\} = P\{\hat{A}_k = -1 | I_k = -1\}.$$

Therefore, we get

$$\alpha_N \geq P\left\{ I_{k-1} > -\frac{1 - \sum_{i=2}^{N} |h_i - w_i^{(k)}|}{|h_1 - w_1^{(k)}|} \right\}. \tag{B.1}$$

B.2 Derivation of Equation (3.16)

From equation (3.15), by conditioning on Y_1, Y_2, Y_3 and Y_4,

$$\begin{aligned} & P\{\hat{A}_k = 1 | I_k = 1\} \\ = & \sum_{\substack{y_1(m,k), y_2(m,k) \\ y_3(m,k), y_4(m,k)}} P\left\{ I_{k-1} > -\frac{1 + y_1(m,k) + y_2(m,k) + y_3(m,k) - y_4(m.k)}{|h_1 - w_1^{(k)}|} \right. \\ & \left. | \, Y_1, Y_2, Y_3, Y_4 \right\} \cdot \\ & P\{Y_1 = y_1(m,k), Y_2 = y_2(m,k), Y_3 = y_3(m,k), Y_4 = y_4(m.k)\} \\ = & \sum_{\substack{y_1(m,k), y_2(m,k) \\ y_3(m,k), y_4(m,k)}} P\left\{ I_{k-1} > -\frac{1 + y_1(m,k) + y_2(m,k) + y_3(m,k) - y_4(m,k)}{|h_1 - w_1^{(k)}|} \right\} \cdot \\ & P\{Y_1 = y_1(m,k), Y_2 = y_2(m,k), Y_3 = y_3(m,k), Y_4 = y_4(m,k)\} \end{aligned}$$

$$
\begin{aligned}
& P\{\hat{A}_k = 1 | I_k = 1\} \\
\geq\ & P\left\{ I_{k-1} > -\frac{1 - \sum_{i=2}^{m} |h_i - w_i^{(k)}| - |h_{m+1} + w_{m+1}^{(k)}| - \sum_{i=m+2}^{N} |h_i| - \sum_{i=m+2}^{N} |w_i^{(k)}|}{|h_1 - w_1^{(k)}|} \right\}.
\end{aligned}
$$

Since

$$
P\{\hat{A}_k = 1 | I_k = 1\} = P\{\hat{A}_k = -1 | I_k = -1\}.
$$

Therefore, one can write

$$
\begin{aligned}
& \alpha_m \geq \\
& P\left\{ I_{k-1} > -\frac{1 - \sum_{i=2}^{m} |h_i - w_i^{(k)}| - |h_{m+1} + w_{m+1}^{(k)}| - \sum_{i=m+2}^{N} |h_i| - \sum_{i=m+2}^{N} |w_i^{(k)}|}{|h_1 - w_1^{(k)}|} \right\}.
\end{aligned} \tag{B.2}
$$

B.3 Derivation of Equation (3.22)

From equation (3.22)

$$
\begin{aligned}
P\{\hat{A}_k = 1 | I_k = 1\} &= P\{1 + h_N E_{k-N} + n_k > 0\} \\
&= P\{n_k > -1 - 2h_N\} P\{E_{k-N} = 2\} \\
&\quad + P\{n_k > -1 + 2h_N\} P\{E_{k-N} = -2\} \\
&= Q\left(\frac{-1 - 2h_N}{\sigma}\right) P\{E_{k-N} = 2\} + Q\left(\frac{-1 + 2h_N}{\sigma}\right) P\{E_{k-N} = -2\}.
\end{aligned}
$$

and

$$
\begin{aligned}
P\{\hat{A}_k = -1 | I_k = -1\} &= P\{-1 + h_N E_{k-N} + n_k < 0\} \\
&= P\{n_k < 1 - 2h_N\} P\{E_{k-N} = 2\} \\
&\quad + P\{n_k < 1 + 2h_N\} P\{E_{k-N} = -2\} \\
&= Q\left(\frac{-1 + 2h_N}{\sigma}\right) P\{E_{k-N} = 2\} \\
&\quad + Q\left(\frac{-1 - 2h_N}{\sigma}\right) P\{E_{k-N} = -2\}.
\end{aligned}
$$

Combining the above two equations, we get

$$
\alpha_{N-1} = \frac{1}{2}\left(Q\left(\frac{-1 - 2h_N}{\sigma}\right) + Q\left(\frac{-1 + 2h_N}{\sigma}\right)\right). \tag{B.3}
$$

B.4 Derivation of Eqs. (3.24) and (3.25)

From equation (3.23) we have,

$$P\{\hat{A}_k = 1 | I_k = 1\} = P\{1 + h_{m+1}E_{k-m-1} + Y_{m+1} + n_k > 0\}.$$

Using the total probability theorem the above probability is evaluated in terms of the probabilities of E_{k-m-1} and Y_{m+1}. Therefore,

$$\begin{aligned} & P\{\hat{A}_k = 1 | I_{k-1} = 1\} \\ = & \sum_{y_{m+1}} Q\left(\frac{-1 - 2h_{m+1} - y_{m+1}}{\sigma}\right) \cdot P\{Y_{m+1} = y_{m+1} | E_{k-m-1} = 2\} P\{E_{k-m-1} = 2\} \\ + & \sum_{y_{m+1}} Q\left(\frac{-1 + 2h_{m+1} - y_{m+1}}{\sigma}\right) \cdot P\{Y_{m+1} = y_{m+1} | E_{k-m-1} = -2\}.P\{E_{k-m-1} = -2\} \end{aligned}$$

Define β_m as

$$\beta_m = 2 \sum_{i=m+1}^{N} |h_i|$$

Using the above we can bound the transition probability. It can be shown that

$$\begin{aligned} P\{\hat{A}_k = 1 | I_k = 1\} \geq & \; Q\left(\frac{-1 - 2h_{m+1} + \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = 2\} \\ & + Q\left(\frac{-1 + 2h_{m+1} + \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = -2\}, \quad \text{(B.4)} \end{aligned}$$

and

$$\begin{aligned} P\{\hat{A}_k = 1 | I_k = 1\} \leq & \; Q\left(\frac{-1 - 2h_{m+1} - \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = 2\} \\ & + Q\left(\frac{-1 + 2h_{m+1} - \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = -2\}, \quad \text{(B.5)} \end{aligned}$$

Similarly, it can be shown that

$$\begin{aligned} P\{\hat{A}_k = -1 | I_k = -1\} \geq & \; Q\left(\frac{-1 - 2h_{m+1} + \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = -2\} \\ & + Q\left(\frac{-1 + 2h_{m+1} + \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = 2\}, \quad \text{(B.6)} \end{aligned}$$

and

$$\begin{aligned} P\{\hat{A}_k = -1 | I_k = -1\} \leq & \; Q\left(\frac{-1 - 2h_{m+1} - \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = -2\} \\ & + Q\left(\frac{-1 + 2h_{m+1} - \beta_{m+1}}{\sigma}\right) P\{E_{k-m-1} = 2\}. \quad \text{(B.7)} \end{aligned}$$

From equations (B.4) and (B.6) we can find a lower bound for α_m:

$$\alpha_m \geq \frac{1}{2}\left(Q\left(\frac{-1-2h_{m+1}+\beta_{m+1}}{\sigma}\right)+Q\left(\frac{-1+2h_{m+1}+\beta_{m+1}}{\sigma}\right)\right). \tag{B.8}$$

Similarly, from equations (B.5) and (B.7) an upper bound on α_m can be obtained

$$\alpha_m \leq \frac{1}{2}\left(Q\left(\frac{-1-2h_{m+1}-\beta_{m+1}}{\sigma}\right)+Q\left(\frac{-1+2h_{m+1}-\beta_{m+1}}{\sigma}\right)\right), \tag{B.9}$$

where

$$\beta_m = 2\sum_{i=m+1}^{N}|h_i|. \tag{B.10}$$

BIBLIOGRAPHY

1. J.G. Proakis, *Digital Communications*, second edition, McGraw Hill, New York, 1989.

2. R.W. Lucky, "Techniques for Adaptive Equalization of Digital Communication Systems," *Bell System Technical Journal,* Vol. 45, pp. 255-286, February 1966.

3. S.U.H. Qureshi, "Adaptive Equalization," *Proc. of the IEEE*, 73, pp. 1349-1387, September 1985.

4. J.G. Proakis, "Adaptive Equalization for TDMA Digital Mobile Radio," *IEEE Trans. on Vehicular Technology,* VT-40, pp. 333-341, May 1991

5. R.D. Gitlin, J.E. Mazo, and M.G. Taylor, "On the Design of Gradient Algorithms for Digitally Implemented Adaptive Filters," *IEEE Trans. on Circuit Theory,* CT-20, pp. 125-136, March 1973.

6. D. Godard, "Channel Equalization Using a Kalman Filter for Fast Data Transmission," *IBM Journal of Research and Development,* pp. 267-273, May 1974.

7. R.D. Gitlin and F.R. Magee, "Self-Orthogonalizing Adaptive Equalization Algorithms," *IEEE Trans. on Communications,* COM-25 pp. 666-672, July 1977.

8. G.J. Bierman, *Factorization Methods for Discrete Sequential Estimation,* Academic, New York, 1977.

9. D.D. Falconer and L. Ljung, "Application of Fast Kalman Estimation to Adaptive Equalization," *IEEE Trans. on Communications,* COM-26 pp. 1439-1445, October 1978.

10. B. Friedlander, "Lattice Filters for Adaptive Processing," *Proc. of the IEEE,* Vol. 70, pp. 829-867, August 1982.

11. F. Ling, and J.G. Proakis, "Generalized Least Squares Lattice and Its Application to DFE," *Proc. 1982 IEEE ICASSP* Paris, France, May 1982.

12. E.H. Satoris and J.D. Pack, "Application of Least Squares Lattice Algorithms to Adaptive Equalization," *IEEE Trans. on Communications,* COM-29, pp. 136-142, February 1981.

13. C.A. Belfiore and J.H. Park, "Decision Feedback Equalization," *Proc. of the IEEE,* Vol. 67, pp. 1143-1156, August 1979.

14. J. Salz, "Optimum Mean-Square Decision Feedback Equalization," *Bell System Tech. Journal,* Vol. 52, pp. 1341-1373, October 1973.

15. D.L. Duttweiler, J.E. Mazo, and D.G. Messerschmidt, "An Upper Bound on the Error Probability in Decision-Feedback Equalization," *IEEE Trans. on Information Theory,* July 1974, vol. IT-20, No. 4 pp. 490-497.

16. D. Williamson, R.A. Kennedy and G.W. Pulford, "Block Decision Feedback Equalization," *IEEE Trans. on Communications,* COM-40 pp. 255-264, February 1992.

17. K. Abend and B.D. Fritchman, "Statistical Detection for Communication Channels with Intersymbol Interference," *Proc. of the IEEE,* 58, pp. 779-785, May 1970.

18. G.D. Forney, "Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference," *IEEE Trans. on Information Theory,* IT-18, pp. 363-378, May 1972.

19. G. Ungerboeck, "Adaptive Maximum-Likelihood Receiver for Carrier Modulated Data Transmission," *IEEE Trans. on Communications,* COM-22, pp. 624-636, May 1974.

20. M.Y. Eyuboğlu, and S.U.H. Qureshi, "Reduced-State Sequence Estimation with Set Partitioning and Decision Feedback", *IEEE Trans. on Communications*, Vol. COM 36, pp. 13-20, 1988.

21. A. Duel-Hallen, and C. Heegard, "Delayed Decision-Feedback Sequence Estimation", *IEEE Trans. on Communications*, Vol. COM 37, pp. 428-436.

22. R.E. Kamel and Y. Bar-Ness, "Reduced-State Sequence Estimation of Digital Sequences in Dispersive Channels Using State Partitioning," *Electronics Letters,* Vol. 30, pp. 14-16, 6th January 1994.

23. D.N. Godard, "Self-Recovering Equalization and Carrier Tracking in Two-Dimensional Data Communication Systems," *IEEE Trans. on Communications*, COM-28, pp. 1967-1875, November 1980.

24. Y. Sato, "A Method of Self-Recovering Equalization for Multi-Level Amplitude Modulation," *IEEE Trans. on Communications*, COM-23, pp. 679-682, June 1975.

25. A. Benveniste, M. Goursat, and G. Ruget, "Robust Identification of a Nonminimum Phase System: Blind Adjustment of a Linear Equalizer in Data Communications," *IEEE Trans. on Automatic Control*, AC-25, pp. 385-399, June 1980.

26. S. Bellini and F. Rocca, "Blind Deconvolution: Polyspectra or Bussgang Techniques," In E. Biglieri and G. Prati, editors, *Digital Communications,* pp. 251-263, North-Holland, 1986, Elsevier Science Publishers B. V.

27. O. Shalvi and E. Weinstein, "New Criteria for Blind Deconvolution of non-minimum Phase Systems (Channels)," *IEEE Trans. on Information Theory,* IT-36, 00. 312-321, March 1990.

28. J.R. Treichler and M.G. Agee, "A New Approach to Multipath Correction of Constant Modulus Signals," *Trans. on Acoustics, Speech, and Signal Processing,* ASSP-31, pp. 349-472, April 1983.

29. J.R. Treichler and M.G. Larimore, "New Processing Techniques Based on the Constant Modulus Adaptive Algorithm," *IEEE Trans. on Acoustics, Speech, and Signal Processing,* ASSP-33, pp. 420-431, April 1985.

30. C.R. Johnson, "Admissibility in Blind Adaptive Channel Equalization," *IEEE Control Systems Magazine*, pp. 3-15 Jan. 1991.

31. Z. Ding, R.A. Kennedy, B.O. Anderson, and C.R. Johnson, "Ill-Convergence of Godard Blind Equalizers in Data Communications," *IEEE Trans. on Communications*, COM-39, pp. 1313-1328, Sept. 1991.

32. Z. Ding, R.A. Kennedy, B.O. Anderson, and C.R. Johnson, "Local Convergence of the Sato Blind Equalizer and Generalizations Under Practical Constraints," *IEEE Trans. on Information Theory*, IT-129 pp. 129-144, Jan. 1993.

33. Z. Ding, *Application Aspects of Blind Adaptive Equalizers in QAM Data Communications*, Ph. D. Dissertation, Cornell University, 1990.

34. S. Verdu, B. Anderson, and R.A. Kennedy, "Anchored Blind Equalization," *1991 Conf. Information Sciences and Systems,* Baltimore, MD. Mar. 1991, pp. 774-779.

35. S. Verdu, "On the Selection of Memoryless Adaptive Laws for Blind Equalization in Binary Communications," *Proc. of 6th Int. Conf. on Analysis and Optimization of Systems,* pp. 239-249, Nice, France, June 1984.

36. S. Vembu, S. Verdu, R.A. Kennedy, and W.A. Sethares, " Convex Cost Functions in Blind Equalization," *Proc. 25th Conf. Info. Sci. and Systems,* pp. 792-797, Baltimore, MD, March 1991.

37. R.A. Kennedy and Z. Ding, "Blind Adaptive Equalizers for Quadrature Amplitude Modulated Communication Systems Based on Convex Cost Functions," *Optical Engineering,* Vol. 31, pp. 1189-1199, June 1992.

38. B. Porat and B. Friedlander, "Blind Equalization of Digital Communication Channels Using High-Order Moments," *IEEE Trans. on Signal Processing*, SP-39, pp. 522-526, Feb. 1991.

39. D. Hatzinakos and C. Nikias, "Blind Equalization Using a Tricepstrum-Based Algorithm," *IEEE Trans. on Communications*, COM-39, pp. 669-682, May 1991.

40. N. Seshadri, "Joint Data and Channel Estimation Using Fast Blind Trellis Search Techniques," *GLOBECOM '90*, pp. 807.1.1-807.1.5

41. M. Ghosh and C. Weber, "Maximum-likelihood blind equalization, *Optical Engineering* vol. 31, pp. 1224-1228.

42. M. Feder and J.A. Capitovic, "Algorithms for Joint Channel Estimation and Data Recovery-Application to Equalization in Underwater Communications," *IEEE Journal of Ocean Engineering,* Vol. 16, pp. 42-55, Jan. 1991.

43. A.P. Dempster, N.M. Laird, and D.B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of Royal Statistics Society* vol. 39, pp. 1-38, 1977.

44. R.A. Kennedy, G. Pulford, B.D.O. Anderson, and R. R. Bitmead, "When Has a Decision-Directed Equalizer Converged?," *IEEE Trans. on Communications,* COM-37, pp. 879-884, August 1989.

45. R.E. Kamel and Y. Bar-Ness, "Equalization of AR Channels Using Decorrelation," *Memorandum,* August 21, 1993.

46. R.E. Kamel and Y. Bar-Ness, "Blind Decision Feedback Equalization Using the Decorrelation Criteria," *Communication Theory Mini Conference,* GLOBECOM '93, Houston TX, November 1993.

47. R.E. Kamel and Y. Bar-Ness, "Fast Blind DFE," *Memorandum,* August 14, 1993.

48. R.E. Kamel and Y. Bar-Ness, "Error Analysis of the Blind Decision Feedback Equalizer Using the Decorrelation Criterion," submitted to the *MILCOM '94.*

49. R.E. Kamel and Y. Bar-Ness, "Anchored Constant Modulus Algorithm (ACMA) for Blind Equalization," *SUPERCOMM/ICC '94*, New Orleans, LA, May 1994.

50. B. -P. Paris, "Self-Adaptive Maximum-Likelihood Sequence Estimation," *Communication Theory Mini Conference,* GLOBECOM '93, pp. 92-96.

51. R.E. Kamel and Y. Bar-Ness, "Blind Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference," *Electronics Letters,* to appear.

52. R.E. Kamel and Y. BarNess, "Reduced-State Viterbi Equalization Using State Partitioning," submitted to *Communication Theory Mini Conference,* GLOBECOM '94.

53. S.U.H. Qureshi and E.E. Newhall, "An Adaptive Receiver for Data Transmission over Time-Dispersive Channels," *IEEE Trans. on Information Theory*, Vol. IT 19, pp. 448-457, 1973.

54. W.U. Lee and F.S. Hill, " A Maximum-Likelihood Sequence Estimator with Decision-Feedback Equalization," *IEEE Trans. on Communications*, Vol COM 25, pp. 971-979, 1977.

55. W.H. Sheen and G.L. Stuber, "Error Probability for Reduced -State Sequence Estimation", *IEEE Journal on Selected Areas in Communications*, Vol. JSAC 10, 571-578.

56. K. Dogancay and R.A. Kennedy, "Testing for Parameter Convergence in Blind Adaptive Channel Equalization," *Proc. of Int. Workshop on Intelligent Signal Processing and Communication Systems*, 1993, pp. 1-6.

57. G. Ungerboeck, "Trellis-Coded Modulation with Redundant Signal Sets Part I: Introduction," *IEEE Communications Magazine,* vol. 25, pp. 5-11, February 1987.