Combined Channel Estimation and Turbo Equalization for Wireless Channels

Fu-Sheng Shiao, B.E. (Hons I)

A thesis submitted in partial fulfilment
of the requirements for the degree of
Master of Engineering
in
Electrical and Computer Engineering
at the
University of Canterbury,
Christchurch, New Zealand.

14 February 2007

ABSTRACT

Single-carrier linear modulation techniques combined with frequency-domain equalization provide a viable alternative to multicarrier techniques for combating multipath fading in channels with large delay spread. Such modulations tolerate frequency offset and have well controlled peak to average power ratio. They have comparable complexity to orthogonal frequency division multiplexing (OFDM) systems, and are more robust to synchronization errors. If error correction coding is used, then information can be iteratively passed between the equalizer and the decoder to improve performance. This is referred to as turbo equalization. To date, several turbo equalization schemes have been proposed, but little work has been done to address the problem of channel estimation for the turbo equalization process.

The work in this thesis considers frequency-domain turbo equalization with imperfect channel state information (CSI) at the receiver for different wireless channels. A receiver structure incorporating joint frequency-domain turbo equalization and time-domain channel estimation is developed. The novelty of this scheme lies in the combination of time-domain channel estimation and frequency-domain turbo equalization, and in its extension to high level modulation formats. The performance of the system is investigated by a combination of analysis and computer simulation. It is found that the system performs well over a range of dispersive channels.

ACKNOWLEDGMENTS

In writing this thesis I have been assisted in various way by a number of people and I would like to acknowledge their assistance here.

I would like to thank my supervisors, Prof. Des Taylor and Dr. Philippa Martin, whose patient guidance and encouragement has made this thesis possible.

Thank you to my parents. I am immensely grateful to Mum and Dad for their unfailing support, both moral and financial.

Thanks to the department computer staff and members in the communications laboratory for their friendship and for giving me assistance by answering my questions.

Finally, I would like to acknowledge the financial support of Tait Electronics Ltd., in the form of the Tait Electronics Ltd. Research Fellowship in Communications Engineering.

CONTENTS

ABSTRACT ii

ACKNOWLEDGMENTS iii

CHAPTER 1 INTRODUCTION 1

1.1 Wireless Communications 1
1.2 Notation 3
1.3 Signal Representation 3
1.3.1 Representation of Passband Signals 3
1.3.2 Vector Representation of Signals 6
1.4 The Wireless Channel 8
1.4.1 Physical Description 8
1.4.2 The Classical Uncorrelated Scattering Model 9
1.4.3 Mathematical Model for Mobile Wireless Channel 12
1.4.4 Simulating the Mobile Wireless Channel 13
1.4.5 Mathematical Model for Fixed Wireless Channel 15
1.4.6 Simulating the Fixed Wireless Channel 17
1.5 Scope of Thesis 18
1.6 Organization of Thesis 19

CHAPTER 2 BACKGROUND 20

2.1 Modulation Techniques 21
2.2 Pulse Shaping 22
2.3 Connection Between Fourier Series and Fourier Transforms 24
2.4 Turbo Equalization 25
2.4.1 Minimum Mean Squared Error (MMSE) Linear Estimation 27
2.5 Channel Estimation 28
2.5.1 The Generalized Recursive Least Squares (GRLS) Estimator 29
2.5.2 The Generalized Least Mean Squares (GLMS) Estimator 32
2.6 Error Correction Coding 33
2.6.1 Binary Convolutional Codes 33
2.6.2 Decoding Convolutional Codes 37

2.7 Summary 39

CHAPTER 3 FREQUENCY-DOMAIN TURBO LINEAR EQUALIZATION WITH CHANNEL ESTIMATION 40
3.1 System Overview 41
3.2 Channel Estimation Employing the GLMS Channel Estimator 48
3.3 The Equalizer Filter Matrix 52
3.3.1 Effect of Channel Estimation Error 56
3.4 Calculating Soft Information in the Decoder 59
3.4.1 Example for QPSK 63
3.4.2 Example for 16-QAM 66
3.5 Summary 68

CHAPTER 4 COMPUTER SIMULATIONS 69
4.1 Mobile Wireless Channel 70
4.1.1 System Error Performance with ideally known CSI 70
4.1.2 GLMS Estimator Performance 74
4.1.3 Overall System Error Performance 77
4.2 Fixed Wireless Channel 81
4.2.1 System Error Performance with ideally known CSI 81
4.2.2 GLMS Estimator Performance 84
4.2.3 Overall System Error Performance 87
4.3 Summary 91

CHAPTER 5 CONCLUSION 92
5.1 Summary 92
5.2 Future Work 94

APPENDIX A SUPPLEMENTARY SIMULATION RESULTS 95

REFERENCES 97

LIST OF FIGURES

1.1 Multipath propagation of electromagnetic waves. 9

1.2 Doppler power spectrum for each of the 11 fading processes of the 11-TAP channel. 13

1.3 Complex baseband model used to generate a single fading process. 14

1.4 Example of a single fading process simulated using the filtered noise approach of power with $f_dT = 0.05$. 15

1.5 Doppler power spectrum for each of the 3 fading processes of the SUI channel. 16

1.6 Example of SUI-5 channel power response simulated using the filtered noise approach sampled at $f_s = 20$Hz. 18

2.1 A typical wireless communication system. 20

2.2 Signal constellations employing Gray mapping. 22

2.3 Response of ideal ($\alpha = 0$) and raised cosine pulse shapes for different rolloff factors, α. 24

2.4 Encoder for the rate 1/2 $(5, 7)$ convolutional code. 33

2.5 Trellis diagram for the 1/2 rate convolutional encoder. 34

2.6 Minimum weight path starting and ending with the all-zero state. 35

2.7 Encoder for the rate 1/2 $(133, 171)$ convolutional code. 36

3.1 Proposed system using FD-TLE with channel estimation. 40

3.2 Overall proposed system operation flowchart. 41

3.3 The composition of a single transmission frame. 42

3.4 Example to show the composite channel response $h(t)$ spanning, for example, $L_h = L_p + L_c = 6$ symbol intervals. 44

3.5 Noise PSD for (a) $w(t)$ and (b) $n(t)$. 46

3.6 Example to show how the discrete sub-channel impulse response for $\gamma = 0$ is the symbol-rate sampled discrete impulse response of c(t). 51

3.7 Example of a received signal in a system employing a QPSK signal constellation. 64

3.8 Example of a received signal in a system employing a 16-QAM signal constellation. 66

4.1 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation. 71

4.2 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation. 71

4.3 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 2dB and 16-QAM signal constellation. 71

4.4 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 14dB and 16-QAM signal constellation. 72

4.5 BER performance of the FD-TLE system for the 11-TAP channel with 1/2-rate 4-state $(5, 7)$ and 64-state $(133, 171)$ convolutional codes and perfectly known CSI. 73

4.6 GLMS estimator performance for the 11-TAP channel with $L_T = 52$. 75

4.7 GLMS estimator performance for the 11-TAP channel with $L_T = 104$. 75

4.8 GLMS estimator performance for the 11-TAP channel with $L_T = 156$. 76

4.9 GLMS estimator performance for the 11-TAP channel with $L_T = 208$. 76

4.10 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation. 78

4.11 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation. 78

4.12 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 2dB and 16-QAM signal constellation. 78

4.13 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 16dB and 16-QAM signal constellation. 79

4.14 Overall system performance for the 11-TAP channel with the 1/2-rate 4-state $(5, 7)$ convolutional code. 80

4.15 Overall system performance for the 11-TAP channel with the 1/2-rate 64-state $(133, 171)$ convolutional code. 80

4.16 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation. 82

4.17 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation. 82

4.18 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a 2dB and 16-QAM signal constellation. 82

4.19 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a 14dB and 16-QAM signal constellation. 83

4.20 BER performance of the FD-TLE system for the SUI-5 channel with a 1/2-rate 4-state $(5, 7)$ convolutional code and 64-state $(133, 171)$ convolutional code and perfectly known CSI. 84

4.21 GLMS estimator performance for the SUI-5 channel with $L_T = 52$. 85

4.22 GLMS estimator performance for the SUI-5 channel with $L_T = 104$. 85

4.23 GLMS estimator performance for the SUI-5 channel with $L_T = 156$. 86

4.24 GLMS estimator performance for the SUI-5 channel with $L_T = 208$. 86

4.25 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation. 88

4.26 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation. 88

4.27 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 2dB and 16-QAM signal constellation. 89

4.28 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 16dB and 16-QAM signal constellation. 89

4.29 Overall system performance for the SUI-5 channel with the 1/2-rate 4-state $(5, 7)$ convolutional code. 90

4.30 Overall system performance for the SUI-5 channel with the 1/2-rate 64-state $(133, 171)$ convolutional code. 90

A.1 FD-TLE error performance comparison. 95
A.2 GLMS estimator performance comparison with 26 training symbols. 96

LIST OF TABLES

1.1 11-TAP Channel Parameters. 12

1.2 SUI channel models and the associated terrain type. 16

1.3 SUI-5 Channel Parameters [17]. 17

2.1 Polynomial Predictor Coefficients For Different P_G and N_G [37]. 31

2.2 Some rate 1/2 maximum free distance convolutional codes [21]. 36

4.1 GLMS estimator parameters for the 11-TAP channel. 77

4.2 GLMS estimator parameters for the SUI-5 channel. 87

Chapter 1

INTRODUCTION

1.1 WIRELESS COMMUNICATIONS

The history of wireless communications began in 1895 when Italian inventor Guglielmo Marconi successfully demonstrated the ability to communicate with ships sailing the English channel [1]. Since then, new wireless communication methods and services have been developed by people around the globe.

By 1934, a portion of the police stations in the United States had adopted wireless communication systems for public safety using amplitude modulation (AM). Frequency modulation (FM) was later adopted for these systems after Edwin Armstrong demonstrated FM for the first time in 1935 [2]. After the late 1930s, FM was used as the primary modulation technique.

Following the development of the cellular concept from Bell Laboratories in conjunction with the development of highly reliable solid-state, radio frequency hardware in the 1970s, cellular radio and personal communication services (PCS) became available to the general public [2]. The first generation (1G) cellular systems continued to use analog modulation techniques such as FM. However, due to the explosive growth in the number of users, further advances in technology and the increasing demand for the transmission of digital data, digital modulation techniques were developed and introduced into second generation (2G) cellular systems [3].

Recently, analog systems have been rapidly converted to digital systems [4]. As technology continued to advance at a tremendous rate, third generation cellular (3G) systems were born [3]. They allow users to receive live music, conduct interactive

web sessions and have simultaneous voice and data access with multiple parties using a single mobile handset, whether driving, walking, or standing still in an office setting [2].

Concurrent with the development of cellular systems, there has been an exponential growth in other areas employing wireless communications. These include internet applications such as local and wide area wireless networks, and multimedia applications such as satellite and television broadcasting [5]. New standards and technologies are being implemented to allow wireless networks to replace fiber optic or copper lines between fixed points several kilometers apart. Similarly, wireless networks have been increasing used as a replacement to wired networks within metropolitan areas [3, 6].

Today, the development of wireless communication systems continues by addressing many of the operating constraints apparent in present systems. These include the pressure to support more subscribers, the finite and tightly controlled allocations of radio frequency spectrum, the desire for longer battery life, the desire to reduce the cost of a system and the demand for higher data rates and quality of service (QoS). Moreover, as communication systems continue to improve following these constraints, the need to insert new technologies into commercial products quickly requires that the design be done in a timely, cost-effective manner. These demands can be met through the use of powerful computer-aided analysis [7], which is often the first step in the development process.

In addition to the constraints mentioned above, channel characteristics in the physical layer also need to be taken into consideration in the development process. Wireless channels are almost dispersive in nature. They allow signals transmitted from the transmitter to travel through different paths before arriving at the receiver. This introduces intersymbol interference (ISI) and it has a direct impact on system performance. To insure successful transmission, the effect of ISI must be remedied.

The goal of this thesis is to demonstrate the functionality of a theoretical communications system by means of mathematical analysis and computer simulation. The proposed system focuses on means of reducing ISI through channel estimation and frequency domain Turbo Equalization. Results obtained from this study may be of use in the future for practical implementation or further research purposes.

The remainder of this chapter is organized as follows. First, the notation used throughout this thesis will be defined. Second, the representation of signals in this thesis is described. Third, the wireless channel including its characteristics and its modeling are discussed. Finally, the scope and contribution of this thesis is explained.

1.2 NOTATION

The following notation will be employed throughout this thesis. Time-domain variables will be denoted by lower-case letters and frequency-domain by upper-case letters. Vectors and matrices will be shown in bold font and scalars in italic font. In addition, $\mathbf{I}_N$ denotes the $(N \times N)$ identity matrix and $\mathbf{0}_{M,N}$ denotes the $(M \times N)$ all-zeros matrix. Furthermore, $diag(\cdot)$ denotes the diagonalization matrix operation, $(\cdot)^*$ denotes the complex conjugate operation, $(\cdot)^H$ denotes the Hermitian operation, $(\cdot)^T$ denotes the transpose operation, and $\otimes$ denotes the convolution operation.

1.3 SIGNAL REPRESENTATION

In order to analyze communication systems, we must be able to represent signals in a rigorous mathematical format. In this section, we summarize the various signal representations employed in this thesis.

1.3.1 Representation of Passband Signals

In modern wireless communication systems, signals are transmitted in passband form, where the center frequency is often orders of magnitude higher than the bandwidth occupied by the signal. If we wish to analyze such systems, it is often convenient to consider the complex baseband equivalent signal model.

Following the discussion in [8], consider a passband signal $x(t)$ with bandwidth $2W$ centered about some carrier frequency f_c. This may be expressed as

$$\begin{aligned} x(t) &= Re\{\tilde{x}(t)\exp(j2\pi f_c t)\} \\ &= x_I(t)\cos(2\pi f_c t) - x_Q(t)\sin(2\pi f_c t), \end{aligned} \tag{1.1}$$

where $\tilde{x}(t)$ is the complex envelope of $x(t)$ and may be written as

$$\tilde{x}(t) = x_I(t) + jx_Q(t). \tag{1.2}$$

The signals $x_I(t)$ and $x_Q(t)$ are both real valued baseband signals, and are referred to as the in-phase and quadrature components of $x(t)$, respectively. Let the signal $x(t)$ be applied to a linear time-invariant passband system with impulse response $h(t)$ and frequency response $H(f)$. Assuming that $h(t)$ has a bandwidth of $2B$ centered at f_c with $B \leq W$, we may express $h(t)$ in terms of its in-phase and quadrature components as

$$\begin{aligned} h(t) &= Re\{\tilde{h}(t)\exp(j2\pi f_c t)\} \\ &= h_I(t)\cos(2\pi f_c t) - h_Q(t)\sin(2\pi f_c t), \end{aligned} \tag{1.3}$$

where

$$\tilde{h}(t) = h_I(t) + jh_Q(t). \tag{1.4}$$

Let the output of the system be denoted by $y(t)$, where

$$\begin{aligned} y(t) &= h(t) \otimes x(t) \\ &= \int_{-\infty}^{\infty} h(\tau)x(t-\tau)\ d\tau. \end{aligned} \tag{1.5}$$

Rewriting (1.5) in terms of the complex envelopes of $x(t)$ and $h(t)$, and using the property that [8]

$$\begin{aligned} & \int_{-\infty}^{\infty} Re\left\{\tilde{h}(\tau)\exp(j2\pi f_c\tau)\right\} Re\left\{\tilde{x}(\tau)\exp(j2\pi f_c\tau)\right\}\ d\tau \\ = \ & \frac{1}{2}Re\left\{\int_{-\infty}^{\infty} \tilde{h}(\tau)\exp(j2\pi f_c\tau)\left(\tilde{x}(\tau)\exp(j2\pi f_c\tau)\right)^*\ d\tau\right\} \\ = \ & \frac{1}{2}Re\left\{\int_{-\infty}^{\infty} \tilde{h}(\tau)\exp(j2\pi f_c\tau)\tilde{x}^*(\tau)\exp(-j2\pi f_c\tau)\ d\tau\right\}, \end{aligned} \tag{1.6}$$

we obtain

$$\begin{aligned} y(t) &= \frac{1}{2}Re\left\{\int_{-\infty}^{\infty}\tilde{h}(\tau)\exp(j2\pi f_c\tau)\tilde{x}(t-\tau)\exp(j2\pi f_c(t-\tau))\ d\tau\right\} \\ &= \frac{1}{2}Re\left\{\exp(j2\pi f_c t)\int_{-\infty}^{\infty}\tilde{h}(\tau)\tilde{x}(t-\tau)\ d\tau\right\}. \end{aligned} \tag{1.7}$$

Note that using $x(-\tau)$ in place of $x(\tau)$ has the effect of removing the complex conjugation of the right-hand side of (1.6). Since $y(t)$ is also a passband signal, we may represent $y(t)$ in terms of its complex envelope $\tilde{y}(t)$ as

$$\begin{aligned} y(t) &= Re\{\tilde{y}(t)\exp(j2\pi f_c t)\} \\ &= y_I(t)\cos(2\pi f_c t) - y_Q(t)\sin(2\pi f_c t). \end{aligned} \tag{1.8}$$

If we compare equations (1.7) and (1.8), we see that $\tilde{y}(t)$ is related to $\tilde{x}(t)$ and $\tilde{h}(t)$ as

$$\begin{aligned} 2\tilde{y}(t) &= 2y_I(t) + j2y_Q(t) \\ &= \int_{-\infty}^{\infty}\tilde{h}(\tau)\tilde{x}(t-\tau)\ d\tau \\ &= \tilde{h}(t)\otimes\tilde{x}(t). \end{aligned} \tag{1.9}$$

Lastly, substituting (1.2) and (1.4) into (1.9) we obtain

$$\begin{aligned} 2\tilde{y}(t) &= [h_I(t) + jh_Q(t)]\otimes[x_I(t) + jx_Q(t)] \\ &= [h_I(t)\otimes x_I(t) - h_Q(t)\otimes x_Q(t)] \\ &\quad +j[h_Q(t)\otimes x_I(t) + h_I(t)\otimes x_Q(t)]. \end{aligned} \tag{1.10}$$

Hence,

$$2y_I(t) = h_I(t)\otimes x_I(t) - h_Q(t)\otimes x_Q(t) \tag{1.11}$$

and

$$2y_Q(t) = h_Q(t)\otimes x_I(t) + h_I(t)\otimes x_Q(t). \tag{1.12}$$

Note that there is a factor of two when relating the complex baseband signal model to a passband signal.

The significance of this result is that it is possible to represent any passband system in terms of its baseband equivalents. This representation provides the basis for efficient simulation of passband systems as well as DSP implementation of complex filter structures. Throughout the remainder of this thesis, complex baseband equivalent will be used.

1.3.2 Vector Representation of Signals

Many signals in a communication system may be modeled as sample functions of random processes [9]. If we wish to express the probability distributions associated with these random processes, it is often convenient to represent a continuous signal in vector form, which is usually constructed over an arbitrary interval. The vector components are then random variables.

Consider a continuous time signal $s(t)$ of duration $T_0 = [t_a, t_b]$. One method for constructing a vector representation of a continuous time signal is to express the signal as a weighted sum of orthonormal basis functions $\phi_n(t)$, where

$$\int_{T_0} \phi_i(t)\phi_j(t) = \begin{cases} 1, & i = j \\ 0, & \text{otherwise.} \end{cases} \tag{1.13}$$

In cases where the continuous time signal is known to be from a set of M energy signals $\{s_i(t)\}$, it is possible to express $s(t)$ exactly in terms of N orthonormal basis functions, where $N \leq M$ [10]. This leads to

$$s_i(t) = \sum_{n=1}^{N} s_{ij}\phi_j(t) \quad \begin{cases} t_a \leq t \leq t_b \\ i = 1, 2, \ldots, M, \end{cases} \tag{1.14}$$

where the coefficients s_{ij} are defined by

$$s_{ij} = \int_{T_0} s_i(t)\phi_j(t)\ dt \quad \begin{cases} i = 1, 2, \ldots, M \\ \\ j = 1, 2, \ldots, N. \end{cases} \tag{1.15}$$

The vector representation is then given by

$$\mathbf{s}_i = [s_{i1}, s_{i2}, \ldots, s_{iN}]^T, \quad i = 1, 2, \ldots, M. \tag{1.16}$$

One common use of this technique is representing signals within a signal constellation. The normalized sine and cosine functions with period T_0 are the orthonormal basis functions, and each signal is represented as a 2-element vector.

An alternative approach to constructing vector representations of continuous time signals is to sample the signal at evenly spaced instants, resulting in the vector

$$\mathbf{s} = [s(t_0), s(t_1), \ldots, s(t_{N-1})]^T, \tag{1.17}$$

where

$$t_n = t_a + n\tau, \tag{1.18}$$

with the sampling period defined as

$$\tau = \frac{t_b - t_a}{N}, \tag{1.19}$$

where N is the total number of sampling instants.

Let f_{max} be the highest frequency component present in $s(t)$. The sampling frequency $f_s = 1/\tau$ is set to be greater than $2f_{max}$ so that $s(t)$ may be reconstructed from the samples with no errors according to the sampling theorem. This technique is often used in signal detection and processing where T_0 is the period for each frame of signals.

In this thesis, signals within the signal constellation sets are represented as complex numbers, which are equivalent to the 2-element vectors corresponding to the sine and cosine basis functions. Sampling is used at the receiver to construct vector representa-

tions of the received signal for signal detection and further processing.

1.4 THE WIRELESS CHANNEL

The effective design, assessment, and installation of a wireless system requires an accurate characterization of the channel. The channel characteristics vary from one environment to another. In this section we will discuss these characteristics and the way they may be modeled mathematically and in computer simulations.

In this thesis we consider wireless channels for two different scenarios. The first scenario occurs when the communication terminals are non-stationary. In this scenario, the channel model represents a mobile wireless channel and is suitable for communication between mobile transceivers. The second scenario occurs when the communications terminals are stationary. In this scenario, the channel represents a fixed wireless channel and is suitable for communication between base stations or computer terminals, such as wireless communication for local area networks (LANs) and metropolitan area networks (MANs).

1.4.1 Physical Description

The environment in which a wireless communication system operates often contains obstacles that are capable of reflecting electromagnetic waves radiated from the transmitter. Examples of such obstacles include buildings, vehicles, trees, hills, and even the ionosphere. As a result, the transmitted signal arrives at the receiver from various directions using many different paths. This phenomenon is known as multipath propagation and is illustrated in Fig. 1.1.

In general, the phase and amplitude of the component signals arriving at the receiver antenna will be different due to different obstacles in each path and the difference in path lengths [11]. The resulting additive combination of the multipath components at the receiver can be constructive or destructive depending on the phase fluctuations in the received signals. These effects constitute the phenomenon commonly referred to as multipath fading.

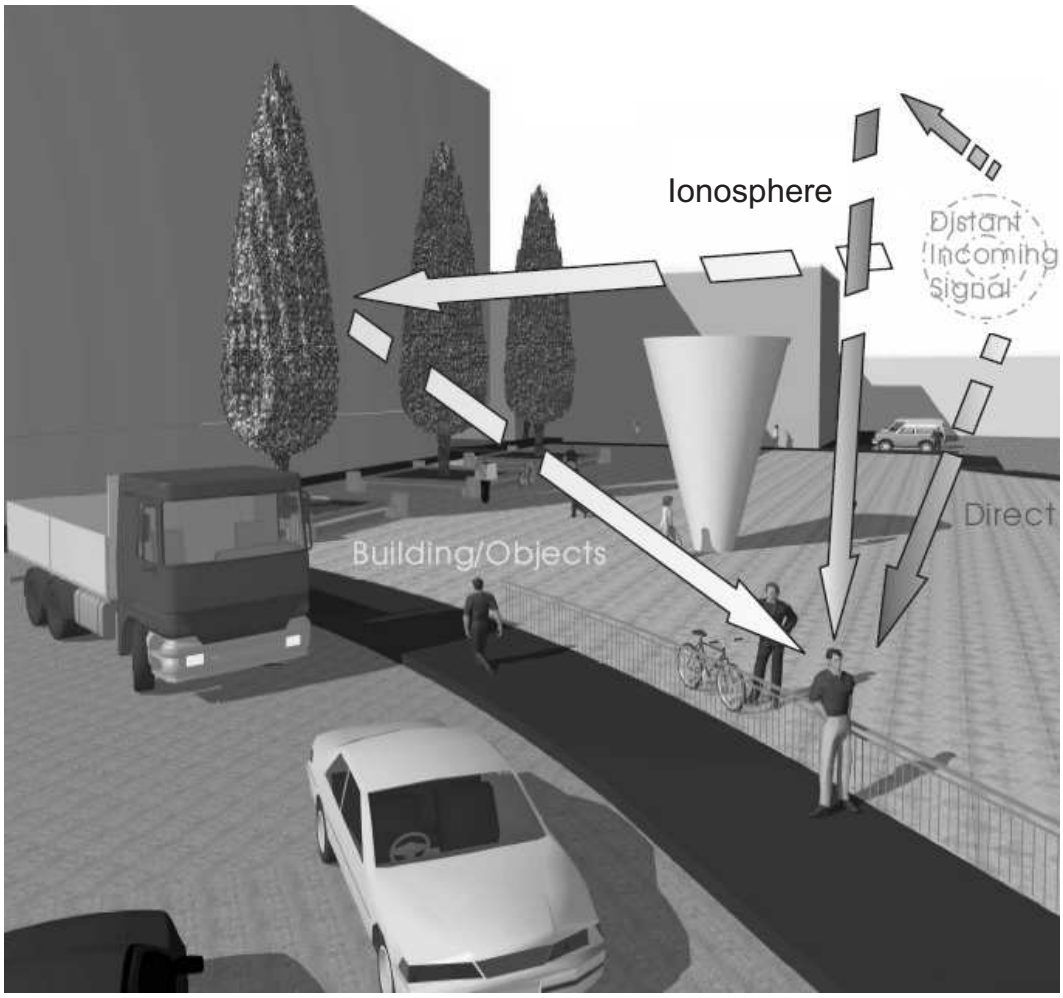


Figure 1.1 Multipath propagation of electromagnetic waves.

In addition to multipath fading, any relative motion between the transmitter and the receiver will result in Doppler shifting of the received signals. Furthermore, due to the variation in velocity of movement for each arriving path, the received signal exhibits a spectrum of Doppler shifting. This spectrum is often referred to as the Doppler spectrum. The amount of spreading of the transmitted signal frequency associated with the Doppler spectrum is referred to as the Doppler spread.

1.4.2 The Classical Uncorrelated Scattering Model

The classical uncorrelated scattering channel model was developed to describe signal transmissions over a variety of wireless channels having randomly time-varying impulse responses. This is the underlying model used for the channels employed in this thesis.

Here we define the classical uncorrelated scattering model following the discussion presented in [12].

We start by defining the time-varying channel impulse response. If we transmit an impulse $\delta(t)$ over a multipath transmission medium, the received signal will be a sum of several impulses with different arrival times, magnitudes and phases. Assuming that the effects of the medium are sufficiently random and the number of multipath signal components is sufficiently large, the central limit theorem may be invoked. The composite impulse response of the channel can therefore be accurately represented by a complex Gaussian process given by

$$h(\tau, t) = \sum_{i=1}^{L} \mathcal{G}_i(t) \exp(j\phi_i(t))\delta(\tau - \tau_i), \tag{1.20}$$

where $\mathcal{G}_i(t)$ and $\phi_i(t)$ represent the amplitude and phase of the i^{th} path arriving at delay τ_i and L is the number of paths. This may be modeled as a tapped delay line with time varying complex tap gains.

Now we assume that the signal variations on paths arriving at different delays are uncorrelated and that the correlation properties of the channel are stationary. This is referred to as wide-sense stationary uncorrelated scattering (WSSUS) [13]. This means the autocorrelation of the observed impulse response at two different delays and two different times has a non-zero value only when the delays are the same, and depends solely on the difference in time of occurrence of the two impulse responses. In mathematical notation, this is given by

$$R_{hh}(\tau_1, \tau_2; t_1, t_2) = E\{h(\tau_1, t_1)h(\tau_2, t_2)\} = R_{hh}(\tau_1; \Delta t)\delta(\tau_1 - \tau_2). \tag{1.21}$$

where $\Delta t = t_1 - t_2$. For the special case where $\Delta t = 0$, we get

$$Q(\tau) = R_{hh}(\tau; 0). \tag{1.22}$$

This is referred to as the *delay power spectrum* of the channel and it represents the received power as a function of delay.

Since $h(\tau, t)$ is a wide-sense stationary Gaussian process, the frequency response $H(f; t)$ obtained by taking the Fourier transform on $h(\tau, t)$ with respect to τ is also a wide-sense stationary zero-mean Gaussian process. To characterize the variations in frequency statistically, we compute the frequency-domain autocorrelation function defined as

$$\begin{aligned} R_{Hh}(f_1, f_2; \Delta t) &= E\{H^*(f_1; t)H(f_2; t + \Delta t)\} \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} E\{h^*(f_1; t)h(f_2; t + \Delta t)\} \\ &\quad \cdot \exp\left(j2\pi(f_1\tau_1 - f_2\tau_2)\right)\ d\tau_1 d\tau_2 \\ &= \int_{-\infty}^{\infty} R_{hh}(\tau_1; \Delta t) \exp\left(j2\pi\Delta f\tau_1\right)\ d\tau_1 \\ &= R_{Hh}(\Delta f; \Delta t), \end{aligned} \tag{1.23}$$

where $\Delta f = f_1 - f_2$. Taking the Fourier transform of (1.23) with respect to Δt we get

$$R_{HH}(\Delta f; \lambda) = \int_{-\infty}^{\infty} R_{Hh}(\Delta f; \Delta t) \exp\left(j2\pi\lambda\Delta t\tau_1\right)\ d(\Delta t). \tag{1.24}$$

For the special case where $\Delta f = 0$, the transform gives

$$D(\lambda) = R_{HH}(0; \lambda). \tag{1.25}$$

This is referred to as the *Doppler power spectrum* of the channel and it represents the power distribution of the Doppler shift.

The *scattering function* of a channel,

$$S(\tau, \lambda) = R_{hH}(\tau, \lambda) \tag{1.26}$$

is defined as the inverse Fourier transform of $R_{HH}(\Delta f, \lambda)$ with respect to the Δf variable, which is the Fourier transform of $R_{hh}(\tau, \Delta t)$ with respect to the Δt variable. With the usual assumption that the time and frequency components are independent, the formulation of the uncorrelated scattering function is obtained by decomposing the

scattering function into the delay and Doppler spectra and is defined by

$$S(\tau, \lambda) \quad = \quad Q(\tau) \cdot D(\lambda). \tag{1.27}$$

In practice, in order to measure the scattering function of the channel, the received signal in individual taps of a tapped delay line is analyzed in the frequency domain.

Having defined the scattering function for the classical uncorrelated scattering model, various channels may be defined by specifying the delay power profile and the Doppler power spectrum associated with each channel. In addition, for channels where there is a line-of-sight (LOS) path between the transmitter and receiver, the channel is defined by the scattering function plus a constant function. This function is often specified in a similar form to the scattering function with the Doppler spectrum being a single impulse.

1.4.3 Mathematical Model for Mobile Wireless Channel

The mobile wireless channel model used in the simulations of this thesis is based on that of [14, 15]. The channel response is assumed to have no LOS component and uses 11 equally spaced taps with $1\mu s$ spacing. The delay power profile for this particular channel $Q_{11-TAP}(\tau)$ is set to a truncated exponential function with mean $2.5\mu s$ and maximum delay $\tau_{max} = 10\mu s$. This channel will be referred to as the 11-TAP channel throughout the rest of the thesis, and the parameters for this channel are summarized in Table. 1.1.

Table 1.1 11-TAP Channel Parameters.

No. of Taps	11
Tap spacing	1 μs
Doppler ($f_d T$)	0.001 for all taps
Power delay profile	Exponential with mean $2.5\mu s$, $\tau_{max} = 10\mu s$

The Doppler power spectrum for the 11-TAP channel is set to have the following

response [11]

$$D_{11-TAP}(f) = \begin{cases} \frac{1}{\pi\sqrt{f_d^2 - f^2}}, & |f| \leq f_d \\ 0, & \text{otherwise,} \end{cases} \tag{1.28}$$

where f is the frequency and f_d is the maximum Doppler frequency. A graphical representation of this Doppler power spectrum is show in Fig. 1.2.

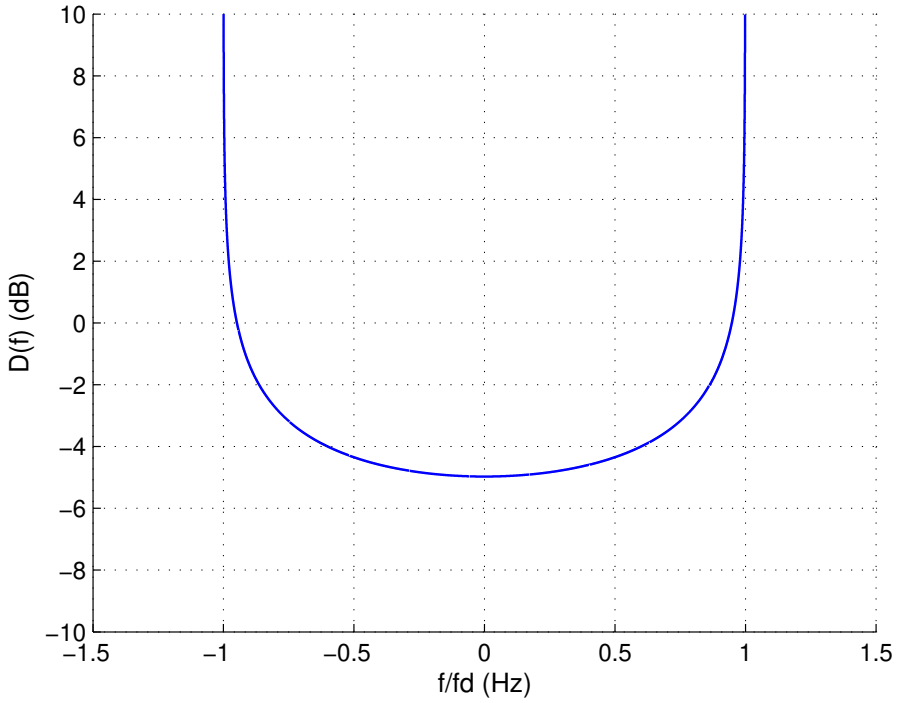


Figure 1.2 Doppler power spectrum for each of the 11 fading processes of the 11-TAP channel.

1.4.4 Simulating the Mobile Wireless Channel

To simulate a single fading process (from a total of 11) of the 11-TAP channel, we use the filtered noise approach of [16]. For the channel impulse response $h_{Mobile}(\tau_i, t)$ at a set delay τ_i, the filtered noise approach passes two independent additive white Gaussian noise (AWGN) sources through specially designed Doppler filters, then combines them to form the simulated complex channel impulse response as shown in Fig. 1.3

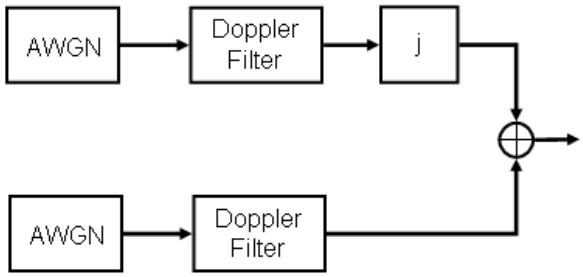


Figure 1.3 Complex baseband model used to generate a single fading process.

The Doppler filters play an important role as they determine the amount of Doppler spread. To obtain the Doppler spectrum of (1.28), the required filters have the impulse response [16]

$$h_d(t) \quad = \quad K\frac{J_{1/4}(2\pi f_d T t)}{\sqrt[4]{t}}, \tag{1.29}$$

where K is a time-independent constant, $J_{1/4}(\cdot)$ is the Bessel function defined as

$$J_{1/4}(t) \quad = \quad \frac{1}{2\pi}\int_{-\pi}^{\pi} \exp(-j(\alpha\tau - t\sin\tau))\ d\tau\Big|_{\alpha=1/4}. \tag{1.30}$$

The value of $h_d(t)$ at $t = 0$ is given by

$$\lim_{t\to 0} K\frac{J_{1/4}(2\pi f_d T t)}{\sqrt[4]{t}} \quad = \quad K\frac{\sqrt[4]{f_d T \pi}}{\Gamma(5/4)} \simeq 1.468813\sqrt[4]{f_d T}, \tag{1.31}$$

where $\Gamma(\cdot)$ is the Gamma function defined as

$$\Gamma(t) = \int_0^{\infty} \tau^{t-1}\exp(-\tau)\ d\tau. \tag{1.32}$$

Since (1.29) has infinite length, a windowed approximation is required. A suitable windowing function, such as the Hanning window, is used to smooth the abrupt transition caused by the truncation. Fig. 1.4 illustrates the time variation of power of a fading process simulated using the filtered noise approach for a normalized Doppler frequency of $f_d T = 0.05$.

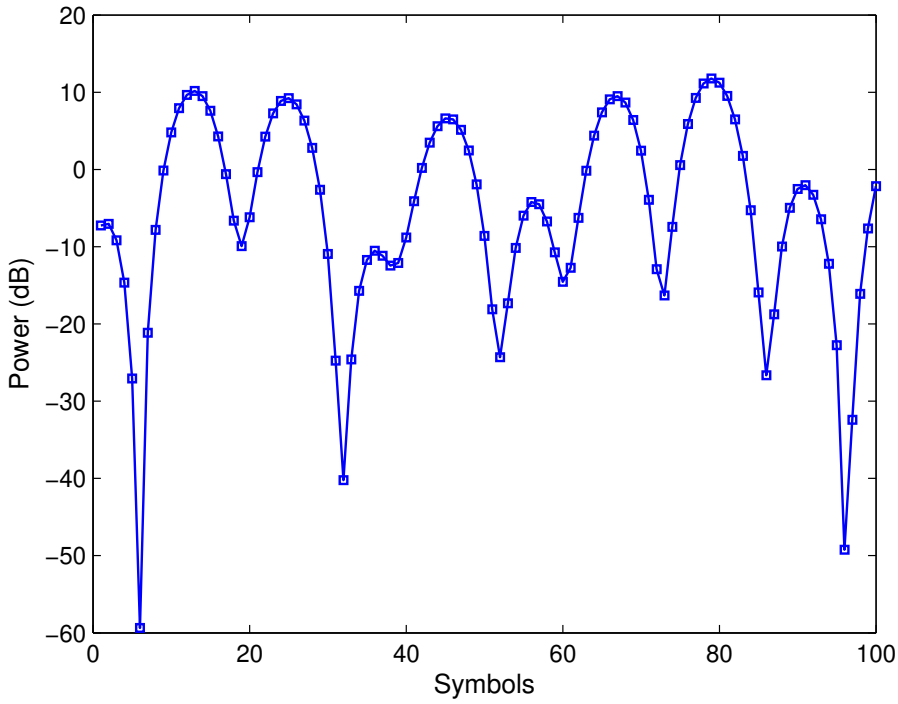


Figure 1.4 Example of a single fading process simulated using the filtered noise approach of power with $f_d T = 0.05$.

For the complete channel $h_{11-TAP}(\tau, t)$, the filtered noise approach is applied for each of the 11 delays τ_i with the noise power adjusted so that the output power follows the power delay profile $Q_{11-TAP}(\tau)$.

1.4.5 Mathematical Model for Fixed Wireless Channel

The fixed wireless channel being considered in this thesis originates from a family of channel models known as the Stanford University interim (SUI) channel models, which have been adopted by the IEEE 802.16a standards committee for evaluating broadband wireless systems in the $2 - 11$GHz bands [17, 18]. There are six typical SUI channel models, which are suitable for the three main suburban terrain types that are typical of the continental US [19]. Terrain category A corresponds to a hilly environment with moderate to heavy tree density. Terrain category B corresponds to a hilly environment with light tree density (or a flat environment with moderate to heavy tree density), and terrain category C corresponds to a flat environment with light tree density. The SUI models corresponding to each terrain type are listed in Table 1.2.

Table 1.2 SUI channel models and the associated terrain type.

Terrain Type	SUI Channels
C	SUI-1, SUI-2
B	SUI-3, SUI-4
A	SUI-5, SUI-6

Each of the six SUI channels is modeled as 3 distinct paths with different relative powers and delays. For those channels that consists of a LOS path, the Doppler power spectrum of the constant component associated with the LOS path is modeled as an impulse at $f = 0$Hz. The Doppler power spectrum of the scattering component can be described by the scattering function approximated by

$$D_{SUI}(f) = \begin{cases} 1 - 1.72\left(\frac{f}{f_d}\right)^2 + 0.785\left(\frac{f}{f_d}\right)^4 & \left|\frac{f}{f_d}\right| \leq 1 \\ 0 & \left|\frac{f}{f_d}\right| > 1 \end{cases} \tag{1.33}$$

and a graphical representation of this particular spectrum is shown in Fig. 1.5.

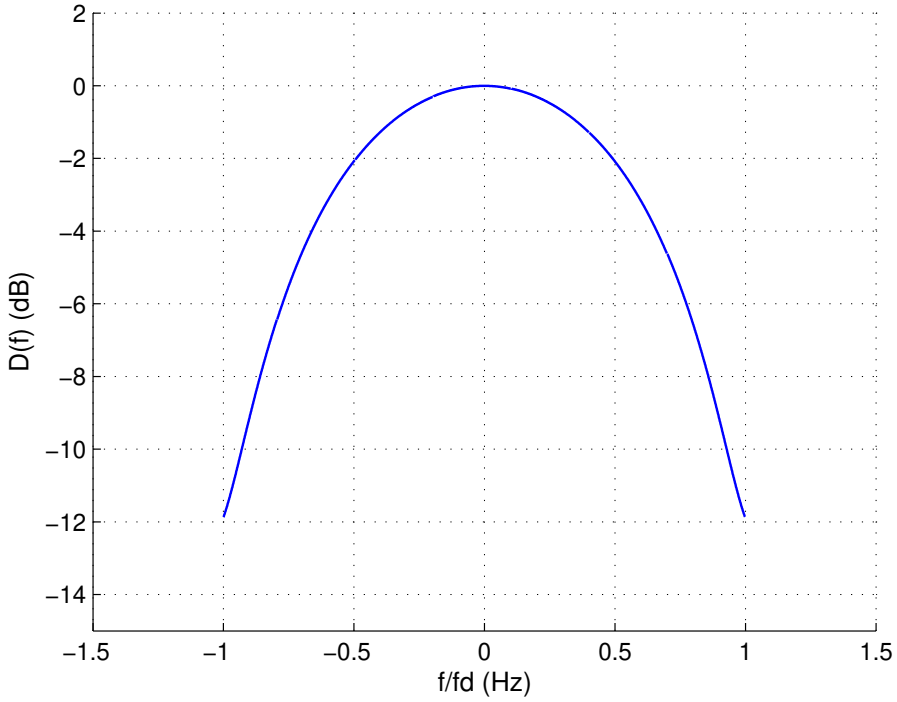


Figure 1.5 Doppler power spectrum for each of the 3 fading processes of the SUI channel.

For the work presented in this thesis, the SUI-5 channel model is considered. The characteristics of the SUI-5 channel are summarized in Table. 1.3. Since no constant component is present for each tap of the SUI-5 channel, it assumes no LOS.

Table 1.3 SUI-5 Channel Parameters [17].

	Tap 1	Tap 2	Tap 3	Units
Delay	0	5	10	μs
Power (constant component)	0	0	0	linear
Power (scatter component)	0	-5	-10	dB
Doppler f_d	2	1.5	2.5	Hz

1.4.6 Simulating the Fixed Wireless Channel

The filtered noise approach is also employed to simulate the SUI-5 channel [17]. The Doppler filter coefficients are obtained as follows. First, we take the square root of the Doppler spectrum $D_{SUI}(f)$ to obtain $|H_{SUI}(f)|$ for each of the 3 channel taps. The Doppler spectrum for each of the 3 taps differs as they have different f_d values. Second, the real part of the inverse Fourier transform of $|H_{SUI}(f)|$ is taken as the impulse response for the Doppler filters. Third, the impulse response is sampled according to the number of filter taps required to produce the filter coefficients. The number of filter taps can be determined as a tradeoff between accuracy and complexity. Lastly, the filtered outputs are normalized to ensure that the average channel energy sums to unity.

A simulated example of the SUI-5 channel power response following the characteristics defined in Table. 1.3 with 256 Doppler filter taps is shown in Fig. 1.6. Note that the SUI-5 channel has a maximum Doppler of 2.5Hz, so the minimum sampling rate required is 5Hz. However, it is possible to increase the channel sampling rate to suit the system of interest by means of interpolation. In Fig. 1.6, the sampling rate is increased to 20Hz for better resolution.

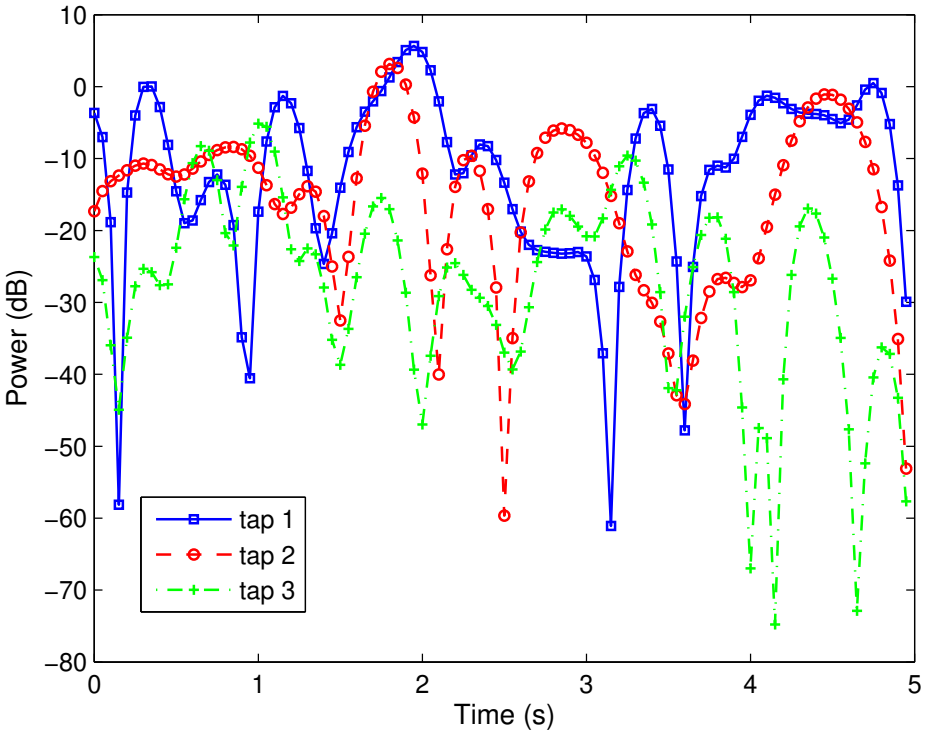


Figure 1.6 Example of SUI-5 channel power response simulated using the filtered noise approach sampled at $f_s = 20$Hz.

1.5 SCOPE OF THESIS

The thesis is primarily concerned with the investigation of frequency-domain turbo linear equalization (FD-TLE) with imperfect channel state information (CSI). The novelty lies in the proposed receiver structure, which is obtained by modifying and integrating existing channel estimation and FD-TLE algorithms. The integrated system is explored through detailed analysis and computer simulations. To the best of our knowledge no published literature to date included an investigation of FD-TLE with estimated rather than ideal CSI. This provides the motivation for the work of this thesis, as well as the possibility of bringing FD-TLE systems one step closer to practical implementation.

1.6 ORGANIZATION OF THESIS

In this chapter some preliminary topics have been discussed. A brief introduction to the history of wireless communications was given, followed by definitions for the signal representation techniques employed in this thesis. Furthermore, a discussion of the channel models that are of interest for the work presented in this thesis was also included. In Chapter 2, some background information is provided to explain the basic components in a typical communication system. These include, modulation techniques, pulse shaping, introduction to turbo equalization, channel estimation, and error correction coding. In Chapter 3, the proposed system model is defined. An in depth discussion combined with examples of the turbo equalization process combined with channel estimation is also presented in Chapter 3. In Chapter 4, the performance is investigated by means of computer simulation. Lastly, a summary of the work presented in this thesis and a discussion of possible extensions are given in Chapter 5.

Chapter 2

BACKGROUND

This chapter provides an overview of digital wireless communication theory and existing techniques that are relevant to the work presented in this thesis. A typical digital wireless communication system is shown in Fig. 2.1, which consists of a transmitter, a channel, and a receiver. At the transmitter, the encoder provides error correction coding for the system, the modulator maps binary bits onto a signal constellation, and the transmit filter provides pulse shaping of the transmitted signals. During transmission over a wireless channel, the effect of multipath propagation and receiver front end noise alters the transmitted signal and causes data errors at the receiver output. At the receiver, a portion of the overall received signal is used to update knowledge of the current CSI estimate, which is passed from the channel estimator to the equalizer. The equalizer then equalizes the received signals. The decoder then tries to correct any errors using the constraints of the error correction code and outputs the decoded data.

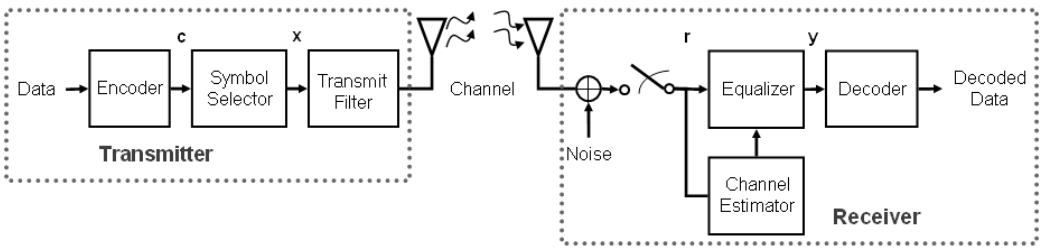


Figure 2.1 A typical wireless communication system.

2.1 MODULATION TECHNIQUES

The modulator of Fig. 2.1 varies the frequency, amplitude and/or phase of a carrier may be varied in order to convey digital information [20]. For example, frequency-shift keying (FSK) modifies the carrier frequency with time, pulse amplitude modulation (PAM) varies the carrier amplitude with time and phase-shift keying (PSK) varies the phase of the carrier with time.

The modulation techniques being considered in this thesis include quadrature phase shift keying (QPSK) and 16 quadrature amplitude modulation (16-QAM), which have 4 and 16 constellation points, respectively. These techniques are considered as they are linear modulations that provide high spectral efficiency while being relatively simple to implement. The carrier frequency remains constant for both QPSK and 16-QAM. Only the phase vary in time for QPSK, while both the amplitude and phase vary with time for 16-QAM. In mathematical terms, the QPSK and 16-QAM signal constellations with unit energy can be expressed as

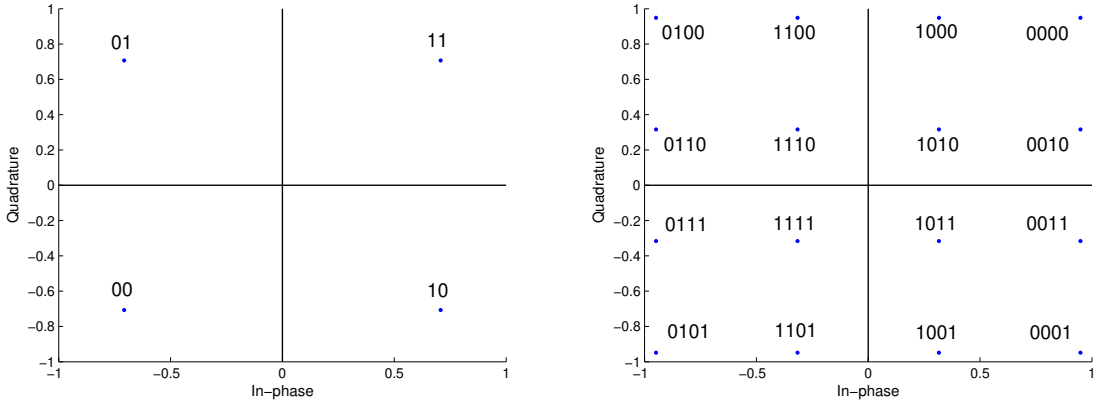
$$\mathcal{X}_{QPSK} = \left\{ \mathcal{X}_m = \exp\left(j\frac{\pi(2m+1)}{4} \right),\ m = 0 \dots 3 \right\}, \tag{2.1}$$

and

$$\mathcal{X}_{16-QAM} = \left\{ \mathcal{X}_{m,n} = \frac{m}{\sqrt{10}} + j\frac{n}{\sqrt{10}},\ m,n \in \{1,3\} \right\}, \tag{2.2}$$

respectively.

Lastly, Gray mapping is used to map the coded bits to points in the signal constellation. The QPSK and 16-QAM signal constellations with the Gray mapping used in this thesis are shown in Fig. 2.2.



(a) QPSK. (b) 16-QAM.

Figure 2.2 Signal constellations employing Gray mapping.

2.2 PULSE SHAPING

Before we can transmit the modulated signal, it is necessary to pass it through a transmit filter as shown in Fig 2.1. The purpose of the transmit filter is to provide a pulse shape to the transmitted signal. Usually, the pulse shape needs to be chosen carefully as it plays an essential role in determining the spectral characteristics of the communication system. In particular, the Nyquist criterion must be satisfied for inter-symbol interference (ISI) free transmission.

Consider the pulse shape $p(t)$. If we wish to transmit real symbols at a rate $R = 1/T$, assuming noise and channel effects are negligible, the Nyquist criterion states that [8, 21]

$$\sum_{n=-\infty}^{\infty} P(f - nR) = T \tag{2.3}$$

must be satisfied. The simplest way to satisfy this criterion is to make $P(f)$ a rectangular function, resulting in the ideal response

$$P_{Ideal}(f) = \begin{cases} \frac{1}{2W}, & -W < f < W \\ 0, & |f| > W, \end{cases} \tag{2.4}$$

where W is the overall system bandwidth defined by

$$W = \frac{R}{2} = \frac{1}{2T}.$$

The special value of the bit rate $R = 2W$ is known as the Nyquist rate, and W is known as the Nyquist bandwidth. Taking the inverse Fourier transform of the desired rectangular function, we obtain the *sinc* function

$$p_{ideal}(t) = \frac{\sin(2\pi\frac{t}{2T})}{2\pi\frac{t}{2T}} = \text{sinc}\left(\frac{t}{T}\right) \tag{2.5}$$

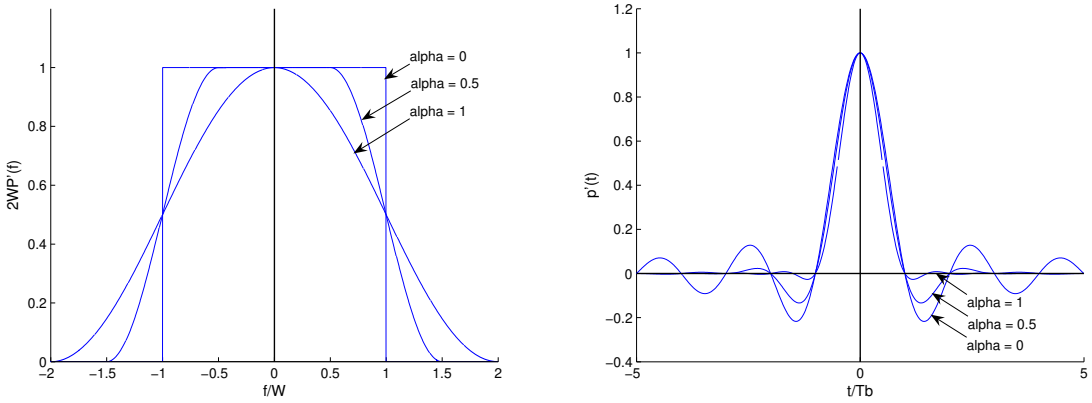
as the ideal pulse shape. However, this is not physically realizable due to the abrupt transition of the rectangular function. A more practical approach is to use a function with a rolloff at the cost of slightly increased bandwidth. To this end, a raised cosine pulse shape is often used and is defined by [8]

$$p_{RC}(t) = \frac{\sin(\pi\frac{\pi t}{T})}{\pi t}\left(\frac{\cos(\pi\alpha t/T)}{1-\left(\frac{4\alpha t}{2T}\right)^2}\right), \tag{2.6}$$

where α is a value between 0 and 1 that governs the amount of roll-off for the pulse shape. Taking the Fourier transform of (2.6) yields

$$P_{RC}(f) = \begin{cases} \frac{1}{2W}, & 0 \le |f| < \frac{(1-\alpha)}{2T} \\ \frac{1}{2}\left[1+\cos\left(\frac{\pi[(2T|f|)-1+\alpha]}{2\alpha}\right)\right], & \frac{(1-\alpha)}{2T} \le |f| \le \frac{(1+\alpha)}{2T} \\ 0, & |f| > \frac{(1+\alpha)}{2T} \end{cases} \tag{2.7}$$

which includes a rolloff portion that has a sinusoidal form. The frequency response of the raised cosine spectrum for different rolloff values is shown in Fig. 2.3(a). It shows that the transitions in the magnitude response are less abrupt than in the ideal case. Therefore, it is easier to approximate the raised cosine filter in comparison to the ideal filter. The time response is shown in Fig. 2.3(b). It shows that the time response decays at a faster rate than the ideal case, which suggests less energy is contained in the tails of the pulse response.



(a) Frequency response. (b) Time response.

Figure 2.3 Response of ideal ($\alpha = 0$) and raised cosine pulse shapes for different rolloff factors, α.

Another useful pulse shape that is related to the raised-cosine pulse shape is the root raised cosine pulse shape. This is given by [22]

$$p_{RRC}(t) = \begin{cases} 1 - \alpha + \frac{4\alpha}{\pi}, & t = 0 \\ \frac{\alpha}{\sqrt{2}}\left[\left(1+\frac{2}{\pi}\right)\sin\left(\frac{\pi}{4\alpha}\right) + \left(1-\frac{2}{\pi}\right)\cos\left(\frac{\pi}{4\alpha}\right)\right], & t = \pm\frac{T}{4\alpha} \\ \frac{\sin\left[\pi(1-\alpha)\frac{t}{T}\right] + \frac{4\alpha t}{T}\cos\left[\pi(1+\alpha)\frac{t}{T}\right]}{\frac{\pi t}{T}\left[1-\left(\frac{4\alpha t}{T}\right)^2\right]}, & \text{otherwise,} \end{cases} \tag{2.8}$$

where the Fourier transform of (2.8) can be found by taking the square root of the right hand side of (2.7), hence the name root raised cosine. When this pulse shape is used at the transmitter for pulse shaping, the receiver often uses another root raised cosine filter as a matched filter, resulting in an overall raised cosine pulse response for the system. We note that both raised cosine and root raised cosine pulse shapes have infinite duration. Therefore, they are usually truncated to a fixed number of symbol intervals.

2.3 CONNECTION BETWEEN FOURIER SERIES AND FOURIER TRANSFORMS

When a system employs frequency-domain processing, it is often necessary to represent signals in the frequency-domain by taking the Fourier Transform of the time-domain

signal. Furthermore, if the signal of interest is periodic, there will be a connection between the Fourier series expansion and the Fourier transform of the signal. This connection is useful as it can be applied to find the Fourier transform of a periodic signal without evaluating complex integrals.

Following the discussion presented in [23], consider a periodic signal $x_p(t)$ with period T and fundamental frequency $f_0 = 1/T$. It is possible to represent $x_p(t)$ as a linear combination of the fundamental frequency f_0 and its harmonics. This is known as the Fourier series expansion of the periodic sequence $x_p(t)$. In mathematical terms, the Fourier series expansion of $x_p(t)$ in exponential form is given by

$$x_p(t) = \sum_{k=-\infty}^{\infty} X_k \exp(j2\pi k f_0 t), \tag{2.9}$$

where the index k ranges from $-\infty$ to ∞ and

$$X_k = \frac{1}{T} \int_T x_p(t) \exp(-j2\pi k f_0 t)\ dt.$$

Note that the coefficient X_{-k} is simply the complex conjugate of X_k.

If we multiply X_k by the period T and let $T \to \infty$, which means $kf_0 \to f$, we obtain the Fourier transform of the infinitely stretched out pulse $x(t)$ corresponding to one period of $x_p(t)$. That is

$$X(f) = \lim_{T\to\infty} TX_k = \int_{-\infty}^{\infty} x(t) \exp(-j2\pi f t)\ dt. \tag{2.10}$$

This provides a frequency-domain representation of the aperiodic signal $x(t)$. The Fourier series coefficients for $x_p(t)$ and the Fourier transform of $x(t)$ therefore have the following relationship

$$X_k = \frac{1}{T} X(f)\Big|_{f=kf_0}. \tag{2.11}$$

2.4 TURBO EQUALIZATION

Multipath propagation and receiver front end noise corrupts the transmitted signal causing errors in the received signal. Equalization is the process of recovering the

original transmitted signal from the received signal. This process can be performed in the time or the frequency-domain, and is carried out by the equalizer shown in Fig. 2.1. Recent work has shown that frequency-domain equalization has advantages over time-domain equalization when the delay spread of the channel extends over many symbols [18]. However, the drawback of frequency-domain equalization is that block oriented processing is required due to the block oriented nature of the discrete Fourier transform (DFT). Hence, it will only work for quasi-static channels, which remain essentially constant over at least one block.

If error correction coding is used, the receiver performance can be improved if we allow information to be iteratively passed between the equalizer and the decoder. This allows both stages to iteratively use information from each other to help make their decisions on the received signal. This is referred to as turbo equalization and was first proposed in [24]. Recently, several turbo equalization schemes have been proposed [25–32]. A time-domain turbo equalization scheme was proposed in [25]. This design requires knowledge of the channel state information (CSI), which is a drawback if the channel is expected to change rapidly (fast fading) or if the CSI is difficult to obtain. Following the work of [25], a frequency-domain turbo linear equalization (FD-TLE) scheme was proposed in [26], which also required knowledge of the CSI. Using the ideas in [26], a new frequency-domain turbo equalization scheme was proposed in [32]. This algorithm was derived using fewer approximations than the scheme in [26]. However, it also assumed perfect CSI was available in the receiver and did not consider any form of channel estimation.

There has been some research into turbo equalizers that do not require a priori knowledge of the CSI [33–35]. One approach is to iteratively minimize a cost function [33, 34]. This approach was employed in [31] to jointly estimate the channel in parallel with time-domain turbo equalization. Its disadvantage is that the variance of the estimation error is not easily obtained from the soft outputs of the soft-input soft-output (SISO) decoder. A similar time-domain scheme was given in [35]. It structured channel estimation as a Kalman state estimator. This is achieved by processing the soft information from the decoding process as part of the statistical description of

the channel. However, it leads to a very complex structure, especially for high-level modulation formats.

The work presented in this thesis also considers the case where perfect CSI is not available at the receiver. Here, the CSI is estimated in the receiver using pilot sequences. The channel is assumed to be quasi-static so that frequency-domain turbo equalization can be applied. Furthermore, the channel is assumed to have a relatively long delay spread, which is well suited to the use of frequency-domain processing. Using the work in [32], the minimum mean squared error (MMSE) linear estimation method is applied in the turbo equalization process.

2.4.1 Minimum Mean Squared Error (MMSE) Linear Estimation

Following the derivation presented in [33], suppose that we have two sequences of random variables $\{x_n\}$ and $\{y_n\}$ that are correlated. We wish to estimate x_t for some particular time t using the observation of $\{y_n\}$ at some set of times $a \leq n \leq b$. Let us consider the linear estimator

$$\widehat{x}_t = \sum_{n=a}^{b} \mathfrak{h}_{t,n}\ y_n + \mathfrak{c}_t, \tag{2.12}$$

where $\widehat{x}_n$ is the estimate of x_n, and $\mathfrak{h}_{t,a}, \ldots, \mathfrak{h}_{t,b}$ and $\mathfrak{c}_t$ are scalar parameters of the estimator. We would like to find the best such estimate in the sense that the mean square error (MSE) is minimized, that is we want

$$\min E\left\{(\widehat{x}_t - x_t)^2\right\} \tag{2.13}$$

for all linear combinations of $\{y_n\}$. The orthogonality principle states that $\widehat{x}_t$ solves the above equation if and only if,

$$E\left\{(\widehat{x}_t - x_t)\ y_n\right\} = 0 \tag{2.14}$$

for all y_n, $a \leq n \leq b$. As a result, for the MMSE solution,

$$\mathfrak{c}_t = E\{x_t\} - \sum_{n=a}^{b} \mathfrak{h}_{t,n}\, E\{y_n\} \tag{2.15}$$

$$\mathfrak{H}_t = \mathfrak{R}_y^{-1}\, \mathfrak{R}_{xy}(t), \tag{2.16}$$

where $\mathfrak{H}_t = [\mathfrak{h}_{t,a}, \ldots, \mathfrak{h}_{t,b}]^T$, $\mathfrak{R}_y$ is the autocovariance matrix of the vector $(y_a, \ldots, y_b)^T$ and $\mathfrak{R}_{xy}(t)$ is the crosscovariance vector between x_t and $\{y_n\}$. Hence, the MMSE estimate of $\widehat{x}_t$ is given by

$$\widehat{x}_t = E\{x_t\} + \sum_{n=a}^{b} \mathfrak{h}_{t,n}\, (y_n - E\{y_n\}). \tag{2.17}$$

2.5 CHANNEL ESTIMATION

Referring to Fig. 2.1, a channel estimator is required in the receiver to estimate the CSI. In general, the channel estimator uses the difference between a local copy of a transmitted training sequence and the received signal as an error signal to estimate the CSI.

The Kalman filter based approach of [35, 36] is a possible candidate for channel estimation. However, in order to avoid the computational complexity of the Kalman approach, we employ the generalized least mean squares (GLMS) estimator of [37] to estimate the channel. It was formulated from its predecessor, the generalized recursive least squares (GRLS) estimator of [38]. Both the GLMS and GRLS estimators are Kalman like in structure, but have the benefit of not requiring a prior knowledge of the channel correlation matrix. This is an advantage when the channel correlation matrix is not easily obtainable. Furthermore, although the GLMS estimator performance has been shown to be worse than the GRLS estimator in [37], it achieves a significant reduction in complexity.

2.5.1 The Generalized Recursive Least Squares (GRLS) Estimator

Following the work of [38], we now discuss the underlying model and the estimation algorithm for the GRLS channel estimator. Then following the work of [37], we will discuss the estimation algorithm for the GLMS channel estimator.

Consider a composite channel impulse response $h(t,\tau) = p(\tau) \otimes c(t,\tau)$, where $p(\tau)$ is the pulse shape and $c(t,\tau)$ is the channel impulse response. Let $h(t,\tau)$ span $L_h = L_p + L_c$ symbol intervals, as a result of $p(\tau)$ and $c(\tau)$ spanning L_p and L_c symbol intervals, respectively. Let $y(t)$ be the received signal due to the transmission of a known training sequence $\{d_i\}$ over the channel $h(t,\tau)$. The received signal $y(t)$ is sampled at a rate of $1/T_s = N_s/T$, where N_s is the number of samples per symbol interval. The i^{th} received sample may be written as

$$y_i = \sum_{k=-L_p/2+\kappa}^{L_p+L_c/2+\kappa} d_k h_{i,i-kN_s} + n_i, \tag{2.18}$$

where $y_i = y(iT_s)$, the channel spans L_h symbol intervals and $\kappa = \lfloor i/N_s \rfloor$. The noise samples n_i are assumed to be Gaussian with zero-mean, and variance σ_n^2.

Following [39], it is possible to decompose the discrete channel impulse response into N_s parallel symbol-rate sub-channel impulse responses. Without loss of generality, the l^{th} received sample of the γ^{th} sub-channel may be written as

$$y_l^{(\gamma)} = \sum_{\mu=-L_p/2+l}^{L_c+L_p/2+l} d_\mu h_{l,l-\mu}^{(\gamma)} + n_l^{(\gamma)}, \tag{2.19}$$

where $y_l^{(\gamma)} = y_{lN_s+\gamma}$, $n_l^{(\gamma)} = n_{lN_s+\gamma}$, and $h_{l,l-\mu}^{(\gamma)} = h_{lN_s+\gamma,(l-\mu)N_s+\gamma}$, for $\gamma = 0,\ldots,N_s-1$.

Provided that the channel is wide sense stationary (WSS), the γ^{th} sub-channel may be modeled as a P_{VAR}^{th} order vector autoregressive (VAR) process [40],

$$\mathcal{H}_l^{(\gamma)} = \sum_{r=1}^{P_{VAR}} \mathcal{A}_r^{(\gamma)} \mathcal{H}_{l-r}^{(\gamma)} + \mathcal{N}_l^{(\gamma)}, \tag{2.20}$$

where $\mathcal{A}_r^{(\gamma)}$ is the r^{th} VAR coefficient matrix,

$$\mathcal{H}_l^{(\gamma)} = [h_{l,-L_p/2}^{(\gamma)}, \ldots, h_{l,0}^{(\gamma)}, \ldots, h_{l,L_p/2+L_c-1}^{(\gamma)}]^T \tag{2.21}$$

and $\mathcal{N}_l^{(\gamma)}$ is the process noise vector. The VAR coefficient matrices may be obtained using the channel statistics and solving matrix-vector Yule-Walker equations [40]. The order of the VAR process is chosen as a trade-off between complexity and accuracy [41].

Observing the VAR process described by (2.20), we see that the VAR coefficient matrices are in fact the coefficient matrices of a length P_{VAR} matrix-vector predictor. An alternative method of obtaining a set of predictor coefficient matrices is to approximate the fading process associated with the μ^{th} channel coefficient of the γ^{th} sub-channel as an N_G^{th} order polynomial series [39]. Following [39], the N_G^{th} order polynomial prediction equation for the μ^{th} coefficient may be expressed as

$$h_{l,\mu}^{(\gamma)} = \sum_{r=1}^{P_G} u_r h_{l-r,\mu}^{(\gamma)} + e_{l,\mu}(N_G, P_G), \tag{2.22}$$

where u_r is the r^{th} tap weight of a polynomial predictor of length P_G and order N_G with $N_G < P_G$ and $e_{l,\mu}^{(\gamma)}(N_G, P_G)$ is the prediction error, which is dependent on the polynomial predictor order N_G and length P_G. This can also be expressed in matrix-vector form as

$$\mathcal{H}_l^{(\gamma)} = \sum_{r=1}^{P_G} \mathcal{U}_r \mathcal{H}_{l-r}^{(\gamma)} + \mathcal{E}_l^{(\gamma)}, \tag{2.23}$$

where the $L_h \times L_h$ polynomial predictor matrix is $\mathcal{U}_r = u_r \mathbf{I}_{L_h}$ and

$$\mathcal{E}_l^{(\gamma)} = [e_{l,1}^{(\gamma)}(N_G, P_G), e_{l,2}^{(\gamma)}(N_G, P_G), \ldots, e_{l,L_h}^{(\gamma)}(N_G, P_G)]^T. \tag{2.24}$$

We note that the computation of the polynomial prediction coefficients $\{u_1, u_2, \ldots, u_r\}$ does not require channel statistics. A list of unique coefficient values [38] for different combinations of N_G and P_G is shown in Table 2.1.

Table 2.1 Polynomial Predictor Coefficients For Different P_G and N_G [37].

Length P_G	Order N_G	Tap Coefficients $\{u_1, u_2, \ldots, u_{P_G}\}$
1	0	{1}
2	0	{1/2, 1/2}
2	1	{2, -1}
3	0	{1/3, 1/3, 1/3}
3	1	{4/3, 1/3, -2/3}
3	2	{3, -3, 1}

Under the assumption that the prediction error $\mathcal{E}_l^{(\gamma)}$ is very small, an alternative state-space model for the γ^{th} sub-channel with unforced dynamics may then be formulated as

$$\mathbf{h}_{l+1}^{(\gamma)} = \mathbf{U}\mathbf{h}_l^{(\gamma)}, \tag{2.25}$$

where $\mathbf{h}_l^{(\gamma)} = [\mathcal{H}_l^{(\gamma)T}, \mathcal{H}_{l-1}^{(\gamma)T}, \ldots, \mathcal{H}_{l-P_G+1}^{(\gamma)T}]^T$ and

$$\mathbf{U} = \left(\begin{array}{cccc|c} \mathcal{U}_1 & \mathcal{U}_2 & \ldots & \mathcal{U}_{P_G-1} & \mathcal{U}_{P_G} \\ \hline & & \mathbf{I}_{L_h(P_G-1)} & & \mathbf{0}_{L_h(P_G-1),L_h} \end{array} \right). \tag{2.26}$$

Using the state-space model of (2.25), the GRLS algorithm may be summarized as [38]

$$\mathbf{h}_{l|l-1}^{(\gamma)} = \mathbf{U}\mathbf{h}_{l-1|l-1}^{(\gamma)} \tag{2.27}$$

$$\mathbf{P}_{l|l-1}^{(\gamma)} = \lambda^{-1}\mathbf{U}\mathbf{P}_{l-1|l-1}^{(\gamma)}\mathbf{U}^H \tag{2.28}$$

$$\mathbf{h}_{l|l}^{(\gamma)} = \mathbf{h}_{l|l-1}^{(\gamma)} + \mathbf{k}_l^{(\gamma)}\left(y_l^{(\gamma)} - \mathbf{d}_l\mathbf{h}_{l|l-1}^{(\gamma)}\right) \tag{2.29}$$

$$\mathbf{P}_{l|l}^{(\gamma)} = (\mathbf{I}_{L_hP_G} - \mathbf{k}_l^{(\gamma)}\mathbf{d}_l)\mathbf{P}_{l|l-1}^{(\gamma)} \tag{2.30}$$

$$\mathbf{k}_l^{(\gamma)} = \frac{\mathbf{P}_{l|l-1}^{(\gamma)}\mathbf{d}_l^H}{1 + \mathbf{d}_l\mathbf{P}_{l|l-1}^{(\gamma)}\mathbf{d}_l^H}, \tag{2.31}$$

where λ is a number less than unity, $\mathbf{P}_{l|l}^{(\gamma)}$ is the inverse of the channel correlation matrix and $\mathbf{k}_l^{(\gamma)}$ is the Kalman gain vector. The l^{th} data vector is defined as

$$\mathbf{d}_l = [d_{l+L_c+L_p/2}, \ldots, d_{l+1}, d_l, d_{l-1}, \ldots, d_{l-L_p/2} \mid \mathbf{0}_{1,L_h(P_G-1)}]. \tag{2.32}$$

The algorithm is initialized by setting the estimated channel state vector to the null vector and $\mathbf{P}_{0|-1}^{(\gamma)} = \zeta^{-1}\mathbf{I}_{L_hP_G}$, where ζ is a small positive real number.

2.5.2 The Generalized Least Mean Squares (GLMS) Estimator

The GLMS algorithm is a reduced complexity version of the GRLS algorithm. It is derived by replacing the instantaneous inverse channel correlation matrix $\mathbf{P}_{l|l}^{(\gamma)}$ with the matrix $\widehat{\mathbf{P}}^{(\gamma)} = (\lim_{l\to\infty} E[\mathbf{P}_{l|l}^{(\gamma)}])^{-1}$, which is the inverse of the steady-state mean of the correlation matrix. In order to evaluate $\widehat{\mathbf{P}}^{(\gamma)}$, we require knowledge of the autocorrelation matrix of the training data vector $\mathbf{R}_d = E[\mathbf{d}_l^H\mathbf{d}_l]$, the state transition matrix $\mathbf{U}$ and the value of λ. These are known a priori, so we may evaluate $\widehat{\mathbf{P}}^{(\gamma)}$ using the offline recursion

$$\widehat{\mathbf{P}}_{l|l-1}^{(\gamma)} = \lambda^{-1}\mathbf{U}\widehat{\mathbf{P}}_{l-1|l-1}^{(\gamma)}\mathbf{U}^H \tag{2.33}$$

$$\mathbf{\Psi}_{l|l-1}^{(\gamma)} = (\mathbf{I}_{L_hP_G} + \mathcal{D}\widehat{\mathbf{P}}_{l|l-1}^{(\gamma)}\mathcal{D}^H)^{-1} \tag{2.34}$$

$$\mathbf{P}_{l|l}^{(\gamma)} = \widehat{\mathbf{P}}_{l|l-1}^{(\gamma)} - \widehat{\mathbf{P}}_{l|l-1}^{(\gamma)}\mathcal{D}^H\mathbf{\Psi}_{l|l-1}^{(\gamma)}\mathcal{D}\widehat{\mathbf{P}}_{l|l-1}^{(\gamma)}, \tag{2.35}$$

where $\mathcal{D}$ is a $(L_hP_G \times L_hP_G)$ matrix, such that $\mathbf{R}_d = E[\mathbf{d}_l^H\mathbf{d}_l] = \mathcal{D}^H\mathcal{D}$, and $\widehat{\mathbf{P}}_{0|-1}^{(\gamma)} = \mathbf{P}_{0|-1}^{(\gamma)} = \zeta_{-1}\mathbf{I}_{L_hP_G}$. The steady-state approximation of the inverse correlation matrix is therefore $\widehat{\mathbf{P}}^{(\gamma)} = \widehat{\mathbf{P}}_{l|l}^{(\gamma)}$ for large values of l. The number of recursions required to obtain a good approximation was not mentioned in [37]. However, it has been found by means of experimentation that no more than 500 iterations were sufficient for the channels employed in this thesis work.

Recognizing that $\mathbf{k}_l^{(\gamma)} = \mathbf{P}_{l|l}^{(\gamma)}\mathbf{d}_l^H$ [37] and replacing $\mathbf{P}_{l|l}^{(\gamma)}$ by $\widehat{\mathbf{P}}^{(\gamma)}$, we may write the l^{th} update equation of the GLMS algorithm as

$$\mathbf{h}_{l+1|l}^{(\gamma)} = \mathbf{U}\left(\mathbf{h}_{l|l-1}^{(\gamma)} + \widehat{\mathbf{P}}^{(\gamma)}\mathbf{d}_l^H\left(y_l^{(\gamma)} - \widehat{y}_{l|l-1}^{(\gamma)}\right)\right). \tag{2.36}$$

Here, We note that the special case when $P_G = 1$ and $N_G = 1$, (2.36) reduces to the standard least mean squares (LMS) algorithm.

2.6 ERROR CORRECTION CODING

As already discussed, equalization combined with channel estimation can allow the receiver to remove some of the channel impairments. To further improve the performance, error correction coding is employed to help correct the errors that exist after the equalization process, as shown in Fig. 2.1. In this thesis, binary convolutional codes [42] will be used.

2.6.1 Binary Convolutional Codes

A binary convolutional code is generated by passing the message sequence through a linear finite-state register (or finite-state machine). The redundant bits are generated using modulo-2 convolutions defined by generator equations which describe the connections to the modulo-2 adders [21]. Here we assume the convolutional encoder structure has a single input as this is sufficient for the work presented in this thesis. This is illustrated in Fig. 2.4.

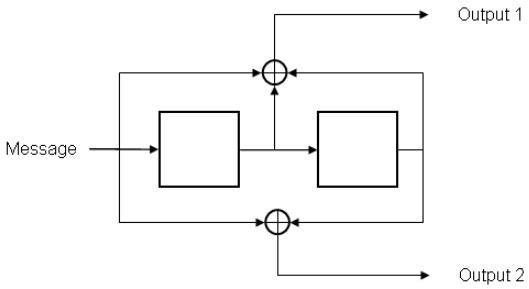


Figure 2.4 Encoder for the rate $1/2$ $(5, 7)$ convolutional code.

In general, let $\mathcal{M}$ be the memory or number of delay elements within the register, and $\mathcal{V}$ be the number of modulo-2 adders. If we wish to encode a message sequence $\mathbf{m}$ of length $\mathcal{L}$, the resulting codeword sequence $\mathbf{c}$ will have length $\mathcal{V}(\mathcal{L}+\mathcal{M})$. Hence, we may express the code rate $\mathcal{R}$ as

$$\mathcal{R} = \frac{\mathcal{L}}{\mathcal{V}(\mathcal{L}+\mathcal{M})}. \tag{2.37}$$

If the length of the message sequence $\mathcal{L}$ is significantly longer than the number of delay elements $\mathcal{M}$ (which is typically the case) then $\mathcal{L}+\mathcal{M} \simeq \mathcal{L}$ and we may approximate

the code rate by

$$\mathcal{R} = \frac{1}{\mathcal{V}}. \tag{2.38}$$

The constraint length $\mathcal{K}$ of a convolutional code refers to the number of shifts required for an input message bit to be shifted out of the register so that it will not have any more influence on the encoder output. Clearly, $\mathcal{M}+1$ shifts are required for an encoder with $\mathcal{M}$ delay elements. Therefore, $\mathcal{K} = \mathcal{M}+1$. An example of a 1/2 rate convolutional code with 2 delay elements and a constraint length of 3 is shown in Fig. 2.4. The corresponding generator polynomial equations for the top and bottom outputs are $g(D) = 1 + D + D^2$ and $g(D) = 1 + D^2$, respectively. A short hand way of defining the generator equations is to take the coefficients of the generator polynomials and express them as octal numbers. For example, $g(D) = 1 + D + D^2$ may be expressed as 111(binary) = 7(octal) and $g(D) = 1 + D^2$ may be expressed as 101(binary) = 5(octal). The code can therefore be referred to as the $(5, 7)$ convolutional code.

The structural properties of a convolutional encoder can be portrayed using a trellis diagram. We define the state of a convolutional encoder according to the $\mathcal{M}$ message bits stored in the shift register. For the convolutional encoder shown in Fig. 2.4, where $\mathcal{M} = 2$, there are $2^{\mathcal{M}} = 2^2 = 4$ possible states. If we start and finish in the all-zero (00) state, the trellis diagram is shown in Fig. 2.5. The solid and dashed lines represent inputs of 0 and 1, respectively. The branches, or connections between states, are labeled with the corresponding output bits.

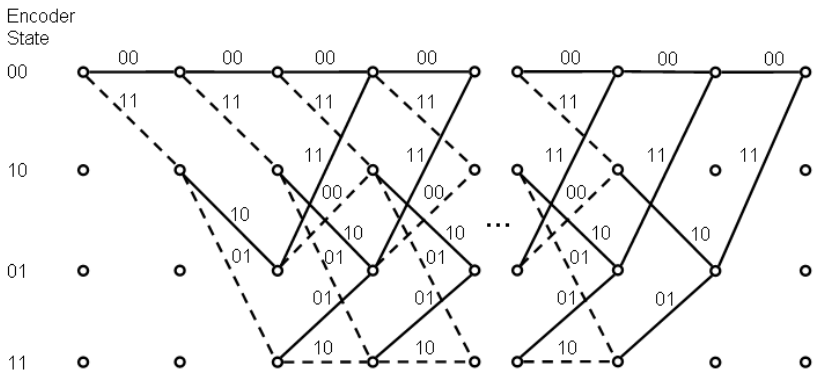


Figure 2.5 Trellis diagram for the 1/2 rate convolutional encoder.

The performance of a convolutional code is closely related to its minimum free distance, d_{free}. This is defined as the Hamming distance between the all-zero codeword associated with the all-zero path, and the codeword associated with the path with the lowest weight starting and ending at the all-zero state, within the code trellis [43]. Using Fig. 2.5 as an example, the all-zero path and the lowest weight path in the trellis are shown in Fig. 2.6. The corresponding codeword outputs are 000000 and 111011. Therefore, the minimum free distance is $d_{free} = 5$. In general, a convolutional code with a minimum free distance of d_{free} can correct up to t errors that occur within a length of $\mathcal{K}$ if and only if d_{free} is greater than $2t$ [8, 43]. Therefore, the minimum free distance provides a good indication of the error correction capability of the convolutional code.

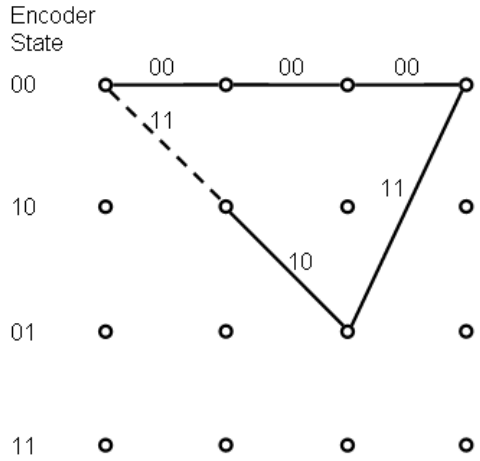


Figure 2.6 Minimum weight path starting and ending with the all-zero state.

Table 2.2 lists some rate 1/2 convolutional codes with maximum d_{free} for a given constraint length (they were obtained using computer search methods) [21]. It is worth noting that the (131,171) code is a standard code used in various real world applications [44] and has a stronger error correction capability than the $(5, 7)$ convolutional code. The corresponding encoder structure for this code is shown in Fig. 2.7.

Table 2.2 Some rate 1/2 maximum free distance convolutional codes [21].

Constraint Length $\mathcal{K}$	Generator Equations (octal)		Minimum Free Distance d_{free}
3	5	7	5
4	15	17	6
5	23	35	7
6	53	75	8
7	133	171	10
8	247	371	10
9	561	753	12
10	1167	1545	12
11	2335	3661	14
12	4335	5723	15
13	10533	17661	16
14	21675	27123	16

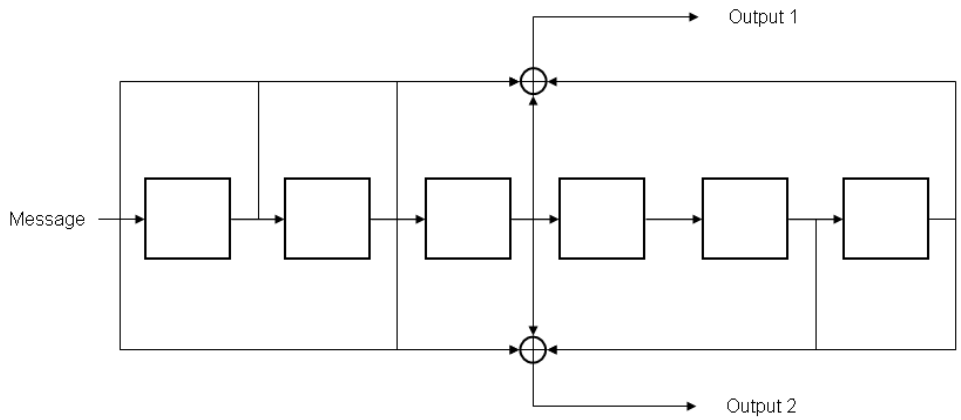


Figure 2.7 Encoder for the rate 1/2 $(133, 171)$ convolutional code.

2.6.2 Decoding Convolutional Codes

Convolutional codes may be decoded using maximum-likelihood (ML) decoding or maximum *a posteriori* probability (MAP) decoding. For ML decoding, consider the $(5, 7)$ convolutional code and the QPSK signal constellation. Let $\mathbf{c}$, $\mathbf{x}$ and $\mathbf{y}$ denote the encoded bits, the symbol sequence obtained from mapping the encoded codeword bits to the signal constellation, and input to the convolutional decoder, respectively, as shown in Fig. 2.1. Each element x_i of the symbol sequence $\mathbf{x}$ corresponds to a point in the signal constellation and each element y_i of the sequence $\mathbf{y}$ corresponds to the received version of x_i. Clearly, y_i will contain errors due to the channel, receiver front end noise, and the equalization process through transmission. Denoting the error as a random variable $\mathfrak{E}$, we may write $y_i = x_i + \mathfrak{e}_i$, where $\mathfrak{e}_i$ is a sample of $\mathfrak{E}$. Often $\mathfrak{E}$ may be assumed to be Gaussian distributed with zero-mean and variance $\sigma^2_{\mathfrak{E}}$.

The goal of ML sequence estimation is to find the symbol sequence $\widehat{\mathbf{x}}_{opt}$, and equivalently $\widehat{\mathbf{c}}_{opt}$, for which the *a posteriori* probability $\mathcal{P}(\mathbf{x}|\mathbf{y})$ is maximized. That is,

$$\widehat{\mathbf{x}}_{opt} = \arg\left(\max_{\mathbf{x}\in\mathcal{C}} \mathcal{P}(\mathbf{x}|\mathbf{y})\right), \tag{2.39}$$

where $\mathcal{C}$ is the set containing symbol sequences associated with every possible codeword. Applying Bayes' theorem[1] to $\mathcal{P}(\mathbf{x}|\mathbf{y})$, we obtain

$$\mathcal{P}(\mathbf{x}|\mathbf{y}) = \frac{\mathcal{P}(\mathbf{x})\mathcal{P}(\mathbf{y}|\mathbf{x})}{\mathcal{P}(\mathbf{y})}. \tag{2.40}$$

Realizing that $\mathcal{P}(\mathbf{y})$ is invariant to the information sequence $\mathbf{x}$, (2.39) is equivalent to

$$\widehat{\mathbf{x}}_{opt} = \arg\left(\max_{\mathbf{x}\in\mathcal{C}} \mathcal{P}(\mathbf{x})\mathcal{P}(\mathbf{y}|\mathbf{x})\right). \tag{2.41}$$

Furthermore, it is often assumed that all sequences $\mathbf{x} \in \mathcal{C}$ are equally likely, so $\mathcal{P}(\mathbf{x})$ is a constant, meaning

$$\widehat{\mathbf{x}}_{opt} = \arg\left(\max_{\mathbf{x}\in\mathcal{C}} \mathcal{P}(\mathbf{y}|\mathbf{x})\right), \tag{2.42}$$

[1] Bayes' theorem is derived from the relationship between the conditional and joint probability of events A and B as follows: $\mathcal{P}(A|B) \cdot \mathcal{P}(B) = \mathcal{P}(B|A) \cdot \mathcal{P}(A) = \mathcal{P}(A, B)$. It may also be written in the alternative form as $\mathcal{P}(A|B) = \frac{\mathcal{P}(B|A)\cdot\mathcal{P}(A)}{\mathcal{P}(B)}$ [45].

which defines maximum likelihood detection.

A method to calculate (2.42) is to perform a step-by-step calculation. First, $\ln \mathcal{P}(y_i|\mathcal{X}_m)$ is calculated for all constellation points, $\mathcal{X}_m$. Second, we associate the encoder output for each state transition to the corresponding $\ln \mathcal{P}(y_i|\mathcal{X}_m)$ as a branch metric. Third, the path metric of a symbol sequence up to time $i+1$ is defined as the sum of the path metric for a symbol sequence up to time i and the branch metric of the i^{th} state transition. The initial path metric, corresponding to $i = 0$, is set to 0. Hence, the path metric of a symbol sequence $\mathbf{x}$ is obtained as the sum of all branch metrics, and the symbol sequence with the largest metric is chosen as $\widehat{\mathbf{x}}_{opt}$. When the error introduced to $\mathbf{y}$ is Gaussian, $\ln \mathcal{P}(y_i|\mathcal{X}_m)$ can be replaced by the Euclidean distance between y_i and $\mathcal{X}_m$.

Although the step-by-step calculation method can be used to decode convolutional codes, it leads to an exponentially increasing number of code sequences and metrics as the length of the codeword increases [46]. Since the structure of the trellis associated with a convolutional code repeats itself, substantial simplification of the decoding procedure can be achieved by considering only the code sequence with the largest metric over all other code sequences with $n > \mathcal{K}$ trace backs that go through the same node. This is known as the Viterbi algorithm (VA), and was first developed in [47]. The VA was later shown to be optimal, as the simplification to the step-by-step approach does not effect the selection of the best possible path through the trellis [48].

For the work of this thesis, the MAP decoding method known as the Bahl-Cocke-Jelinek-Reviv (BCJR) algorithm is used instead of the VA. Although both algorithms employ the trellis structure of convolutional codes [49], the BCJR algorithm is capable of processing a soft input as well as providing soft output information, which is required in the turbo equalization process. The BCJR algorithm is based on the assumption that a convolutional encoder may be modeled as a discrete-time finite-state Markov source, and in comparison to the VA, it maximizes the symbol *a posteriori* probabilities instead of the sequence *a posteriori* probabilities. A detailed discussion of the BCJR algorithm is presented in the next chapter as it is related closely to the turbo equalization structure.

2.7 SUMMARY

This background chapter summarized some of the underlying theory behind a typical wireless communication system. These include modulation and pulse shaping techniques, Fourier series and Fourier transform representation for periodic signals, equalization, channel estimation and error correction coding. The signal constellations and the coding scheme employed in this thesis were also defined in this chapter. However, the proposed FD-TLE system incorporating channel estimation has not yet been defined. It will be presented in the next chapter, where a detailed discussion of the proposed system and examples are given.

Chapter 3

FREQUENCY-DOMAIN TURBO LINEAR EQUALIZATION WITH CHANNEL ESTIMATION

This chapter describes the proposed system. It uses the frequency-domain turbo linear equalization (FD-TLE) algorithm of [32] and the channel estimation algorithm of [37]. A block diagram of the proposed system at complex baseband is shown in Fig. 3.1. The novelty of the scheme lies in the combination of channel estimation with FD-TLE, which has not been considered to date. Furthermore, we consider the extension to high level modulation formats.

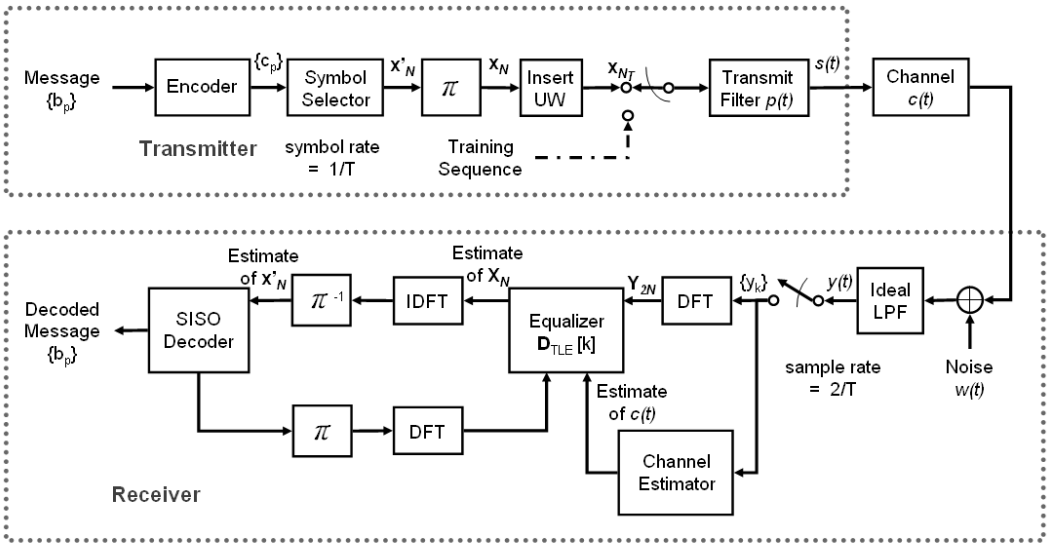


Figure 3.1 Proposed system using FD-TLE with channel estimation.

3.1 SYSTEM OVERVIEW

We may abstract from the system diagram of Fig. 3.1, the signal processing operations that take place in terms of a flow chart as shown in Fig. 3.2. Prior to data transmission, the system is switched to training mode and a training sequence is transmitted to train the channel estimator. The training sequence must be sufficiently long that the estimator essentially converges. The system is then switched to data transmission mode with the training sequence embedded in the unique word (UW) at the start of each frame (c.f. Fig. 3.3). This allows the channel estimator to continue to track and estimate the CSI for each data transmission block.

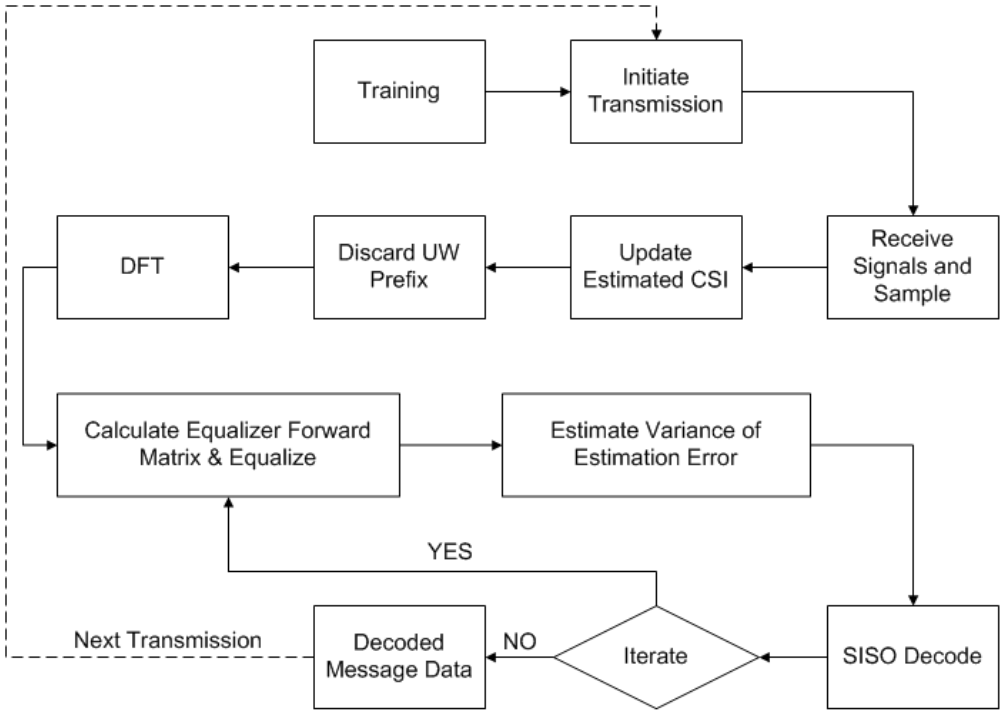


Figure 3.2 Overall proposed system operation flowchart.

At the transmitter, the binary data stream $\{b_p\}$ is encoded using a convolutional code[1]. The encoded bits $\{c_p\}$ are fed to a symbol selector, which maps each group of $\log_2 M$ consecutive bits to a channel symbol from an M-ary constellation set $\mathcal{X} = \{\mathcal{X}_i, \quad i = 0, 1, \ldots, M-1\}$. The signal constellations considered were defined in Chapter 2. The channel symbol stream at rate $f = 1/T$ is divided into non-overlapping

[1] A variety of codes such as turbo and/or low density parity check (LDPC) codes could also be used, but the convolutional code is chosen for simplicity.

blocks each containing N_{CS} channel symbols. Each block is interleaved using a block interleaver to produce the block of channel symbols

$$\mathbf{x}_{N_{CS}} = [x_0, x_1, \ldots, x_{N_{CS}-1}]^T. \tag{3.1}$$

It is assumed that the channel symbols $\{x_n\}$ are zero-mean, independent and identically distributed with variance σ_x^2. A known UW sequence of length N_{UW} is appended to each interleaved channel symbol block both as a prefix and as a suffix. This produces the sequence

$$\mathbf{x}_{N_{Total}} = [x_{-N_{UW}}, \ldots, x_{-1}, x_0, x_1, \ldots, x_{N_{CS}-1}, x_{N_{CS}}, \ldots, , x_{N_{CS}+N_{UW}-1}]^T, \tag{3.2}$$

where $N_{Total} = 2N_{UW} + N_{CS}$. The purpose of the UW prefix and suffix is to make the convolution with the channel appear circular to the DFT [50]. The UW prefix and suffix are also used to update the CSI in the channel estimator and to estimate the variance of the equalization error for each frame of equalized received signals. The composition of each transmission frame is summarized in Fig. 3.3 and the percentage overhead $\mathcal{O}$ required for the transmission of each block can be calculated to be

$$\mathcal{O} = \frac{2 \cdot N_{UW}}{N_{Total}} \times 100\%. \tag{3.3}$$

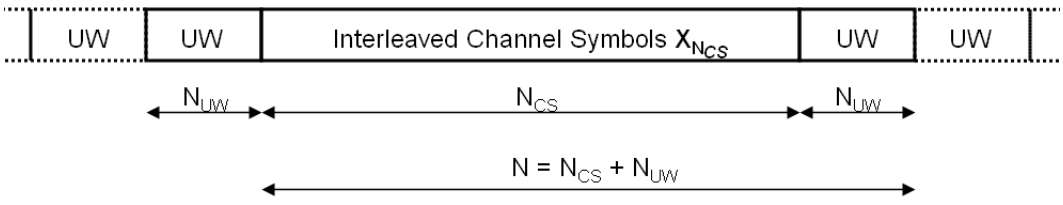


Figure 3.3 The composition of a single transmission frame.

The symbol sequence is passed through a root raised cosine transmitter filter with a rolloff of α and impulse response $p(t)$ truncated to L_p symbol intervals. Here we assume that L_p is large enough so that the change in the filter characteristic due to

the truncation process is negligible. This produces the transmitted complex baseband signal

$$s(t) = \sum_{n=-N_{UW}}^{N_{UW}+N_{CS}-1} x_n p(t - nT), \tag{3.4}$$

which is transmitted over a frequency selective fading channel.

The channel is modeled as a delay line with L_c independent and symbol spaced taps. It is assumed that the channel impulse response which lasts L_c symbol intervals is slowly time varying, and undergoes negligible change over each transmitted block. Then, during the transmission of $\mathbf{x}_{N_{Total}}$, the channel impulse response can be expressed as

$$c(t, \tau) = \sum_{n=0}^{L_c-1} c_n(t)\delta(\tau - \tau_n), \tag{3.5}$$

where $\delta(\cdot)$ is the dirac delta function, and τ_n and $c_n(t)$ are the delay and gain associated with the n^{th} tap at time t, respectively. Since the analysis presented in this section of the chapter focuses on the channel impulse response corresponding to a single data block, we may simplify the notation for the channel impulse response to

$$c(\tau) = \sum_{n=0}^{L_c-1} c_n\delta(\tau - \tau_n), \tag{3.6}$$

assuming the channel to be time-invariant over each block.

The signal $\widetilde{s}(t)$ at the output of the fading channel has the same form as $s(t)$ and is given by

$$\widetilde{s}(t) = \sum_{n=-N_{UW}}^{N_{UW}+N_{CS}-1} x_n h(t - nT), \tag{3.7}$$

where $h(t) = p(t) \otimes c(t)$ is the overall impulse response that spans $L_h = L_p + L_c$ symbols, as illustrated in Fig. 3.4, assuming that the duration of the transmitted signal prefix is set to be longer than the overall channel memory. Hence, $h(t) = 0$ outside the interval $[0, N_{UW}T]$. It is not difficult to see that (3.7) can form a periodic extension based on the interval $0 \leq t \leq (N_{CS} + N_{UW} - 1)T$ of duration NT, where $N = N_{CS} + N_{UW}$ defines its period. The fundamental frequency for this periodic extension is therefore $f_0 = 1/NT$.

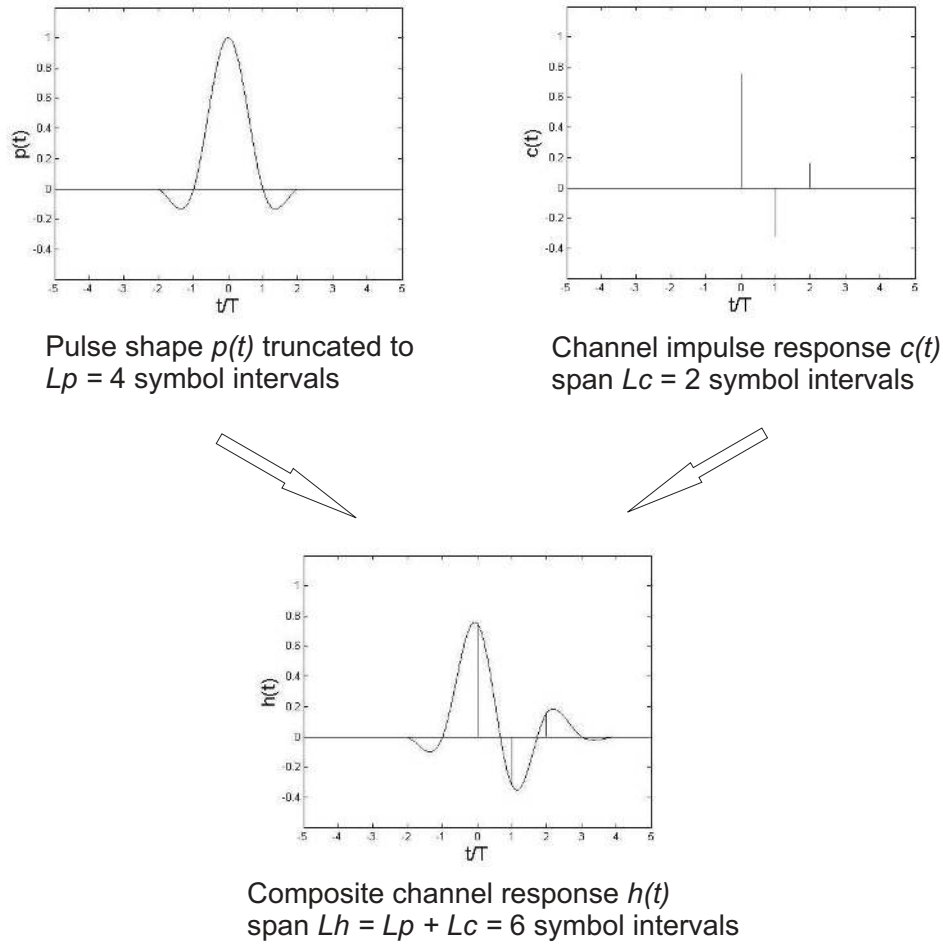


Figure 3.4 Example to show the composite channel response $h(t)$ spanning, for example, $L_h = L_p + L_c = 6$ symbol intervals.

We may express the portion of $\widetilde{s}(t)$ that is used for the periodic extension, in terms of its Fourier series expansion representation as

$$\widetilde{s}_p(t) = \sum_{k=-\infty}^{+\infty} \widetilde{S}_k \exp\left(j\frac{2\pi kt}{NT}\right), \tag{3.8}$$

where $\widetilde{S}_k$ are the series coefficients. Let $\widetilde{S}_p(f)$ be the continuous Fourier transform of $\widetilde{s}_p(t)$. By utilizing the connection between Fourier Series and Fourier Transforms, we

obtain

$$\begin{aligned}
\widetilde{s}_p(t) &= \frac{1}{NT}\sum_{k=-\infty}^{+\infty}\widetilde{S}_p\left(\frac{k}{NT}\right)\exp\left(j\frac{2\pi kt}{NT}\right) \\
&= \frac{1}{NT}\sum_{k=-\infty}^{+\infty}P\left(\frac{k}{NT}\right)X\left(\frac{k}{NT}\right)H\left(\frac{k}{NT}\right)\exp\left(j\frac{2\pi kt}{NT}\right) \\
&= \frac{1}{\sqrt{NT}}\sum_{k=-\infty}^{+\infty}\frac{P\left(\frac{k}{NT}\right)}{\sqrt{T}}\frac{X\left(\frac{k}{NT}\right)}{\sqrt{N}}H\left(\frac{k}{NT}\right)\exp\left(j\frac{2\pi kt}{NT}\right) \\
&= \frac{1}{\sqrt{NT}}\sum_{k=-\infty}^{+\infty}P_kX_kC_k\exp\left(j\frac{2\pi kt}{NT}\right) \qquad (3.9)
\end{aligned}$$

for $0 \le t \le NT$, where $P_k = P(k/NT)/\sqrt{T}$,

$$C_k = C(k/NT) = \sum_{n=0}^{L_c-1}c_n\exp\left(-j\frac{2\pi k\tau_n}{NT}\right) \qquad (3.10)$$

and $P(f)$ and $C(f)$ are the continuous Fourier transforms of $p(t)$ and $c(t)$. In addition, X_k is the k^{th} component of the vector $\mathbf{X}_N = [X_0, X_1, \ldots, X_{N-1}]^T$, resulting from the DFT of $\mathbf{x}_N = [x_0, x_1, \ldots, x_{N-1}]^T$, that is

$$\mathbf{X}_N = \mathbf{Q}_N\mathbf{x}_N, \qquad (3.11)$$

where $\mathbf{Q}_N$ is the N-point DFT matrix, with value $q_{n,k} = \exp(-j2\pi kn/N)/\sqrt{N}$, in the n^{th} row and k^{th} column for $n = 0, 1, \ldots, N-1$ and $k = 0, 1, \ldots, N-1$. Note that (3.9) agrees with the result in [32], despite possible differences in derivation methods. Assuming that $p(t)$ is bandlimited to $B = 1/T$Hz, (3.9) reduces to

$$\widetilde{s}_p(t) = \frac{1}{\sqrt{NT}}\sum_{k=-N}^{N}P_kX_kC_k\exp\left(j\frac{2\pi kt}{NT}\right) \qquad (3.12)$$

for $0 \le t \le NT$, since $P_k = 0$ for $|k| > N$.

The received signal can be expressed as

$$r(t) = \sum_{n=-N_{UW}}^{N_{UW}+N_{CS}-1}x_nh(t-nT) + w(t), \qquad (3.13)$$

where $w(t)$ is additive white Gaussian noise (AWGN) with two-sided power spectral density $2N_0$, as shown in Fig 3.5a. The signal $r(t)$ is passed through an ideal lowpass filter (LPF) having bandwidth $1/T$ and gain $\sqrt{T/2}$, which produces the bandlimited random process

$$y(t) = \sum_{n=-N_{UW}}^{N_{UW}+N_{CS}-1} \sqrt{\frac{T}{2}}\, x_n h(t-nT) + n(t), \tag{3.14}$$

where $n(t)$ is a Gaussian noise process having zero-mean and two-sided power spectral density N_0T for $|f| \leq 1/T$, as shown in Fig 3.5b. An ideal LPF is chosen instead of a matched filter because the received pulse shape is unknown since the channel is random and unknown.

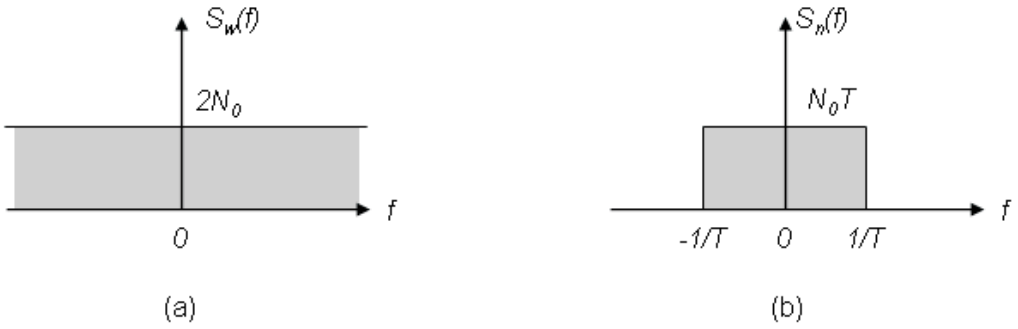


Figure 3.5 Noise PSD for (a) $w(t)$ and (b) $n(t)$.

The signal $y(t)$ is sampled uniformly at rate $f_s = N_s/T$. Assuming $N_s = 2$, two samples are obtained in each symbol period[2]. The samples corresponding to the UW prefix are used by the channel estimator to update the CSI before being discarded. Further explanation of how the UW prefix is used to update the CSI will be presented later in this chapter. Following (3.12) and (3.14), the remaining samples may be expressed

[2]Although N_s can take any integer value to accommodate Doppler, in most practical instances, $N_s = 2$ is sufficient.

as a $2N$-dimensional vector given by $\mathbf{y}_{2N} = [y_0, y_1, \ldots, y_{2N-1}]$, where

$$\begin{aligned} y_i &= y(t_i) = \sum_{n=-N_{UW}}^{N_{UW}+N_{CS}-1} \sqrt{\frac{T}{2}}\, x_n h(t_i - nT) + n_i \\ &= \sqrt{\frac{2}{T}}\, \widetilde{s}_p(t_i) + n_i \\ &= \frac{1}{\sqrt{2N}} \sum_{k=-N}^{N} P_k X_k C_k \exp\left(j\frac{2\pi ki}{2N}\right) + n_i, \end{aligned} \tag{3.15}$$

$t_i = iT/2$ and $i = 0, \ldots, 2N-1$. The noise samples $n_i = n(t_i)$ are independent and identically distributed random variables having zero-mean and variance $\sigma_n^2 = 2N_0$. We note that since σ_n^2 does not undergo dramatic fluctuation, it is assumed to be known at the receiver. This can be achieved by measuring the noise power explicitly.

A $2N$ point DFT is performed on the vector $\mathbf{y}_{2N}$, producing the vector

$$\mathbf{Y}_{2N} = [Y_0, Y_1, \ldots, Y_{2N-1}]^T = \mathbf{Q}_{2N}\mathbf{y}_{2N}, \tag{3.16}$$

whose k^{th} component is given by

$$Y_k = \frac{1}{\sqrt{2N}} \sum_{i=0}^{2N-1} y_i \exp\left(-j\frac{2\pi ki}{2N}\right) \tag{3.17}$$

for $k = 0, \ldots, 2N-1$. It can then be shown that

$$Y_k = R_k X_k + V_k, \tag{3.18}$$

where

$$R_k = \begin{cases} P_k C_k, & k = 0, 1, \ldots, N-1 \\ P_{k-2N} C_{k-2N}, & k = N, N+1, \ldots, 2N-1 \end{cases} \tag{3.19}$$

and the set of random variables $\{V_k\}$ is statistically equivalent to $\{n_i\}$. Furthermore, we can write (3.18) in matrix form as

$$\mathbf{Y} = \mathbf{R}\mathbf{X}_{2N} + \mathbf{V}, \tag{3.20}$$

where $\mathbf{R} = diag(R_k)$ is the composite channel matrix in the frequency domain, $\mathbf{X}_{2N} = \left[\mathbf{X}_N^T|\mathbf{X}_N^T\right]^T$ and $\mathbf{V} = [V_0, V_1, \ldots, V_{2N-1}]^T$.

The matrix form $\mathbf{Y}$ of received samples in the frequency-domain as defined in (3.20) forms the input to the minimum mean-squared error (MMSE) equalizer. Filtering $\mathbf{Y}$ by an $N \times 2N$ equalizer filter matrix $\mathbf{D}_{TLE}[\psi]\big|_{\psi=1}$ produces the N-dimensional frequency-domain estimate vector $\widehat{\mathbf{X}}_N$. Following the IDFT and the deinterleaving process, the time-domain symbol estimate $\widehat{\mathbf{x}}_N^{'}$ is obtained. Passing $\widehat{\mathbf{x}}_N^{'}$ into the soft-input soft-output (SISO) decoder then generates symbol *a posteriori* probabilities (APPs). These are passed back to the equalizer through an interleaver and DFT block to make an updated estimate of $\mathbf{X}_N$ by adjusting the equalizer matrix to $\mathbf{D}_{TLE}[\psi]\big|_{\psi=2}$. This iterative process continues for a set number of iterations, then the decoded data is obtained from the SISO decoder. A detailed discussion of this process will be presented later in this chapter.

3.2 CHANNEL ESTIMATION EMPLOYING THE GLMS CHANNEL ESTIMATOR

As mentioned in the previous section, the UW sequence is used to train the channel estimator and update the channel estimate from one transmission block to another. Here we discuss the channel estimation process in detail, and show how the GLMS channel estimation algorithm is employed.

We begin by considering the design of the UW sequence. First, the length of the UW, N_{UW}, is set to be longer than the maximum delay spread of the overall composite channel response. That is, we choose $N_{UW} \geq L_h$. This is required not only to ensure that the transmitted data block appears circular to the DFT block as mentioned earlier, but also to ensure that the channel is estimated correctly. Second, each element of the UW is a constellation point randomly chosen from the signal constellation of interest. By making the UW sequence as random as possible, we ensure that the spectrum of the UW sequence is as flat as possible, which makes the channel estimation process more effective.

Next we consider the channel estimation process. Prior to training, an offline

recursion is required for the GLMS estimator, as described in Chapter 2, to calculate the so-called intermediate matrix $\widehat{\mathbf{P}}^{(\gamma)}$. This needs to be performed only once, and the result can be stored for use during channel estimation. During training, a training sequence consisting of $N_{Train}+L_h$ symbols is transmitted to train the channel estimator, where N_{Train} corresponds to the number of estimate updates performed during the training phase. This training sequence is denoted by

$$\mathbf{d}_{Train} = [d_{0-L_h/2}, \ldots, d_0, \ldots, d_{N_{Train}}, \ldots d_{N_{Train}+L_h/2-1}]. \tag{3.21}$$

We note that the addition of L_h extra symbols in the training sequence is required to take into account the tail effects arising in linear convolution. As the system samples the received signals at a rate of $2/T$, the i^{th} received sample may be written as

$$y_i = \sum_{k=-L_p/2+\kappa}^{L_c+L_p/2+\kappa} d_k h_{i,i-2k} + n_i. \tag{3.22}$$

By decomposing the channel impulse response into two symbol-rate sub-channel impulse responses, the l^{th} received sample of the γ^{th} sub-channel may be written as

$$y_l^{(\gamma)} = \sum_{\mu=-L_p/2+l}^{L_c+L_p/2+l} d_\mu h_{l,l-\mu}^{(\gamma)} + n_l^{(\gamma)}, \tag{3.23}$$

where $y_l^{(\gamma)}$, $n_l^{(\gamma)}$, and $h_{l,l-\mu}^{(\gamma)}$ for $\gamma = 0, 1$ have all been defined in Chapter 2. We may write the l^{th} update equation of the GLMS algorithm for the γ^{th} discrete sub-channel impulse response estimate as

$$\mathbf{h}_{l+1|l}^{(\gamma)} = \mathbf{U}\left(\mathbf{h}_{l|l-1}^{(\gamma)} + \widehat{\mathbf{P}}^{(\gamma)}\mathbf{d}_l^H(y_l^{(\gamma)} - \widehat{y}_{l|l-1}^{(\gamma)})\right), \tag{3.24}$$

where

$$\mathbf{d}_l = [d_{l+L_h-L_p/2}, \ldots, d_{l+1}, d_l, d_{l-1}, \ldots, d_{l-L_p/2} \mid \mathbf{0}_{1,L_h(P_G-1)}]. \tag{3.25}$$

After N_{Train} training updates, the transmitter is switched to data transmission mode. Now we need to adjust our notation for the received samples as the channel is

assumed to remain constant for the j^{th} transmission block, and the training sequence is now embedded in the UW prefix for each data block. Therefore, we define

$$y_{j,i} = \sum_{k=-L_p/2+\kappa}^{L_h-L_p/2+\kappa} x_{j,k} h_{j,i-2k} + n_{j,i}, \tag{3.26}$$

as the received samples associated with the UW prefix in each block, where $i = -2N_{UW}, \ldots, -1$ to comply with the indexing of the UW prefix. Decomposing the channel impulse response into two symbol-rate sampled sub-channel impulse responses, the l^{th} received sample of the γ^{th} discrete sub-channel for the j^{th} data transmission block may be written as

$$y_{j,l}^{(\gamma)} = \sum_{\mu=-L_p/2+l}^{L_h-L_p/2+l} x_{j,\mu} h_{j,l-\mu}^{(\gamma)} + n_{j,l}^{(\gamma)}, \quad \gamma = 0, 1. \tag{3.27}$$

Only two samples, $y_{j,-N_{UW}/2}^{(0)}$ and $y_{j,-N_{UW}/2}^{(1)}$ are used to make the $(N_{Train}+j)^{th}$ update of the channel since N_{Train} updates already took place during the training mode. Lastly, since the UW sequence is random and independent for each data transmission block, the vector $\mathbf{d}_j$ for the $(N_{Train}+j)^{th}$ update is defined as

$$\mathbf{d}_j = [x_{j,-N_{UW}/2+L_h-L_p/2}, \ldots, x_{j,-N_{UW}/2}, \ldots, x_{j,-N_{UW}/2-L_p/2} \mid \mathbf{0}_{1,L_h(P_G-1)}]. \tag{3.28}$$

Having described the channel estimation process, we note that when the channel impulse response $c(t)$ is symbol spaced and a root-raised cosine pulse shape is employed for $p(t)$, the discrete sub-channel impulse response with $\gamma = 0$ corresponds to the tap gains $\{c_n\}$ of the channel response $c(t)$, as shown in Fig. 3.6. Since the pulse shape $p(t)$ is fixed and known, it is sufficient to estimate only $c(t)$ in order to obtain an estimate of $h(t)$. In particular, we are interested in estimating the tap gains $\{c_n\}$. This suggests that instead of estimating two discrete sub-channels, the estimator now only has to estimate the first sub-channel, which means the computational complexity of the channel estimation process is halved. This is easily achieved by decimating the received samples during the training mode, as well as the received samples that correspond to

the transmission of the UW sequence during the transmission mode by a factor of two. In general, this approach can be extended to a system with N_s samples per symbol interval by decimating the received samples by a factor of N_s where applicable[3].

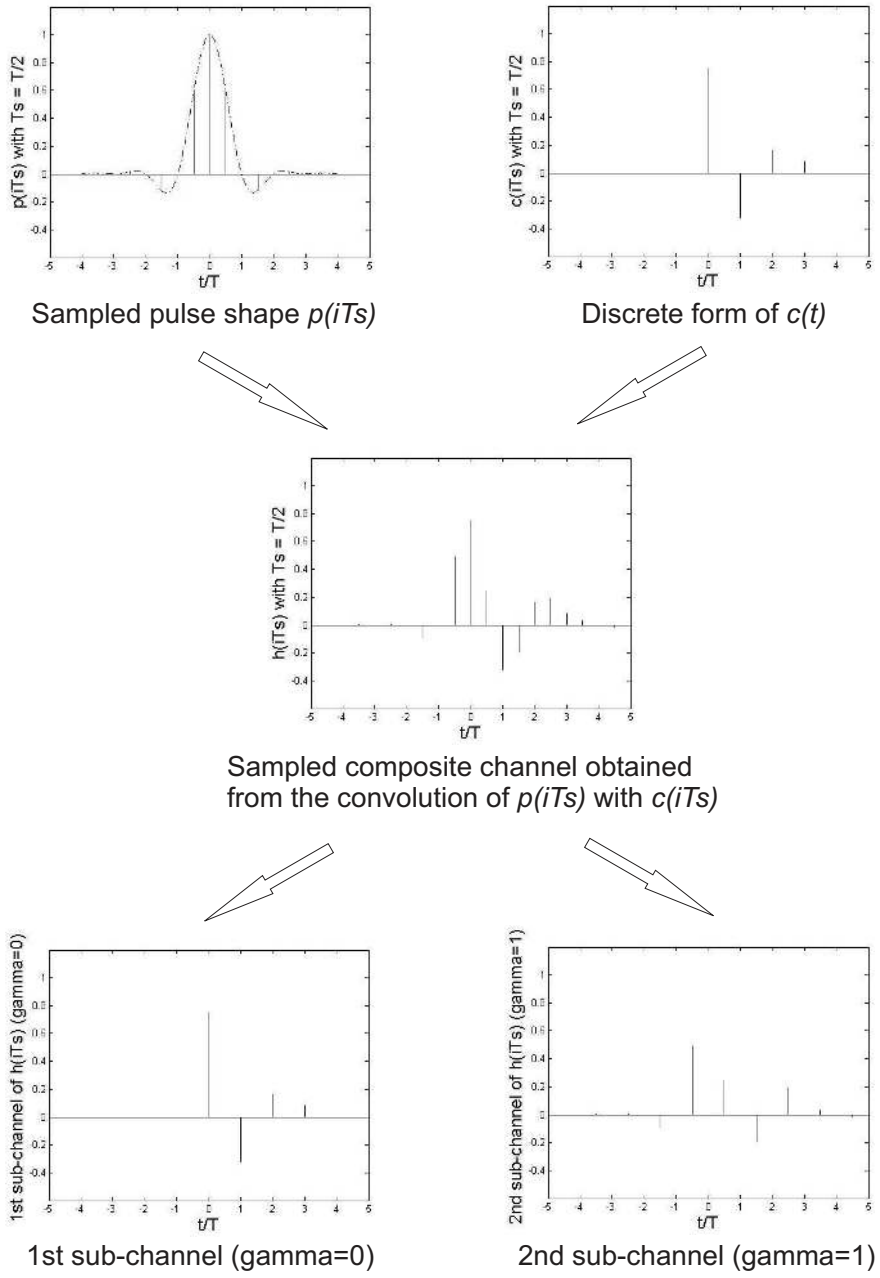


Figure 3.6 Example to show how the discrete sub-channel impulse response for $\gamma = 0$ is the symbol-rate sampled discrete impulse response of c(t).

[3]It is assumed that accurate symbol synchronization is available. In the case where accurate symbol synchronization is not available, all N_s samples will need to be used in the estimation process.

3.3 THE EQUALIZER FILTER MATRIX

We now derive the equalizer filter matrix $\mathbf{D}_{TLE}[\psi]$ for the ψ^{th} iteration in the turbo equalization process. Initially, we assume that the channel impulse response (CIR) is perfectly known at the receiver, and derive the equalizer filter matrix $\mathbf{D}_{TLE}[\psi]$ following the work of [32]. Note that many of the intermediate steps included in the present derivation were not shown in [32]. We then derive the equalizer filter matrix $\mathbf{D}_{TLE}[\psi]$ for the case where only estimated CIR is available at the receiver. This explicitly illustrates the effect of channel estimation error.

Before starting the ψ^{th} iteration of the turbo equalization process, we note that the data APPs produced by the SISO channel decoder in the previous iteration are available except when $\psi = 1$. When $\psi = 1$, the data APPs are set to be equiprobable. We use the notation $APP_{n,l}[\psi - 1] = \mathcal{P}(x_n = \mathcal{X}_l|\widehat{x}_l)$ to denote the APPs for the ψ^{th} iteration.

Following the work of [32], we first define the zero-mean vectors

$$\widetilde{\mathbf{X}}_N[\psi] \quad = \quad \mathbf{X}_N - E_{\mathbf{x},\psi-1}\{\mathbf{X}_N\} \tag{3.29}$$

and

$$\widetilde{\mathbf{X}}_{2N}[\psi] \quad = \quad \mathbf{X}_{2N} - E_{\mathbf{x},\psi-1}\{\mathbf{X}_{2N}\}, \tag{3.30}$$

where

$$E_{\mathbf{x},\psi-1}\Big\{x_n\Big\} = \sum_{l=0}^{M-1} APP_{n,l}[\psi - 1]\mathcal{X}_l. \tag{3.31}$$

is the statistical average of x_n with respect to the data vector $\mathbf{x}$ evaluated on the basis of the APPs produced by the SISO decoder after $\psi - 1$ iterations $\{APP_{n,l}[\psi - 1]\}$. We may then define the modified received vector as

$$\widetilde{\mathbf{Y}}[\psi] = \mathbf{Y} - \mathbf{R}E_{\mathbf{x},\psi-1}\{\mathbf{X}_{2N}\} = \mathbf{R}\widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V}, \tag{3.32}$$

where $\mathbf{R}$ is the composite channel matrix in the frequency-domain as defined in (3.20).

The resulting frequency-domain error is then given by

$$\widetilde{\Delta}_{TLE}[\psi] = \mathbf{D}_{TLE}[\psi]\widetilde{\mathbf{Y}}[\psi] - \widetilde{\mathbf{X}}_N[\psi], \tag{3.33}$$

and the orthogonality principle in the frequency-domain can then be expressed as

$$E\{\widetilde{\Delta}_{TLE}[\psi]\widetilde{\mathbf{Y}}[\psi]^H\} = \mathbf{0}_{N\times 2N}. \tag{3.34}$$

Substituting (3.32) and (3.33) into (3.34), we obtain

$$\begin{aligned}
E & \left\{\widetilde{\Delta}_{TLE}[\psi]\widetilde{\mathbf{Y}}[\psi]^H\right\} \\
= & E\left\{ \left(\mathbf{D}_{TLE}[\psi]\widetilde{\mathbf{Y}}[\psi] - \widetilde{\mathbf{X}}_N[\psi]\right)\widetilde{\mathbf{Y}}[\psi]^H \right\} \\
= & E\left\{ \left(\mathbf{D}_{TLE}[\psi]\left(\mathbf{R}\widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V}\right) - \widetilde{\mathbf{X}}_N[\psi]\right)\left(\mathbf{R}\widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V}\right)^H \right\} \\
= & E\left\{ \left(\mathbf{D}_{TLE}[\psi]\left(\mathbf{R}\widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V}\right) - \widetilde{\mathbf{X}}_N[\psi]\right)\left(\widetilde{\mathbf{X}}_{2N}[\psi]^H\mathbf{R}^H + \mathbf{V}^H\right) \right\} \\
= & E\left\{\mathbf{D}_{TLE}[\psi]\left(\mathbf{R}\widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V}\right)\widetilde{\mathbf{X}}_{2N}[\psi]^H\mathbf{R}^H\right\} \\
& +E\left\{\mathbf{D}_{TLE}[\psi]\left(\mathbf{R}\widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V}\right)\mathbf{V}^H\right\} - E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\mathbf{R}^H\right\} \\
& -E\left\{\widetilde{\mathbf{X}}_N[\psi]\mathbf{V}^H\right\} \\
= & \mathbf{D}_{TLE}[\psi]\mathbf{R}E\left\{\widetilde{\mathbf{X}}_{2N}[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H + \mathbf{D}_{TLE}[\psi]E\left\{\mathbf{V}\mathbf{V}^H\right\} \\
& -E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H \\
= & \mathbf{D}_{TLE}[\psi]\left(\mathbf{R}E\left\{\widetilde{\mathbf{X}}_{2N}[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H + \sigma_n^2\mathbf{I}_{2N}\right) \\
& -E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H \\
= & \mathbf{0}_{N\times 2N}.
\end{aligned} \tag{3.35}$$

Rearranging the terms in the penultimate line of (3.35) gives

$$\mathbf{D}_{TLE}[\psi]\left(\mathbf{R}E\left\{\widetilde{\mathbf{X}}_{2N}[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H + \sigma_n^2\mathbf{I}_{2N}\right) = E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H, \tag{3.36}$$

which may be solved to obtain the desired equalizer matrix as

$$\begin{aligned}\mathbf{D}_{TLE}[\psi] &= E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H \\ &\quad \cdot\left(\mathbf{R}E\left\{\widetilde{\mathbf{X}}_{2N}[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\}\mathbf{R}^H + \sigma_n^2\mathbf{I}_{2N}\right)^{-1}. \qquad (3.37)\end{aligned}$$

Expanding the term $E\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\}$ in (3.36) we obtain

$$\begin{aligned}&E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\} \\ &= E\left\{\left(\mathbf{X}_N - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}\right)\left(\mathbf{X}_{2N} - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_{2N}\right\}\right)^H\right\} \\ &= E\left\{\mathbf{X}_N\mathbf{X}_{2N}^H\right\} - E\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_{2N}^H\right\} - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E\left\{\mathbf{X}_{2N}^H\right\} \\ &\quad + E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_{2N}^H\right\}. \qquad (3.38)\end{aligned}$$

Instead of setting the terms associated with $E\{\mathbf{X}_N\}$ and $E\{\mathbf{X}_{2N}\}$ to zero, we replace them with $E_{\mathbf{x},\psi-1}\{\mathbf{X}_N\}$ and $E_{\mathbf{x},\psi-1}\{\mathbf{X}_{2N}^H\}$, respectively. This leads to the result

$$\begin{aligned}E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\} &= E\left\{\mathbf{X}_N\mathbf{X}_{2N}^H\right\} - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_{2N}^H\right\} \\ &= E\left\{\mathbf{X}_N\left[\mathbf{X}_N^H|\mathbf{X}_N^H\right]\right\} \\ &\quad - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\left[\mathbf{X}_N^H|\mathbf{X}_N^H\right]\right\} \\ &= E\left\{\mathbf{X}_N\mathbf{X}_N^H\right\}[\mathbf{I}_N|\mathbf{I}_N] \\ &\quad - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N^H\right\}[\mathbf{I}_N|\mathbf{I}_N] \\ &= \left(\sigma_x^2\mathbf{I}_N - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N^H\right\}\right)\mathbf{J}, \qquad (3.39)\end{aligned}$$

where $\mathbf{J} = [\mathbf{I}_N|\mathbf{I}_N]$. Similarly,

$$E\left\{\widetilde{\mathbf{X}}_{2N}[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\} = \mathbf{J}^H\left(\sigma_x^2\mathbf{I}_N - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N^H\right\}\right)\mathbf{J}. \qquad (3.40)$$

Since the signal constellations being considered in this thesis are normalized so that

$\sigma_x^2 = 1$, equations (3.39) and (3.40) can be reduced to

$$E\left\{\widetilde{\mathbf{X}}_N[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\} = \left(\mathbf{I}_N - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N^H\right\}\right)\mathbf{J}, \quad (3.41)$$

and

$$E\left\{\widetilde{\mathbf{X}}_{2N}[\psi]\widetilde{\mathbf{X}}_{2N}[\psi]^H\right\} = \mathbf{J}^H\left(\mathbf{I}_N - E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N^H\right\}\right)\mathbf{J}, \quad (3.42)$$

respectively. Lastly, substituting (3.41) and (3.42) into (3.37), we obtain the equalizer matrix as

$$\mathbf{D}_{TLE}[\psi] = \left(\mathbf{I}_N - \mathbf{\Phi}[\psi]\right)\mathbf{J}\mathbf{R}^H\mathbf{K}_T[\psi], \quad (3.43)$$

where

$$\mathbf{\Phi}[\psi] = E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N\right\}E_{\mathbf{x},\psi-1}\left\{\mathbf{X}_N^H\right\} \quad (3.44)$$

and

$$\mathbf{K}_T[\psi] = \left[\mathbf{R}\mathbf{J}^H\left(\mathbf{I}_N - \mathbf{\Phi}[\psi]\right)\mathbf{J}\mathbf{R}^H + \sigma_n^2\mathbf{I}_{2N}\right]^{-1}. \quad (3.45)$$

The data APPs are exploited in the turbo equalizer only for the evaluation of the matrix $\Phi[\psi] = [\phi_{i,p}[\psi]]$, where $\phi_{i,p}[\psi] = E_{\mathbf{x},\psi-1}\{X_i\}E_{x,\psi-1}\{X_p^*\}$. In particular, we have

$$\begin{aligned}\phi_{i,p}[\psi] &= \frac{1}{N}\sum_{n=0}^{N-1} E_{\mathbf{x},\psi-1}\left\{x_n\right\}\exp\left(-j\frac{2\pi in}{N}\right) \\ &\quad \cdot \sum_{l=0}^{N-1} E_{\mathbf{x},\psi-1}\left\{x_l\right\}^*\exp\left(-j\frac{2\pi pl}{N}\right), \end{aligned} \quad (3.46)$$

where

$$E_{\mathbf{x},\psi-1}\left\{x_n\right\} = \sum_{l=0}^{M-1} APP_{n,l}[\psi-1]\mathcal{X}_l. \quad (3.47)$$

Note that we require knowledge of $\mathbf{R}$, which contains information regarding the transmitted pulse shape $p(t)$ and the channel impulse response $c(t)$. Since $p(t)$ does not change and is known for every transmission, only the channel impulse response $c(t)$ needs to be estimated as proposed in the system shown in Fig. 3.1.

3.3.1 Effect of Channel Estimation Error

Now let us consider the situation when only an estimate of the channel response is available. We then have

$$\widehat{\mathbf{R}} = \mathbf{R} + \mathbf{E}, \tag{3.48}$$

where $\mathbf{R}$ is the actual frequency-domain channel response and $\mathbf{E}$ is a diagonal error matrix arising from the estimation of $\mathbf{R}$. Rearranging this, we have

$$\mathbf{R} = \widehat{\mathbf{R}} - \mathbf{E}. \tag{3.49}$$

We then use $\widehat{\mathbf{R}} - \mathbf{E}$ in place of $\mathbf{R}$ in the derivation of the equalizer matrix to get

$$\begin{aligned}
E & \left\{ \widetilde{\Delta}_{TLE}[\psi] \widetilde{\mathbf{Y}}[\psi]^H \right\} \\
= & E\left\{ \left(\mathbf{D}_{TLE}[\psi] \widetilde{\mathbf{Y}}[\psi] - \widetilde{\mathbf{X}}_N[\psi] \right) \widetilde{\mathbf{Y}}[\psi]^H \right\} \\
= & E\left\{ \left(\mathbf{D}_{TLE}[\psi] \left(\left(\widehat{\mathbf{R}} - \mathbf{E} \right) \widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V} \right) - \widetilde{\mathbf{X}}_N[\psi] \right) \right. \\
& \left. \cdot \left(\left(\widehat{\mathbf{R}} - \mathbf{E} \right) \widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V} \right)^H \right\} \\
= & E\left\{ \left(\mathbf{D}_{TLE}[\psi] \left(\left(\widehat{\mathbf{R}} - \mathbf{E} \right) \widetilde{\mathbf{X}}_{2N}[\psi] + \mathbf{V} \right) - \widetilde{\mathbf{X}}_N[\psi] \right) \right. \\
& \left. \cdot \left(\widetilde{\mathbf{X}}_{2N}[\psi]^H \left(\widehat{\mathbf{R}} - \mathbf{E} \right)^H + \mathbf{V}^H \right) \right\} \\
= & \mathbf{D}_{TLE}[\psi] E\left\{ \left(\widehat{\mathbf{R}} - \mathbf{E} \right) \widetilde{\mathbf{X}}_{2N}[\psi] \widetilde{\mathbf{X}}_{2N}[\psi]^H \left(\widehat{\mathbf{R}} - \mathbf{E} \right)^H \right\} \\
& + \mathbf{D}_{TLE}[\psi] E\left\{ \mathbf{V}\mathbf{V}^H \right\} - E\left\{ \widetilde{\mathbf{X}}_N[\psi] \widetilde{\mathbf{X}}_{2N}[\psi]^H \left(\widehat{\mathbf{R}} - \mathbf{E} \right)^H \right\} \\
& - E\left\{ \widetilde{\mathbf{X}}_N[\psi] \mathbf{V}^H \right\} \\
= & \mathbf{D}_{TLE}[\psi] \left(\widehat{\mathbf{R}} E\left\{ \widetilde{\mathbf{X}}_{2N}[\psi] \widetilde{\mathbf{X}}_{2N}[\psi]^H \right\} \widehat{\mathbf{R}} \right) \\
& + \mathbf{D}_{TLE}[\psi] \left(E\left\{ \mathbf{E} \widetilde{\mathbf{X}}_{2N}[\psi] \widetilde{\mathbf{X}}_{2N}[\psi]^H \mathbf{E}^H \right\} + \sigma_n^2 \mathbf{I}_{2N} \right) \\
& - E\left\{ \widetilde{\mathbf{X}}_N[\psi] \widetilde{\mathbf{X}}_{2N}[\psi]^H \right\} \widehat{\mathbf{R}}^H \\
= & \mathbf{0}_{N \times 2N},
\end{aligned} \tag{3.50}$$

which gives the solution

$$\mathbf{D}_{TLE}[\psi] \;=\; (\mathbf{I}_N - \mathbf{\Phi}[\psi])\,\mathbf{J}\widehat{\mathbf{R}}^H\mathbf{K}_T[\psi], \tag{3.51}$$

which is formally identical to (3.43). In fact, we now have

$$\mathbf{K}_T[\psi] = \left[\widehat{\mathbf{R}}\mathbf{J}^H\left(\mathbf{I}_N - \mathbf{\Phi}[\psi]\right)\mathbf{J}\widehat{\mathbf{R}}^H + E\left\{\mathbf{E}\mathbf{E}^H\right\}\mathbf{J}^H\left(\mathbf{I}_N - \mathbf{\Phi}[\psi]\right)\mathbf{J} + \sigma_n^2\mathbf{I}_{2N}\right]^{-1}. \tag{3.52}$$

This explicitly shows the effect of channel estimation error as a bias in the equalizer gain matrix $\mathbf{K}_T[\psi]$. From this, it can be seen that the equalizer is compensating for the estimated channel response $\widehat{\mathbf{R}}$ rather than the actual response $\mathbf{R}$.

Now let us examine the error matrix $\mathbf{E}$ and the matrix $E\{\mathbf{E}\mathbf{E}^H\}$. Starting with the channel response $c(t)$, we can write it in mathematical form as

$$c(t) = \sum_{n=0}^{L_c-1} c_n\delta(t-\tau_n). \tag{3.53}$$

The estimated channel response may be written as

$$\widehat{c}(t) = \sum_{n=0}^{L_c-1} (c_n + e_n)\delta(t-\tau_n), \tag{3.54}$$

where $\{e_n\}$ are the errors introduced in the estimation process. Taking the Fourier transform of $\widehat{c}(t)$ we obtain

$$\begin{aligned}
\widehat{C}(f) &= \sum_{n=0}^{L_c-1} (c_n + e_n)\exp\left(-j2\pi f\tau_n\right) \\
&= \sum_{n=0}^{L_c-1} c_n\exp\left(-j2\pi f\tau_n\right) + \sum_{n=0}^{L_c-1} e_n\exp\left(-j2\pi f\tau_n\right) \\
&= C(f) + E(f).
\end{aligned} \tag{3.55}$$

Initially, both $C(f)$ and $\widehat{C}(f)$ are not band-limited if we examine (3.55). However, the pulse shaping restricts the transmission bandwidth to have a maximum bandwidth of $2/T$. Therefore, the frequency range $[-1/T, 1/T]$ of $\widehat{C}(f)$ is the frequency range of interest. Within this range, $\widehat{C}(f)$ is sampled accordingly to form the estimated discrete

channel response. For the ease of explanation, let us suppose that the pulse shaping function $P(f)$ is a rectangular function with unity gain. Hence, $\widehat{C}(f) = \widehat{C}(f) \cdot P(f)$ in the frequency range $[-1/T, 1/T]$. Relating this to the estimated composite diagonal channel matrix $\widehat{\mathbf{R}}$, each element of $\widehat{\mathbf{R}}$ is a sample obtained from $\widehat{C}(f)$.

By writing $\widehat{\mathbf{R}} = \mathbf{R} + \mathbf{E}$, it is easy to see that each element of the diagonal matrix $\mathbf{E}$ can be obtained by sampling $E(f) = \sum_{n=0}^{L_c-1} e_n \exp(-j2\pi f\tau_n)$, defined previously in (3.55). Assuming that the estimation errors associated with each channel tap $\{e_0, \ldots, e_{L_c-1}\}$ are independent, we then obtain

$$E\left\{\left(\sum_{n=0}^{L_c-1} e_n \exp(-j2\pi f\tau_n)\right)\left(\sum_{n=0}^{L_c} e_n \exp(-j2\pi f\tau_n)\right)^*\right\} = \sum_{n=0}^{L_c-1} E\{e_n e_n^*\}. \quad (3.56)$$

This value is in fact the value for each component of the diagonal matrix $E\{\mathbf{E}\mathbf{E}^H\}$ for the case where a rectangular pulse shape is used. The result is analogous to the MSD calculation used in [37, 38] to evaluate the GRLS and GLMS estimator performances. Here, we note that as L_c increases, the elements of the matrix $E\{\mathbf{E}\mathbf{E}^H\}$ will also increase in value due to the summation $\sum_{n=0}^{L_c-1} E\{e_n e_n^*\}$. Thus, we would expect an increase in the number of channel taps to result in larger channel estimation errors. On the other hand, if a root-raised cosine or a raised cosine pulse response is used for $P(f)$, the elements of the error matrix $\mathbf{E}$ will decrease in amplitude since $P(f) \leq 1$. Hence, $E\{\mathbf{E}\mathbf{E}^H\}$ will have components equal to or smaller than $\sum_{n=0}^{L_c-1} E\{e_n e_n^*\}$, as defined in (3.56).

Ideally, a constant bias needs to be set for each element of the matrix $E\{\mathbf{E}\mathbf{E}^H\}$ in (3.52). The statistical properties of the channel estimation error are not known and are usually not easy to specify. As noted in [37, 38], the estimation error is related to the channel fade rate and the predictor length and order. In order to calculate $E\{\mathbf{E}\mathbf{E}^H\}$ analytically or by means of computer simulation, we require the CSI to be ideally known. This is not easily achieved in a practical system. We were able to find values for $\sum_{n=0}^{L_c-1} E\{\mathbf{E}\mathbf{E}^H\}$ by means of computer simulation in the process of evaluating the GLMS estimator performance for the 11-TAP and SUI-5 channel. The results are presented in a later chapter of this thesis and it is seen that the MSD is

small in magnitude, especially for high SNR values.

Examining the term $E\left\{\mathbf{E}\mathbf{E}^H\right\}\mathbf{J}^H\left(\mathbf{I}_N-\mathbf{\Phi}[\psi]\right)\mathbf{J}$ in (3.52), it is easy to see that since elements of the matrix $E\{\mathbf{E}\mathbf{E}^H\}$ are small, the elements in the matrix resulting from $E\left\{\mathbf{E}\mathbf{E}^H\right\}\mathbf{J}^H\left(\mathbf{I}_N-\mathbf{\Phi}[\psi]\right)\mathbf{J}$ will also be small. Hence, we may set this term to zero as a good approximation and reduce the gain matrix of (3.52) to the approximated form

$$\mathbf{K}_T[\psi]=\left[\widehat{\mathbf{R}}\mathbf{J}^H\left(\mathbf{I}_N-\mathbf{\Phi}[\psi]\right)\mathbf{J}\widehat{\mathbf{R}}^H+\sigma_n^2\mathbf{I}_{2N}\right]^{-1}. \tag{3.57}$$

We note that (3.57) has the same form as (3.45), which was obtained assuming ideally known CSI. Clearly, since $\widehat{\mathbf{R}}=\mathbf{R}+\mathbf{E}$ is used in (3.57) instead of the actual composite channel $\mathbf{R}$, a degradation in receiver performance is expected compared to the case of perfectly known CSI since the equalizer is now "mismatched" to the channel.

3.4 CALCULATING SOFT INFORMATION IN THE DECODER

Having derived the equalizer forward matrix $\mathbf{D}_{TLE}[\psi]$, we now consider how the soft information is calculated by the SISO decoder. Referring to Fig. 3.1, the output of the equalizer is an estimate of the transmitted sequence. After deinterleaving, this sequence becomes an estimate of the encoded message sequence. It is fed into the SISO convolutional decoder, which generates soft *a posteriori* symbol probabilities. Based on the discussion in [51], we now describe the BCJR algorithm [51], which is a symbol-by-symbol maximum *a posteriori* (MAP) algorithm, and is used in this thesis.

Having defined the trellis structure of a convolutional code in Chapter 2, let $\mathcal{S}$ be the number of code states, s_i, where $i=0,1,\ldots,\mathcal{S}-1$. We denote the state of the encoder at time t by S_t and its corresponding output by x_t. A state sequence of the source ranging from time t to time t' is denoted by $\mathbf{S}_t^{t'}=S_t,S_{t+1},\ldots,S_{t'}$, and the corresponding output sequence by $\mathbf{x}_t^{t'}=x_t,x_{t+1},\ldots,x_{t'}$. The state transitions of the convolutional encoder are governed by the transition probabilities

$$p_t(s_i|s_j)=\mathcal{P}(S_t=s_i|S_{t-1}=s_j) \tag{3.58}$$

and the output of the convolutional encoder by the probabilities

$$q_t(\mathcal{X}_l|s_i, s_j) = \mathcal{P}(x_t = \mathcal{X}_l|S_{t-1} = s_j, S_t = s_i), \tag{3.59}$$

where $\mathcal{X}_l$ is a point in the signal constellation. The encoder starts and terminates in the all-zero state so that $S_0 = S_{N-1} = \mathbf{0}$ within each block $\mathbf{x}_{N_{CS}}$, producing the output sequence $\mathbf{x}_0^{N_{CS}-1}$. The decoder examines the equalized signal sequence $\widehat{\mathbf{x}}_0^{N_{CS}-1} = \widehat{x}_0, \widehat{x}_1, \ldots, \widehat{x}_{N_{CS}-1}$ and estimates the APPs of the states and transitions of the encoder. In particular, the conditional APP associated with each node in the trellis is given by

$$\mathcal{P}(S_t = s_i|\widehat{\mathbf{x}}_0^{N_{CS}-1}) = \frac{\mathcal{P}(S_t = s_i, \widehat{\mathbf{x}}_0^{N_{CS}-1})}{\mathcal{P}(\widehat{\mathbf{x}}_0^{N_{CS}-1})}, \tag{3.60}$$

and the APP associated with each branch and the corresponding output symbol in the trellis is given by

$$\mathcal{P}(S_{t-1} = s_j, S_t = s_i|\widehat{\mathbf{x}}_0^{N_{CS}-1}) = \frac{\mathcal{P}(S_{t-1} = s_j, S_t = s_i; \widehat{\mathbf{x}}_0^{N_{CS}-1})}{\mathcal{P}(\widehat{\mathbf{x}}_0^{N_{CS}-1})}. \tag{3.61}$$

To obtain the probabilities

$$\lambda_t(s_i) \triangleq \mathcal{P}(S_t = s_i, \widehat{\mathbf{x}}_0^{N_{CS}-1}) \tag{3.62}$$

and

$$\sigma_t(s_j, s_i) \triangleq \mathcal{P}(S_{t-1} = s_j, S_t = s_i, \widehat{\mathbf{x}}_0^{N_{CS}-1}), \tag{3.63}$$

we consider the probability functions

$$\alpha_t(s_i) = \mathcal{P}(S_t = s_i, \widehat{\mathbf{x}}_0^t), \tag{3.64}$$

$$\beta_t(s_i) = \mathcal{P}(\widehat{\mathbf{x}}_{t+1}^{N_{CS}}|S_t = s_i), \tag{3.65}$$

and

$$\gamma_t(s_j, s_i) = \mathcal{P}(S_t = s_i, \widehat{x}_t|S_{t-1} = s_j). \tag{3.66}$$

Exploiting the Markov property [9] we have [51]

$$\begin{aligned}\lambda_t(s_i) &= \mathcal{P}(S_t = s_i, \widehat{\mathbf{x}}_0^t) \cdot \mathcal{P}(\widehat{\mathbf{x}}_{t+1}^{N_{CS}} | S_t = s_i, \widehat{\mathbf{x}}_0^t) \\ &= \alpha_t(s_i) \cdot \mathcal{P}(\widehat{\mathbf{x}}_{t+1}^{N_{CS}} | S_t = s_i) \\ &= \alpha_t(s_i) \cdot \beta_t(s_i), \end{aligned} \tag{3.67}$$

and

$$\begin{aligned}\sigma_t(s_j, s_i) &= \mathcal{P}(S_{t-1} = s_j, \widehat{\mathbf{x}}_0^{t-1}) \cdot \mathcal{P}(S_t = s_i, \widehat{x}_t | S_{t-1} = j) \cdot \mathcal{P}(\widehat{\mathbf{x}}_{t+1}^{N_{CS}} | S_t = i) \\ &= \alpha_{t-1}(s_j) \cdot \gamma_t(s_j, s_i) \cdot \beta_t(s_i). \end{aligned} \tag{3.68}$$

For $t = 1, 2, \ldots, N_{CS}$ we then obtain [51]

$$\begin{aligned}\alpha_t(s_i) &= \sum_{j=0}^{\mathcal{S}-1} \mathcal{P}(S_{t-1} = s_j, S_t = s_i, \widehat{\mathbf{x}}_0^t) \\ &= \sum_{j=0}^{\mathcal{S}-1} \mathcal{P}(S_{t-1} = s_j, \widehat{\mathbf{x}}_0^t) \cdot \mathcal{P}(S_t = s_i, \widehat{x}_t | S_{t-1} = s_j) \\ &= \sum_{j=0}^{\mathcal{S}-1} \alpha_{t-1}(s_j) \cdot \gamma_t(s_j, s_i). \end{aligned} \tag{3.69}$$

For $t = 0$, we have the boundary conditions

$$\alpha_0(s_i) = \begin{cases} 1, & i = 0 \\ 0, & i \neq 0. \end{cases} \tag{3.70}$$

Similarly, for $t = 0, 1, \ldots, N_{CS} - 1$ we have the recursion

$$\begin{aligned}\beta_t(s_i) &= \sum_{j=0}^{\mathcal{S}-1} \mathcal{P}(S_{t+1} = s_j, \widehat{\mathbf{x}}_{t+1}^{N_{CS}} | S_t = s_i) \\ &= \sum_{j=0}^{\mathcal{S}-1} \mathcal{P}(S_{t+1} = s_j, \mathbf{x}_{t+1} | S_t = s_i) \cdot \mathcal{P}(\widehat{\mathbf{x}}_{t+2}^{N_{CS}} | S_{t+1} = s_j) \\ &= \sum_{j=0}^{\mathcal{S}-1} \beta_{t+1}(s_j) \cdot \gamma_{t+1}(s_j, s_i), \end{aligned} \tag{3.71}$$

with boundary conditions

$$\beta_{N_{CS}}(s_i) = \begin{cases} 1, & i = 0 \\ 0, & i \neq 0. \end{cases} \tag{3.72}$$

These boundary conditions can be achieved by forcing the last $\mathcal{M}$ input bits into the convolutional encoder to be zero. Lastly, we may write

$$\begin{aligned}
\gamma_t(s_j, s_i) &= \sum_{l=0}^{M} \mathcal{P}(S_t = s_i, \widehat{x}_t, x_t = \mathcal{X}_l | S_{t-1} = s_j) \\
&= \sum_{l=0}^{M} \frac{\mathcal{P}(S_t = s_i, \widehat{x}_t, x_t = \mathcal{X}_l, S_{t-1} = s_j)}{\mathcal{P}(S_{t-1} = s_j)} \\
&= \sum_{l=0}^{M} \mathcal{P}(\widehat{x}_t, x_t = \mathcal{X}_l | S_{t-1} = s_j, S_t = s_i) \cdot \frac{\mathcal{P}(S_{t-1} = s_j, S_t = s_i)}{\mathcal{P}(S_{t-1} = s_j)} \\
&= \sum_{l=0}^{M} \mathcal{P}(\widehat{x}_t, x_t = \mathcal{X}_l | S_{t-1} = s_j, S_t = s_i) \cdot \mathcal{P}(S_t = s_i | S_{t-1} = s_j) \\
&= \sum_{l=0}^{M} \frac{\mathcal{P}(\widehat{x}_t, x_t = \mathcal{X}_l, S_{t-1} = s_j, S_t = s_i)}{\mathcal{P}(S_{t-1} = s_j, S_t = s_i)} \cdot \mathcal{P}(S_t = s_i | S_{t-1} = s_j) \\
&= \sum_{l=0}^{M} \mathcal{P}(\widehat{x}_t | x_t = \mathcal{X}_l, S_{t-1} = s_j, S_t = s_i) \\
&\qquad \cdot \frac{\mathcal{P}(x_t = \mathcal{X}_l, S_{t-1} = s_j, S_t = s_i)}{\mathcal{P}(S_{t-1} = s_j, S_t = s_i)} \cdot \mathcal{P}(S_t = s_i | S_{t-1} = s_j) \\
&= \sum_{l=0}^{M} \mathcal{P}(\widehat{x}_t | x_t = \mathcal{X}_l, S_{t-1} = s_j, S_t = s_i) \\
&\qquad \cdot \mathcal{P}(x_t = \mathcal{X}_l | S_{t-1} = s_j, S_t = s_i) \cdot \mathcal{P}(S_t = s_i | S_{t-1} = s_j) \\
&= \sum_{l=0}^{M} \mathcal{P}(\widehat{x}_t | x_t = \mathcal{X}_l, S_{t-1} = s_j, S_t = s_i) \cdot q_t(\mathcal{X}_l | s_i, s_j) \cdot p_t(s_i | s_j).
\end{aligned} \tag{3.73}$$

Examining (3.73), the term $\mathcal{P}(x_t = \mathcal{X}_l | S_{t-1} = s_j, S_t = s_i) = q_t(\mathcal{X}_l | s_i, s_j)$ equals 1 if $\mathcal{X}_l$ corresponds to the output from the convolutional encoder changing from state s_j to state s_i, and 0 otherwise. This allows expression (3.73) to be reduced to

$$\gamma_t(s_j, s_i) = \sum_{l=0}^{M} \mathcal{P}(\widehat{x}_t | x_t = \mathcal{X}_l) \cdot q_t(\mathcal{X}_l | s_i, s_j) \cdot p_t(s_i | s_j). \tag{3.74}$$

The term $\mathcal{P}(S_t = s_i | S_{t-1} = s_j) = p_t(s_i | s_j)$ is assumed to be equal for all valid state

transitions, since there is no knowledge of the input to the encoder at the receiver. This implies that if there are two branches out of each state, then $\mathcal{P}(S_t = s_i|S_{t-1} = s_j) = 1/2$. In general, if there are k inputs to a convolutional encoder, there will be 2^k branches originating from each state. Therefore, $\mathcal{P}(S_t = s_i|S_{t-1} = s_j) = 1/2^k$. To calculate $\mathcal{P}(\widehat{x}_t|x_t = \mathcal{X}_l)$, we require knowledge of the distribution of the equalization error σ_e^2. Assuming that this error is Gaussian, we can estimate its variance for each block transmission by taking the mean squared error of the equalized signal and the UW suffix within each block. This method of calculating the variance of the estimation error σ_e^2 is relatively simple and accurate since the UW suffix is known. Furthermore, we can calculate $\mathcal{P}(\widehat{x}_t|x_t = \mathcal{X}_l)$ by employing the Gaussian probability density function and examining the Euclidean distance between $\widehat{x}_t$ and $x_t = \mathcal{X}_l$ over all possible $\mathcal{X}_l$, followed by a normalization process to convert the densities into probabilities.

Lastly, we divide $\lambda_t(s_i)$ and $\sigma_t(s_j, s_i)$ by $\mathcal{P}(\widehat{\mathbf{x}}_0^{N_{CS}-1})$ to obtain the conditional probabilities

$$\mathcal{P}(S_t = s_i|\widehat{\mathbf{x}}_0^{N_{CS}-1}) = \frac{\mathcal{P}(S_t = s_i, \widehat{\mathbf{x}}_0^{N_{CS}-1})}{\mathcal{P}(\widehat{\mathbf{x}}_0^{N_{CS}-1})} \tag{3.75}$$

and

$$\mathcal{P}(S_{t-1} = s_j, S_t = s_i|\widehat{\mathbf{x}}_0^{N_{CS}-1}) = \frac{\mathcal{P}(S_{t-1} = s_j, S_t = s_i; \widehat{\mathbf{x}}_0^{N_{CS}-1})}{\mathcal{P}(\widehat{\mathbf{x}}_0^{N_{CS}-1})}, \tag{3.76}$$

following Bayes' theorem.

3.4.1 Example for QPSK

An example is used to demonstrate how the probabilities $\mathcal{P}(\widehat{x}_t|x_t = \mathcal{X}_l)$ are obtained. In this case, we use a QPSK signal constellation. Using (3.74) we can then calculate the symbol APPs. Suppose that we have the equalized signal sample $\widehat{x} = 0.3323 + j0.4172$ as shown in Fig. 3.7.

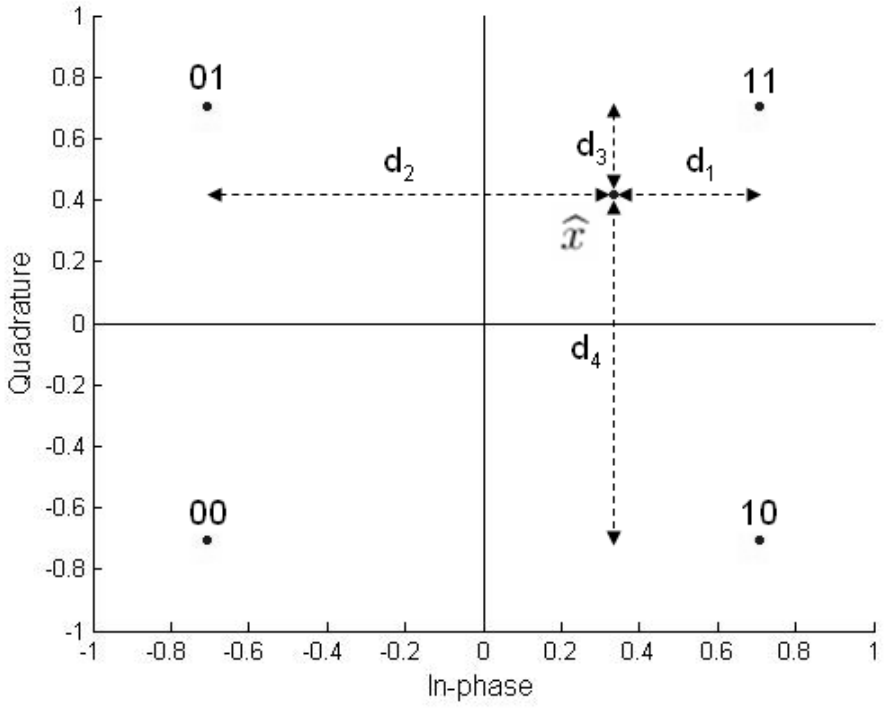


Figure 3.7 Example of a received signal in a system employing a QPSK signal constellation.

The values d_1^2, d_2^2, d_3^2, and d_4^2 are calculated to be

$$d_1^2 = \left(\frac{1}{\sqrt{2}} - 0.3323\right)^2 = 0.1405 \tag{3.77}$$

$$d_2^2 = \left(\frac{-1}{\sqrt{2}} - 0.3323\right)^2 = 1.080 \tag{3.78}$$

$$d_3^2 = \left(\frac{1}{\sqrt{2}} - 0.4172\right)^2 = 0.0840 \tag{3.79}$$

$$d_4^2 = \left(\frac{-1}{\sqrt{2}} - 0.4172\right)^2 = 1.2641 \tag{3.80}$$

Let us assume that the variance of the equalization error is $\sigma_e^2/2 = N_0 = 0.5$ in each dimension[4]. It is obtained by calculating the mean squared error between the UW suffix and the portion of the equalized signals corresponding to the UW suffix.

We may then obtain the relative density values from the Gaussian probability density

[4] $N_0 = 0.5$ is just an assumed value for the purpose of this example. It does not reflect on actual variance of equalization errors for the proposed system

function as

$$\begin{aligned} p_{d_1} &= \frac{1}{2\pi \cdot \sqrt{\frac{\sigma_e^2}{2}}} \exp\left[-\frac{\widehat{x} - \mathcal{X}_l}{\sigma_e^2}\right] \\ &= \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{0.1405}{2 \cdot 0.5}\right] = 0.1956 \quad (3.81) \\ p_{d_2} &= \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{1.0804}{2 \cdot 0.5}\right] = 0.0764 \quad (3.82) \\ p_{d_3} &= \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{0.0840}{2 \cdot 0.5}\right] = 0.2069 \quad (3.83) \\ p_{d_4} &= \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{1.2641}{2 \cdot 0.5}\right] = 0.0636, \quad (3.84) \end{aligned}$$

which gives the normalized probabilities as

$$\begin{aligned} \mathcal{P}(\widehat{x}|\text{The 1st bit is 1}) &= \frac{0.1956}{0.1956 + 0.0764} = 0.7191 \quad (3.85) \\ \mathcal{P}(\widehat{x}|\text{The 1st bit is 0}) &= \frac{0.0764}{0.1956 + 0.0764} = 0.2809 \quad (3.86) \\ \mathcal{P}(\widehat{x}|\text{The 2nd bit is 1}) &= \frac{0.2069}{0.2069 + 0.0636} = 0.7649 \quad (3.87) \\ \mathcal{P}(\widehat{x}|\text{The 2nd bit is 0}) &= \frac{0.0636}{0.2069 + 0.0636} = 0.2351 \quad (3.88) \end{aligned}$$

and hence,

$$\begin{aligned} \mathcal{P}(\widehat{x}|x = 11) &= 0.7191 \cdot 0.7649 = 0.5500 \quad (3.89) \\ \mathcal{P}(\widehat{x}|x = 10) &= 0.7191 \cdot 0.2351 = 0.1691 \quad (3.90) \\ \mathcal{P}(\widehat{x}|x = 01) &= 0.2809 \cdot 0.7649 = 0.2149 \quad (3.91) \\ \mathcal{P}(\widehat{x}|x = 00) &= 0.2351 \cdot 0.2809 = 0.0660 \quad (3.92) \end{aligned}$$

These estimated probabilities are then used in (3.74) to calculate the symbol APPs.

3.4.2 Example for 16-QAM

The same concept may be extended to a 16-QAM signal constellation to calculate the symbol APPs. However, some modification is required as demonstrated by the following example. Suppose that we have an equalized signal sample $\widehat{x} = 0.5123 + j0.4172$ as shown in Fig. 3.8.

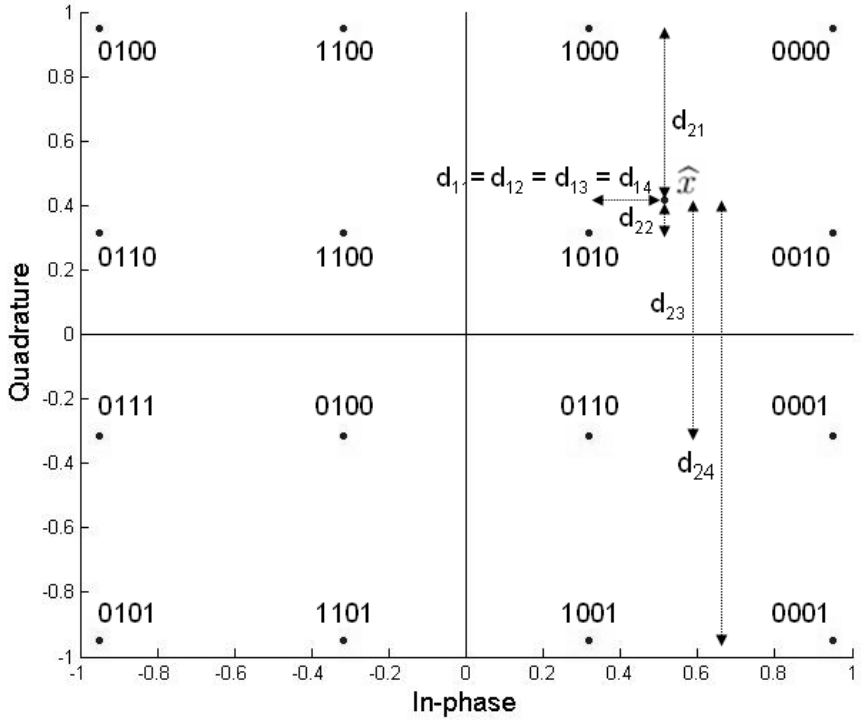


Figure 3.8 Example of a received signal in a system employing a 16-QAM signal constellation.

The values d_{11}^2, d_{12}^2, d_{13}^2, d_{14}^2, d_{21}^2, d_{22}^2, d_{23}^2, and d_{24}^2 are calculated as

$$d_{11}^2 = d_{12}^2 = d_{13}^2 = d_{14}^2 = \left(\frac{1}{\sqrt{10}} - 0.5123\right)^2 = 0.03844 \tag{3.93}$$

$$d_{21}^2 = \left(\frac{3}{\sqrt{10}} - 0.4172\right)^2 = 0.2825 \tag{3.94}$$

$$d_{22}^2 = \left(\frac{1}{\sqrt{10}} - 0.4172\right)^2 = 0.0102 \tag{3.95}$$

$$d_{23}^2 = \left(\frac{-1}{\sqrt{10}} - 0.4172\right)^2 = 0.5379 \tag{3.96}$$

$$d_{24}^2 = \left(\frac{-3}{\sqrt{10}} - 0.4172\right)^2 = 1.8656 \tag{3.97}$$

Again, let us assume that the variance of the estimation error is $\sigma_e^2/2 = N_0 = 0.5$ in each dimension. We may then obtain the following estimated probability density values (assuming a Gaussian distribution)

$$p_{d_{11}} = p_{d_{12}} = p_{d_{13}} = p_{d_{14}} = \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{0.03844}{2 \cdot 0.5}\right] = 0.2166 \quad (3.98)$$

$$p_{d_{21}} = \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{0.2825}{2 \cdot 0.5}\right] = 0.0537 \quad (3.99)$$

$$p_{d_{22}} = \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{0.0102}{2 \cdot 0.5}\right] = 0.2228 \quad (3.100)$$

$$p_{d_{23}} = \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{0.5379}{2 \cdot 0.5}\right] = 0.1314 \quad (3.101)$$

$$p_{d_{24}} = \frac{1}{2\pi \cdot \sqrt{0.5}} \exp\left[-\frac{1.8656}{2 \cdot 0.5}\right] = 0.0348 \quad (3.102)$$

which can be used to calculate the probability metrics as

$$\mathcal{P}(\widehat{x}|x = 1000) = p_{d_{11}} \cdot p_{d_{21}} = 0.2166 \cdot 0.0537 = 0.5500 \quad (3.103)$$

$$\mathcal{P}(\widehat{x}|x = 1010) = p_{d_{12}} \cdot p_{d_{22}} = 0.2166 \cdot 0.2228 = 0.1691 \quad (3.104)$$

$$\mathcal{P}(\widehat{x}|x = 1011) = p_{d_{13}} \cdot p_{d_{23}} = 0.2166 \cdot 0.1314 = 0.1314 \quad (3.105)$$

$$\mathcal{P}(\widehat{x}|x = 1001) = p_{d_{14}} \cdot p_{d_{24}} = 0.2166 \cdot 0.0348 = 0.0348 \quad (3.106)$$

Following this procedure and remembering to normalize the metrics, it is possible to obtain $\mathcal{P}(\widehat{x}|x = \mathcal{X}_l)$ for all $\mathcal{X}_l$ in the signal constellation. However, we can not substitute $\mathcal{P}(\widehat{x}|x = \mathcal{X}_l)$ into (3.74) since each $\mathcal{X}_l$ is labeled by outputs of the convolutional encoder corresponding to two time instances. Therefore, we need to sum appropriate terms to obtain $\mathcal{P}(\widehat{x}|00 \times \times)$, $\mathcal{P}(\widehat{x}|01 \times \times)$, $\mathcal{P}(\widehat{x}|10 \times \times)$, $\mathcal{P}(\widehat{x}|11 \times \times)$, $\mathcal{P}(\widehat{x}| \times \times 00)$, $\mathcal{P}(\widehat{x}| \times \times 01)$, $\mathcal{P}(\widehat{x}| \times \times 10)$ and $\mathcal{P}(\widehat{x}| \times \times 11)$. For example, $\mathcal{P}(\widehat{x}|10 \times \times) = \mathcal{P}(\widehat{x}|x = 1000) + \mathcal{P}(\widehat{x}|x = 1010) + \mathcal{P}(\widehat{x}|x = 1011) + \mathcal{P}(\widehat{x}|x = 1001) = 0.8853$. These values are then used in (3.74) to calculate the symbol APPs.

3.5 SUMMARY

This chapter has provided a detailed discussion of the proposed system. It includes an overview of the system, the derivation of the equalizer forward matrix and shows how it is related to the turbo equalization process. A detailed description of how the soft information is calculated by the SISO decoder is also included with the aid of examples to help the understanding of this concept. Simulation results for the proposed system will be presented in the next chapter.

Chapter 4

COMPUTER SIMULATIONS

This chapter uses simulation to investigate the performance of the GLMS channel estimator and the combined turbo equalization system for both the 11-Tap and SUI-5 channels. In all of the simulation results, it is assumed that the symbol period is $T = 1\mu s$[1], and that the root-raised cosine pulse shape $p(t)$ has a rolloff of $\alpha = 0.4$ truncated to $L_p = 10$ symbol intervals. The DFT order is $N = 1024$, where $N = N_{CS} + N_{UW}$ as defined in Chapter 3. The interleaver is a 64-by-16 block interleaver. For simulation results involving bit-error rate (BER) measurements, a minimum of 250 bit errors were collected for each signal-to-noise ratio (SNR). The SNR is defined as E_b/N_0, where E_b is the average received energy per information bit. This is given by [42]

$$E_b = \frac{E_s}{\mathcal{R} \cdot \log_2 M}, \tag{4.1}$$

where E_s is the average received symbol energy for the M-ary signal constellation and $\mathcal{R}$ is the code rate of the convolutional code.

The channel sampling rate needs to be set carefully to obtain valid simulation results. For the two channels being considered in this thesis, the sampling rate of the 11-TAP channel does not require altering as the channel is defined in terms of the normalized Doppler frequency f_dT. On the other hand, the sampling rate of the SUI-5 channel needs to be set as the channel is defined in terms of the Doppler frequency f_d. Detailed explanation regarding the SUI-5 channel sampling rate will be presented in a later part of this chapter.

[1]The symbol period is set to $1\mu s$ to result in a relatively long symbol delay spread for the system operating with 11-TAP and SUI-5 channels.

4.1 MOBILE WIRELESS CHANNEL

Referring to the description of the 11-TAP channel in Chapter 1, the 11-TAP channel taps gains are $1\mu s$ apart, and have a maximum overall delay spread of $\tau_{max} = 10\mu s$. Having defined the symbol period $T = 1\mu s$, each channel tap is equivalently symbol spaced.

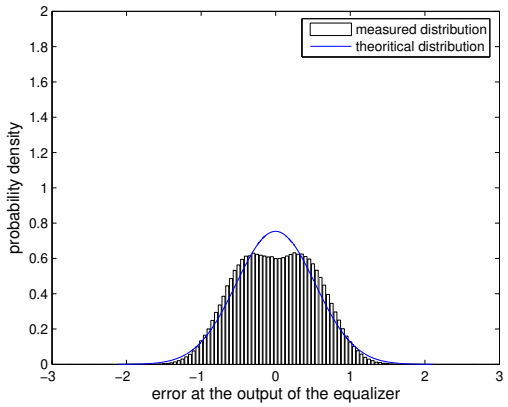
The length of the UW sequence N_{UW} is set so that it is longer than the duration of the composite channel impulse response. For the 11-TAP channel, the maximum delay is $10\mu s$, which corresponds to $L_c = 10$ symbol intervals. Therefore, the composite channel impulse response will have a duration of $L_c+L_p = 10+10 = 20$ symbol intervals. Hence, N_{UW} is set to 25, which is greater than 20. Having defined $N = 1024$, the overall length of the transmission block is $N + N_{UW} = 1049$ and the amount of overhead for each transmission is

$$\mathcal{O}_{11-TAP} = \frac{25 \times 2}{1049} \times 100\% = 4.77\%. \tag{4.2}$$

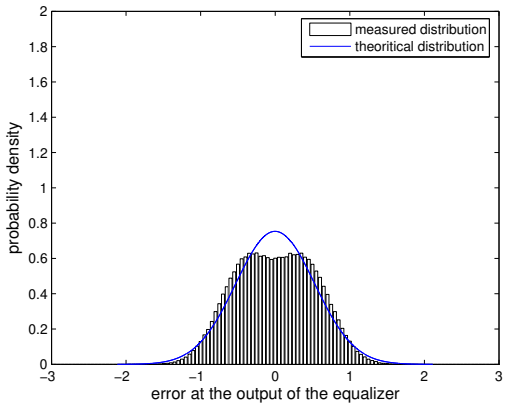
The length N_{CS}, which corresponds to the number of encoded channel symbols, is therefore $N - N_{UW} = 999$.

4.1.1 System Error Performance with ideally known CSI

Before simulating the system performance with ideally known CSI for the 11-TAP channel, we conducted a series of simulations to investigate the distribution of the equalization error. To do this, we transmitted blocks of symbols chosen randomly from the signal constellation of interest (no coding) and plotted the histogram of the equalization error at various received SNRs as shown in Fig. 4.1, 4.2, 4.3 and 4.4. The solid line is a Gaussian density function with zero-mean and the mean squared value of the equalization error (MSE) as the variance. The estimation error obtained after 2000 transmissions was plotted in a histogram with the area normalized to unity. Comparing the histogram and the solid line, we can see that there is a close resemblance. The resemblance is particularly good in the tails of the distribution, but at low SNR it is not as good at the peak. At high SNR the error is almost exactly Gaussian. This shows that the assumption that the equalization error is Gaussian is reasonable.

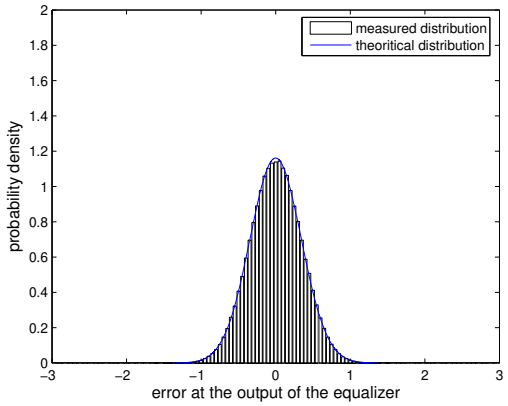


(a) In-phase with MSE = 0.2799.

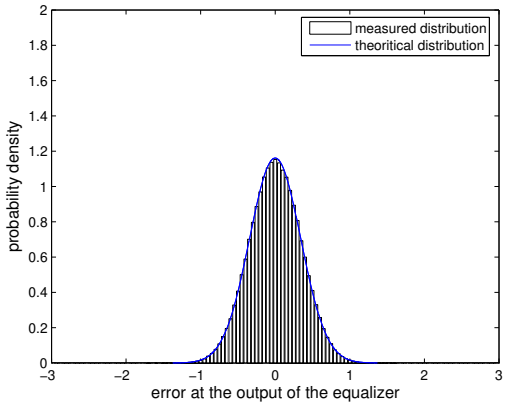


(b) Quadrature with MSE = 0.2799.

Figure 4.1 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation.

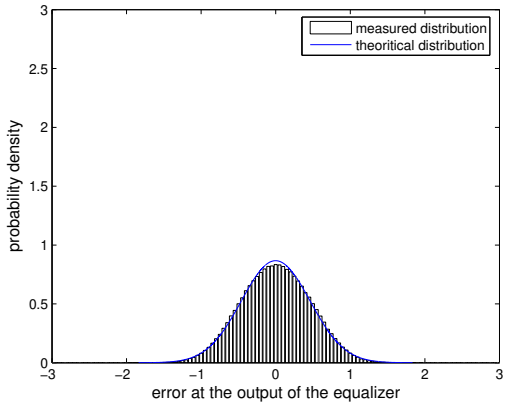


(a) In-phase with MSE = 0.1180.

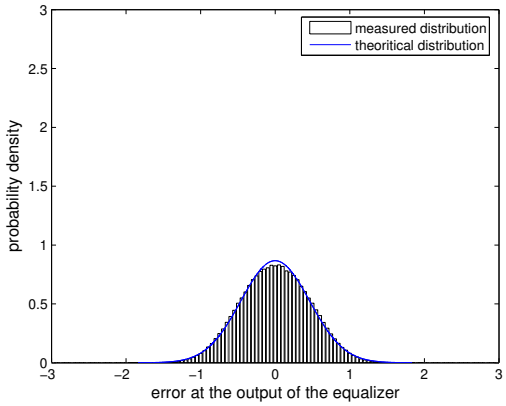


(b) Quadrature with MSE = 0.1180.

Figure 4.2 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation.

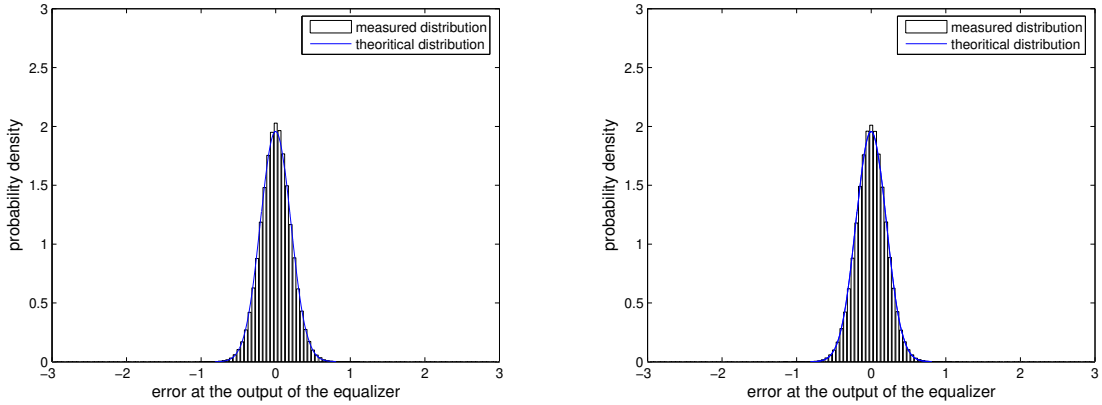


(a) In-phase with MSE = 0.2116.



(b) Quadrature with MSE = 0.2116.

Figure 4.3 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 2dB and 16-QAM signal constellation.



(a) In-phase with MSE = 0.0415. (b) Quadrature with MSE = 0.0415.

Figure 4.4 Distribution of the equalization error for the 11-TAP channel with ideally known CSI after the initial iteration at a received SNR of 14dB and 16-QAM signal constellation.

System error performance with ideally known CSI for the 11-TAP channel with 4-state $(5, 7)$ and 64-state $(133, 171)$ convolutional codes is shown in Fig. 4.5. Observing the error performance for the 4-state $(5, 7)$ convolutional code, we notice that most of the gain is realized with 2 iterations. After 3 iterations, there is a gain of approximately 0.6dB and 2dB at a BER of 2×10^{-4} for QPSK and 16-QAM, respectively. For a fixed SNR, it appears that we get a larger performance improvement from using 3 iterations when a 16-QAM signal constellation is used instead of a QPSK signal constellation. A possible reason for this phenomenon may be the non-uniform energy levels and the larger number of signal constellation points in the 16-QAM signal constellation. Since the iterative process is effectively trying to remove the mean associated with each block of received signals using the calculated symbol APPs, if the signal constellation is large with non-uniform energy levels, it is more likely that the mean will have a more significant deviation from zero in comparison to a small signal constellation with uniform energy levels. However, as the number of constellation points increases, the Euclidean distance between the constellation points decreases, which increases the sensitivity to noise.

The error performance for the 64-state $(133, 171)$ convolutional code shows similar characteristics to those for the 4-state $(5, 7)$ convolutional code. However, due to the increase in the error correction capability, there is an improvement in error performance

of approximately 2dB and 0.5dB, at a BER of 2×10^{-4} for QPSK and 16-QAM, respectively, in comparison to the error performance for the 4-state $(5, 7)$ convolutional code. Here we note that the amount of coding gain is greater for QPSK than 16-QAM. This is because the symbol selector maps encoder outputs after two consecutive time instances to a symbol in the signal constellation for 16-QAM, which means each symbol error introduced in the equalization process can result in a maximum of 4 consecutive bit errors in the decoder. Furthermore, these errors can not be separated by the symbol interleaver. This is a good motivation for the use of bit-interleaved coded modulation (BICM) for 16-QAM. On the other hand, encoder outputs are mapped to a symbol in the signal constellation directly for QPSK. So, each symbol error now corresponds to a maximum of 2 bit errors in the decoder. Therefore, if we increase the error correcting capabilities in the system, the coding gain will be greater when a QPSK signal constellation is used compared to 16-QAM.

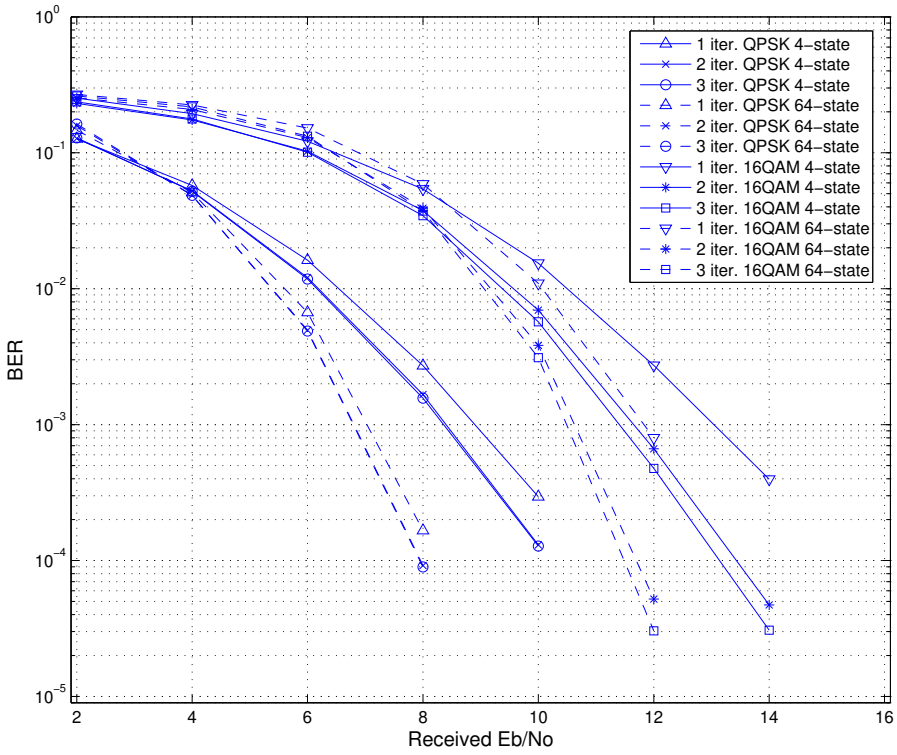


Figure 4.5 BER performance of the FD-TLE system for the 11-TAP channel with 1/2-rate 4-state $(5, 7)$ and 64-state $(133, 171)$ convolutional codes and perfectly known CSI.

4.1.2 GLMS Estimator Performance

Before simulating the overall system, the performance of the GLMS channel estimator is assessed. This is achieved by simulating the system in the training phase followed by the transmission mode but with only the UW prefix being transmitted. The number of UW prefix sequences transmitted in the transmission mode, which constitutes the MSD measure of the estimated CSI is set to 232, and the number of initial CSI estimate updates in the training mode, denoted by L_T, may be varied. The mean square deviation (MSD) error is used as a performance metric and is defined by [37, 38]

$$MSD = E\left[||\mathcal{H}_l - \widehat{\mathcal{H}}_{l|l-1}||^2\right], \tag{4.3}$$

where $||\cdot||$ denotes the Euclidean norm. We note that since the channel tap gains are symbol spaced, estimation of the composite channel impulse response is the same as estimating the tap gains of the channel impulse response $c(t)$. Therefore, we may substitute the equivalent $\mathcal{C}_l$ for $\mathcal{H}_l$ to obtain

$$MSD = E\left[||\mathcal{C}_l - \widehat{\mathcal{C}}_{l|l-1}||^2\right]. \tag{4.4}$$

The number of initial offline recursions for computing the intermediate matrix for the GLMS estimator is set to 500 with $\lambda = 0.975$ and $\zeta = 0.1$. The results for $L_T = 52, 104, 156$ and 208 are shown in Fig. 4.6, Fig. 4.7, Fig. 4.8 and Fig. 4.9, respectively.

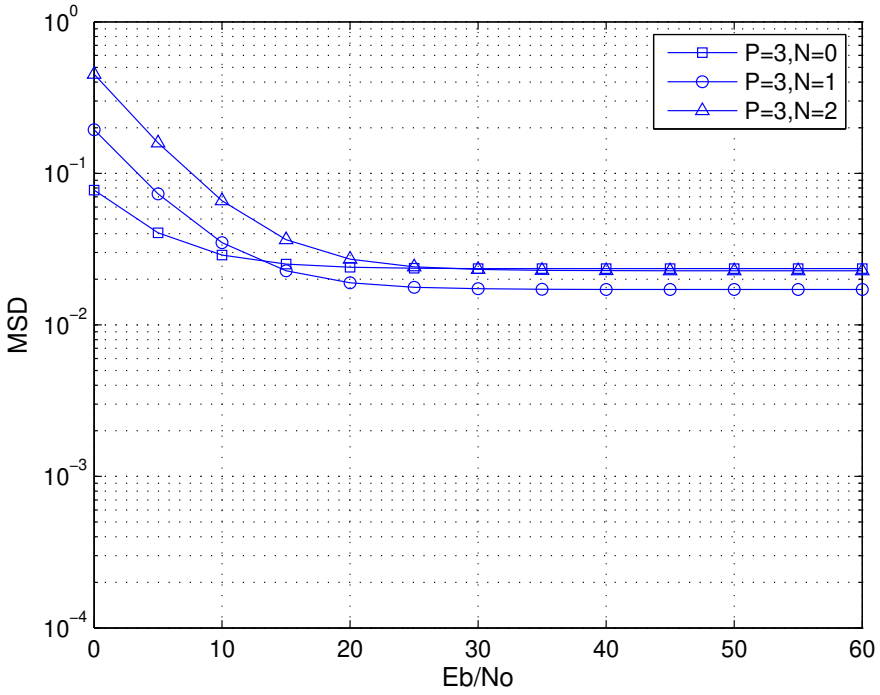


Figure 4.6 GLMS estimator performance for the 11-TAP channel with $L_T = 52$.

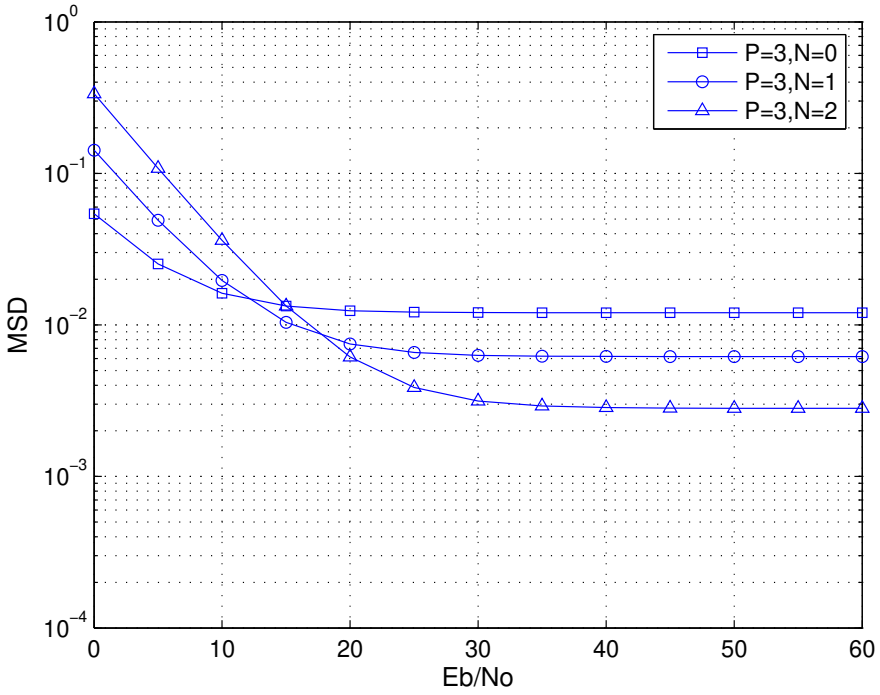


Figure 4.7 GLMS estimator performance for the 11-TAP channel with $L_T = 104$.

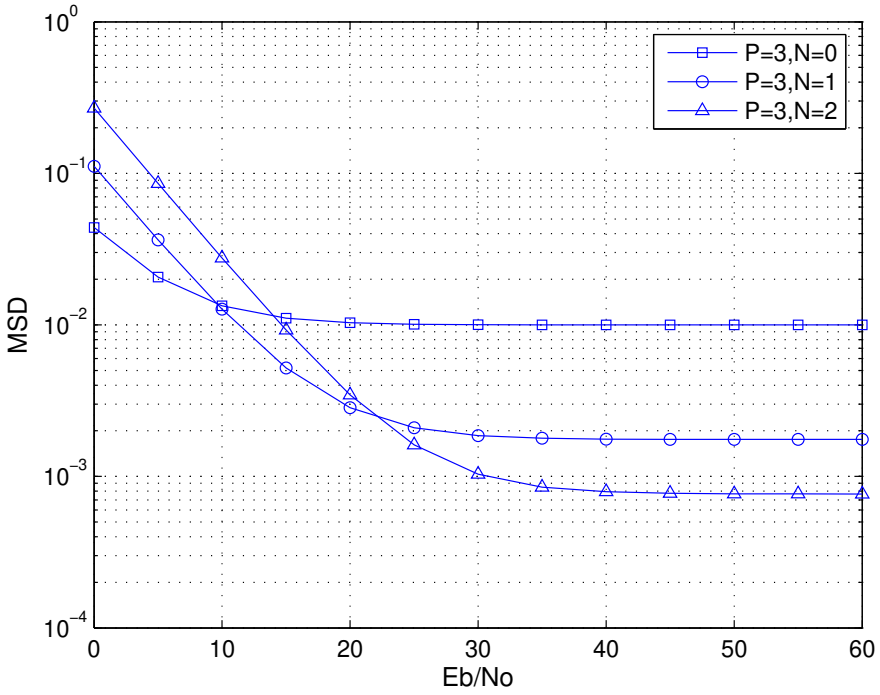


Figure 4.8 GLMS estimator performance for the 11-TAP channel with $L_T = 156$.

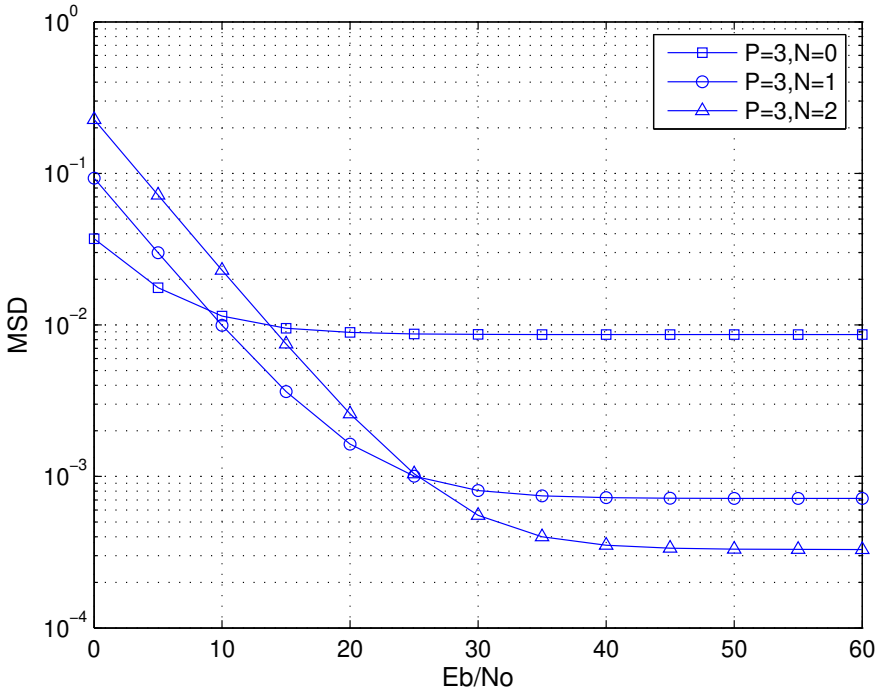


Figure 4.9 GLMS estimator performance for the 11-TAP channel with $L_T = 208$.

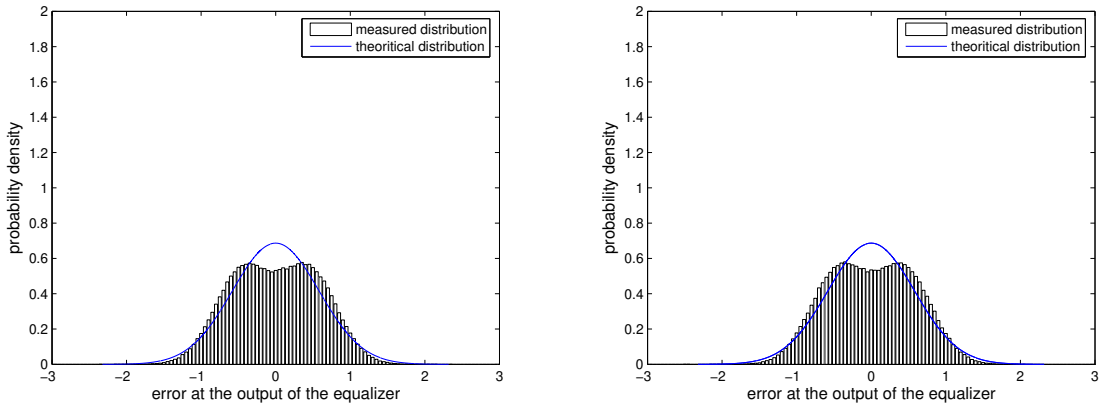
Observing the results we notice that in general, as L_T increases, the estimator performance improves especially for $P_G = 3$ and $N_G = 1, 2$ at high SNRs. The estimator settings of $P_G = 3$ and $N_G = 0$ perform best for SNRs up to around 10dB and then $P_G = 3$ and $N_G = 1$ perform best for SNRs up to around 20dB for $L_T = 156$. For SNRs beyond 20dB, the estimator setting of $P_G = 3$ and $N_G = 2$ perform best. Since most realistic high rate systems operate above 20dB SNR, the channel estimator should use $P_G = 3$ and $N_G = 2$. However, for system evaluation purposes, we are also interested in the SNR regions below the operating SNR value. Therefore, the setting of $P_G = 3$ and $N_G = 1$ is chosen for the evaluation of the overall system error performance for the 11-TAP channel. Furthermore, for $L_T \geq 156$, the performance of the estimator alters only slightly up to 20dB. Hence, the number of initial training symbols for the overall system is set to 156. At 20dB SNR with $L_T = 156$, the estimator MSD is approximately 3×10^{-3}. Table. 4.1 summarizes the estimator parameters chosen for the simulation of the overall system.

Table 4.1 GLMS estimator parameters for the 11-TAP channel.

Parameter	Value
No. of offline recursions	500
Forget factor (λ)	0.975
ζ	0.1
Predictor length (P_G)	3
Predictor order (N_G)	1
No. of training symbols (L_T)	156

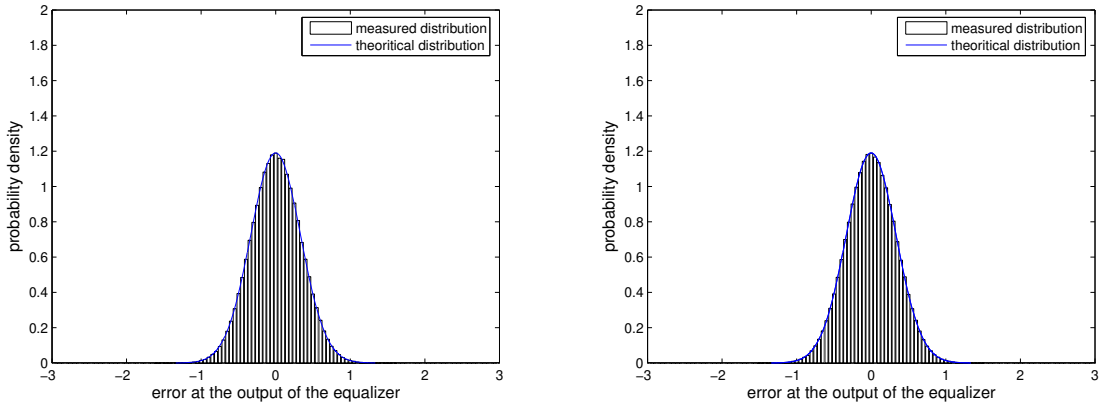
4.1.3 Overall System Error Performance

Before simulating the overall system performance for the 11-TAP channel, we conducted simulations to determine the distribution of the equalization error similar to what was shown in Section 4.1.1. Histograms of the equalization error at various received SNRs are shown in Fig. 4.10, 4.11, 4.12 and 4.13.



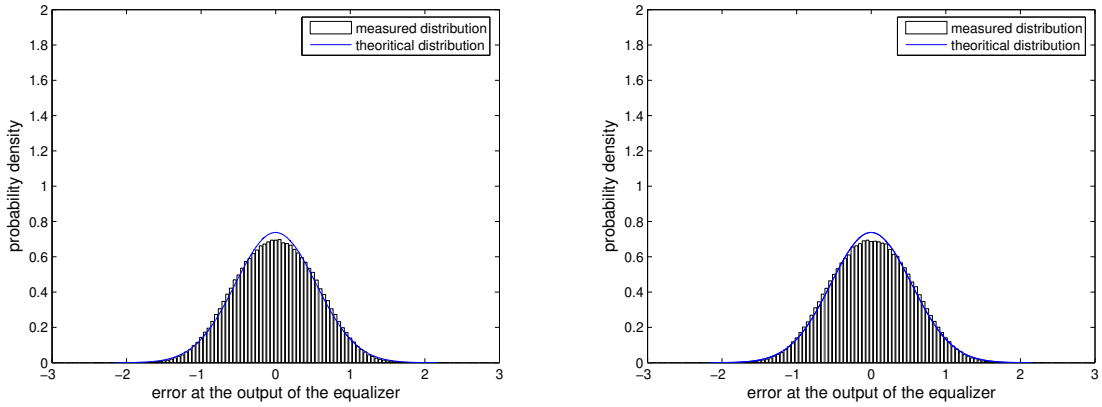
(a) In-phase with MSE = 0.3374. (b) Quadrature with MSE = 0.3374.

Figure 4.10 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation.



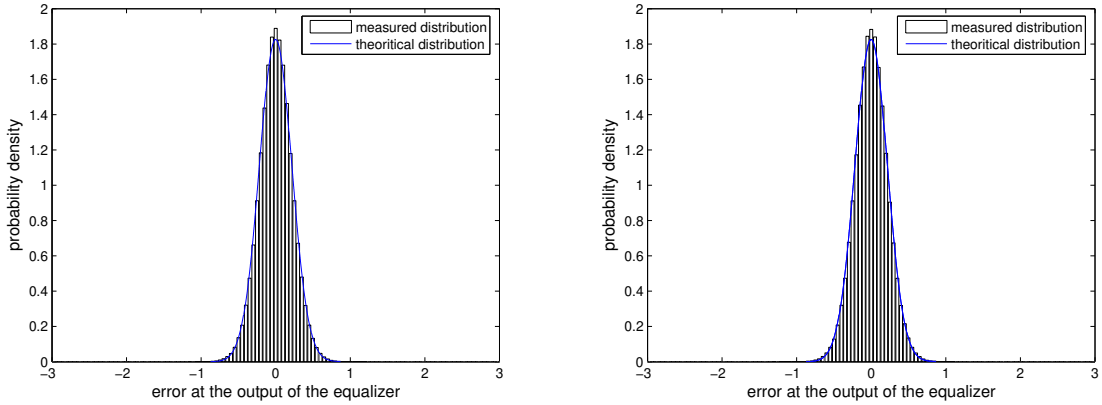
(a) In-phase with MSE = 0.1123. (b) Quadrature with MSE = 0.1123.

Figure 4.11 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation.



(a) In-phase with MSE = 0.2920. (b) Quadrature with MSE = 0.292.

Figure 4.12 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 2dB and 16-QAM signal constellation.



(a) In-phase with MSE = 0.0477.

(b) Quadrature with MSE = 0.0477.

Figure 4.13 Distribution of the equalization error for the 11-TAP channel with estimated CSI after the initial iteration at a received SNR of 16dB and 16-QAM signal constellation.

Comparing the histogram and the theoretical Gaussian distribution, we can see that there is a close resemblance especially at higher SNRs. For QPSK at low SNR, the distribution of the equalization error deviates slightly from the theoretical Gaussian distribution around the mean. However, it is still very similar overall in the tails of the distribution. This demonstrates that the assumption of a Gaussian equalization error is reasonable when estimated CSI is used in the equalizer.

System error performance when using a channel estimator with the settings of Table 4.2 for the 11-TAP channel employing the 4-state $(5, 7)$ or 64-state $(133, 171)$ convolutional code is shown in Fig. 4.14 and Fig. 4.15, respectively. Note that 156 initial training symbols were used, which corresponds to a delay of $156 \times 1\mu s = 156\mu s$ before the transmission of data.

The simulation results illustrate that when only estimated CSI is available at the receiver, the FD-TLE process improves performance using 2 iterations, but almost no additional gain is achieved with 3 iterations. Moreover, the amount gained by the second iteration has decreased in comparison to the perfect CSI case. When the 4-state $(5, 7)$ convolutional code is used, there is a performance degradation compared to ideal CSI case at a BER of 10^{-4} of approximately 2dB and 4dB for QPSK and 16-QAM, respectively, after 3 iterations. Similar degradation is present when the 64-state $(131, 171)$ convolutional code is used.

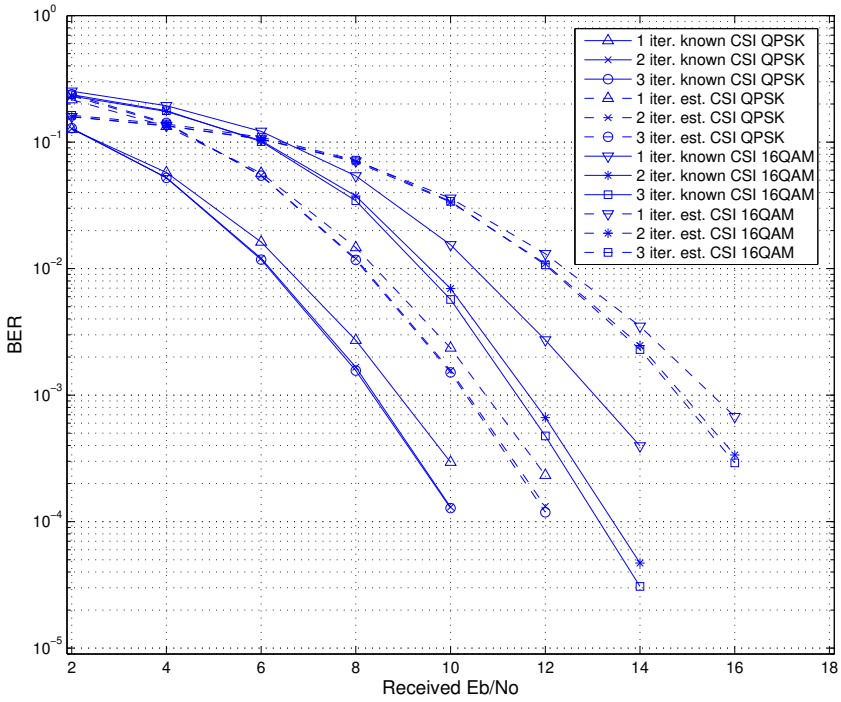


Figure 4.14 Overall system performance for the 11-TAP channel with the 1/2-rate 4-state $(5, 7)$ convolutional code.

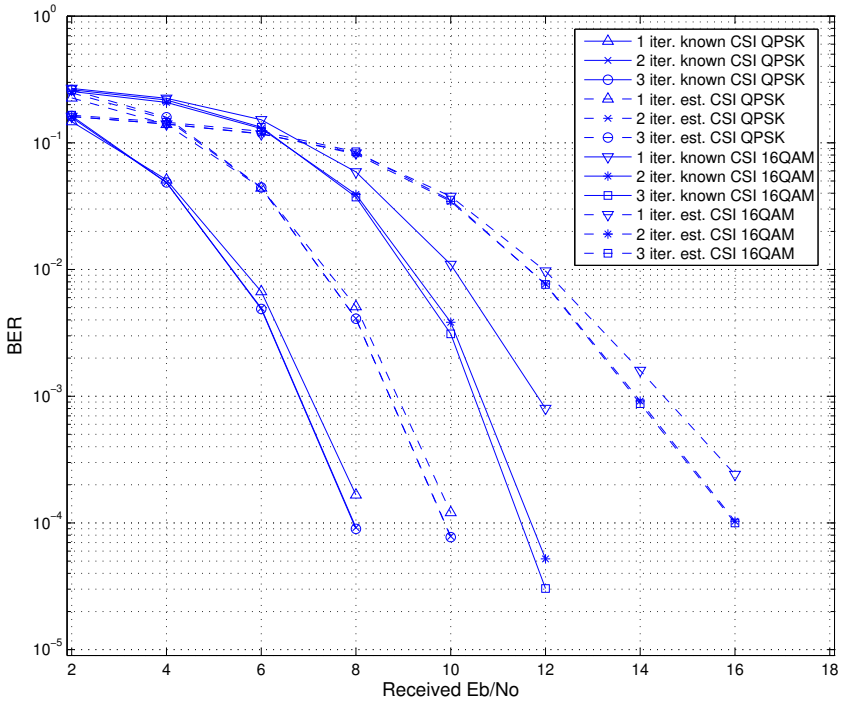


Figure 4.15 Overall system performance for the 11-TAP channel with the 1/2-rate 64-state $(133, 171)$ convolutional code.

4.2 FIXED WIRELESS CHANNEL

As described in Chapter 1, the SUI-5 channel spans $10\mu s$ which corresponds to 10 symbol intervals. Although the SUI-5 channel only contains 3 tap gains at 0, 5 and $10\mu s$, the channel estimator estimates the channel assuming that tap gains exist for every $1\mu s$ duration. This is similar to the case with the 11-TAP channel. However, the tap gains that are of zero value will be estimated to have values close to zero, provided that the estimator is tracking the channel correctly. Hence, the UW sequence used to train the channel estimator is also set to have length $N_{UW} = 25$. Following the same argument as before for the 11-Tap channel, the amount of overhead for each transmission for the SUI-5 channel is

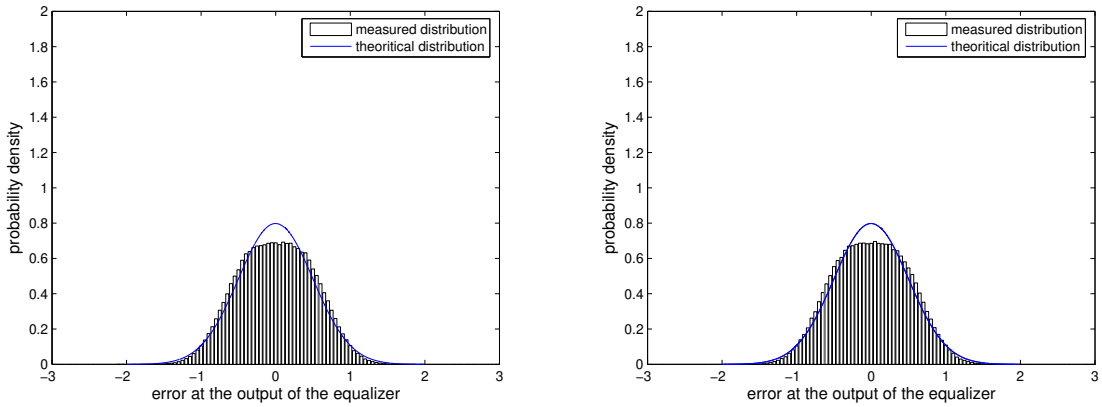
$$\mathcal{O}_{SUI-5} = \frac{25 \times 2}{1049} \times 100\% = 4.77\%, \tag{4.5}$$

with the number of encoded channel symbols being $N_{Cs} = N - N_{UW} = 999$.

The sampling rate for the SUI-5 channel is set as follows. First, we sample once for the duration of every transmission frame rather than for every symbol interval. Second, since the symbol interval is set at $1\mu s$, each transmission frame corresponds to a duration of $1049 \times 1\mu s = 1.049ms$. Third, using this value as the sampling period, the sampling rate is calculated to be $f_s = (1.049 \times 10^{-3})^{-1} = 953.29\text{Hz} \cong 950\text{Hz}$.

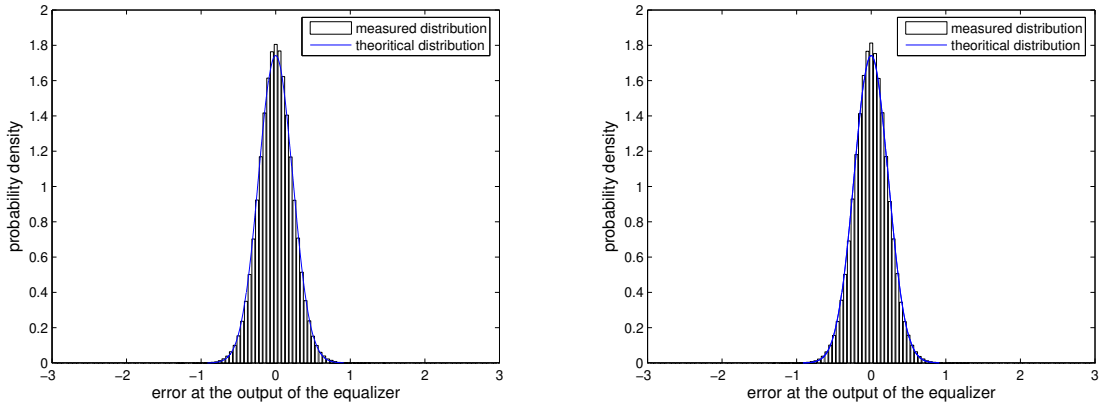
4.2.1 System Error Performance with ideally known CSI

Similar to the 11-TAP channel, we conduct simulations to investigate the distribution of the equalization error before simulating the system performance for the SUI-5 channel with ideally known CSI. Again, we transmit blocks of symbols chosen randomly from the signal constellation of interest (no coding) and plot histograms of the equalization error at various received SNRs as shown in Fig. 4.16, 4.17, 4.18 and 4.19. The solid line is the Gaussian density function with zero-mean and the mean squared value of the equalization error (MSE) as the variance. The estimation errors obtained as a result of 2000 transmissions were plotted in a histogram form with the area being normalized to unity.



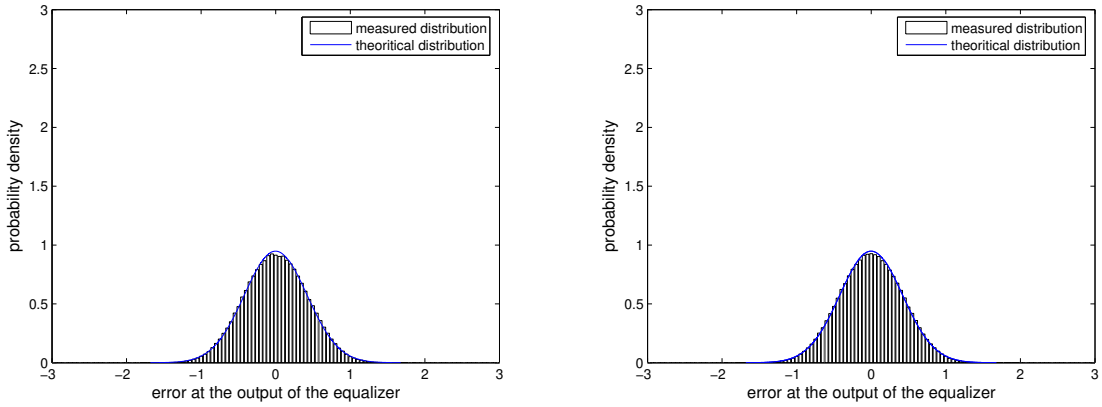
(a) In-phase with MSE = 0.24937. (b) Quadrature with MSE = 0.24937.

Figure 4.16 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation.



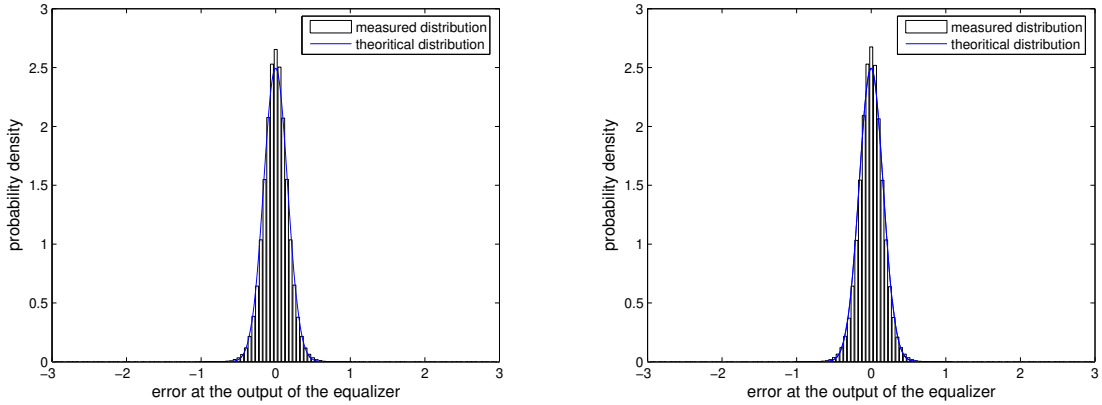
(a) In-phase with MSE = 0.05244. (b) Quadrature with MSE = 0.05244.

Figure 4.17 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation.



(a) In-phase with MSE = 0.17668. (b) Quadrature with MSE = 0.17668.

Figure 4.18 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a 2dB and 16-QAM signal constellation.



(a) In-phase with MSE = 0.02561. (b) Quadrature with MSE = 0.02561.

Figure 4.19 Distribution of the equalization error for the SUI-5 channel with ideally known CSI after the initial iteration at a 14dB and 16-QAM signal constellation.

Comparing the histogram and the solid line, we can see that there is a close resemblance, which shows that the assumption of a zero-mean Gaussian distribution for the equalization error is reasonable.

System error rate performances with ideally known CSI for the SUI-5 channel with 4-state $(5, 7)$ and 64-state $(133, 171)$ convolutional codes are shown is Fig. 4.20. Observing the error performance for the 4-state $(5, 7)$ convolutional code, we notice that most of the gain is obtained with 2 iterations. After 3 iterations, there is a gain of approximately 0.5dB and 1.5dB at a BER of 6×10^{-5} and 6×10^{-4} for QPSK and 16-QAM, respectively. For a fixed SNR, it appears that we get more improvement in performance after the second iteration using a 16-QAM signal constellation than a QPSK signal constellation. These results are similar to what we found for the 11-TAP channel. The error performance for the $(133, 171)$ convolutional code also shows that not much can be gained with 3 iterations. The coding gain obtained by using the $(131, 171)$ convolutional code as opposed to the $(5, 7)$ convolutional code is approximately 2dB at a BER of 10^{-4} for QPSK and 0.5dB at a BER of 10^{-4} for 16-QAM after 3 iterations.

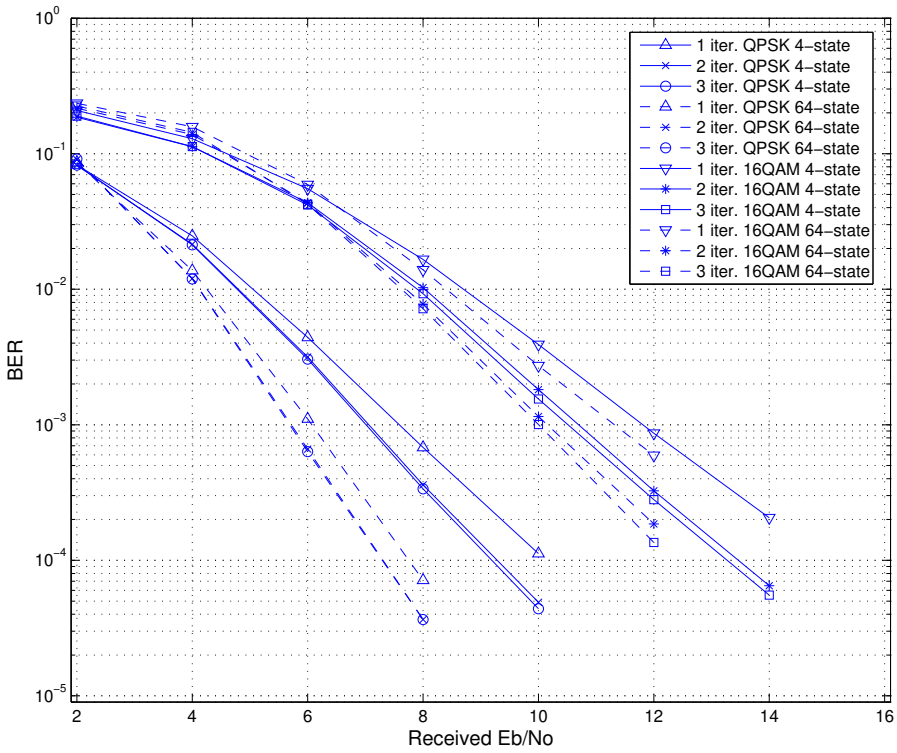


Figure 4.20 BER performance of the FD-TLE system for the SUI-5 channel with a 1/2-rate 4-state $(5, 7)$ convolutional code and 64-state $(133, 171)$ convolutional code and perfectly known CSI.

4.2.2 GLMS Estimator Performance

GLMS estimator performance for the SUI-5 channel is assessed using the same methodology described earlier for the 11-TAP channel. GLMS estimator performances for the SUI-5 channel with $L_T = 52, 104, 156$ and 208 are shown in Fig. 4.21, Fig. 4.22, Fig. 4.23 and Fig. 4.24, respectively.

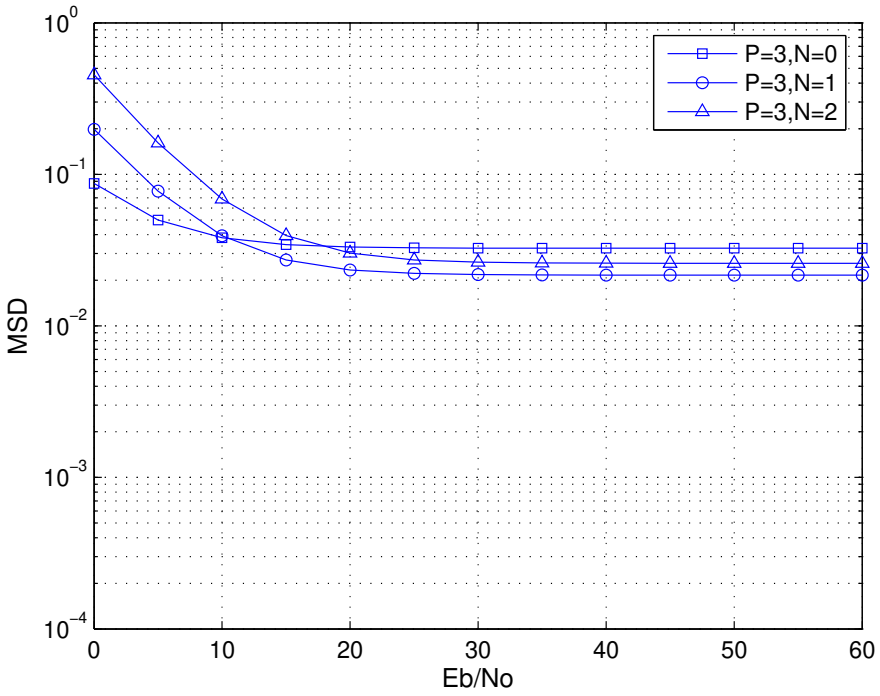


Figure 4.21 GLMS estimator performance for the SUI-5 channel with $L_T = 52$.

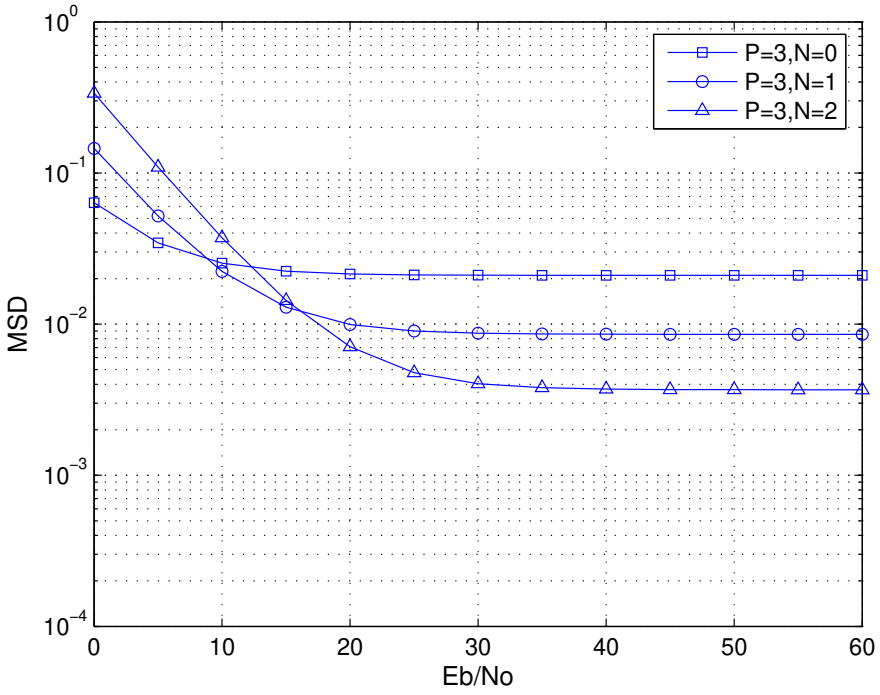


Figure 4.22 GLMS estimator performance for the SUI-5 channel with $L_T = 104$.

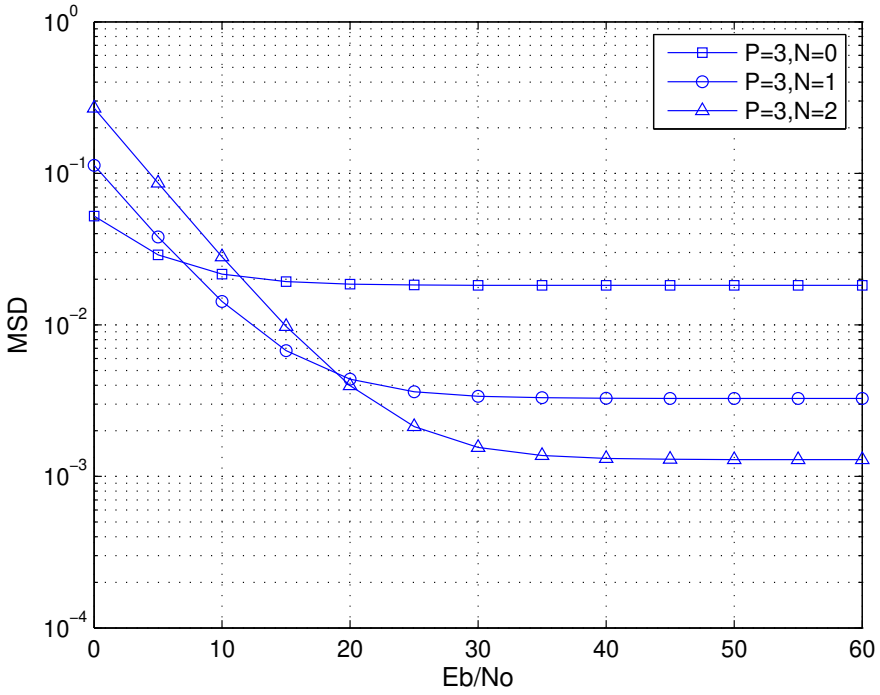


Figure 4.23 GLMS estimator performance for the SUI-5 channel with $L_T = 156$.

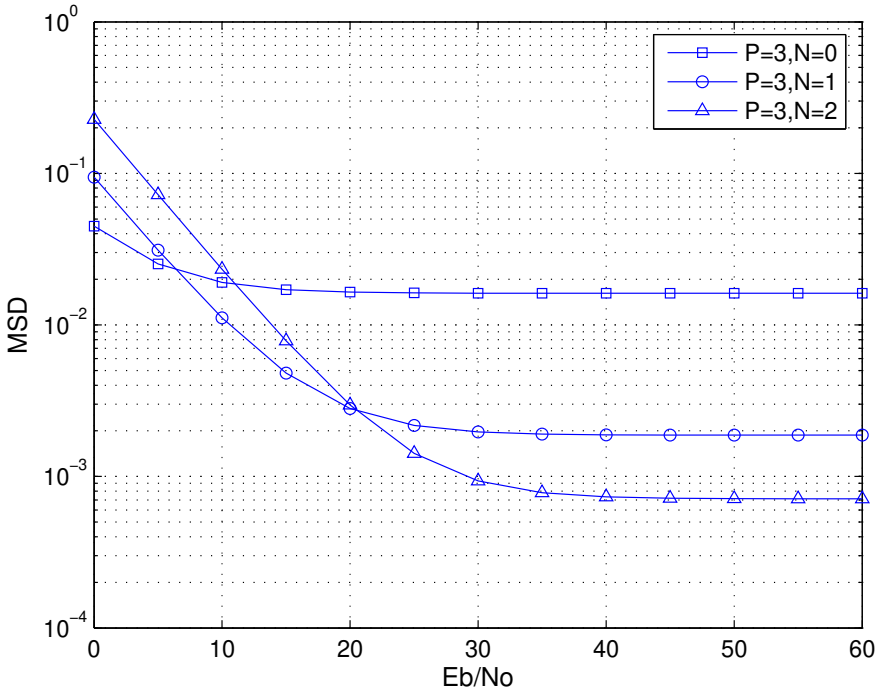


Figure 4.24 GLMS estimator performance for the SUI-5 channel with $L_T = 208$.

We notice that as L_T increases, the performance of the GLMS estimator for the SUI-5 channel follows similar trends to those observed for the 11-Tap channel. The estimator settings of $P_G = 3$ and $N_G = 0$ perform best up to around 10dB SNR and then $P_G = 3$ and $N_G = 1$ perform best up to around 20dB for $L_T = 156$. For SNRs beyond 20dB, the estimator setting of $P_G = 3$ and $N_G = 2$ perform best. Since most realistic high rate systems will operate above 20dB SNR, $P_G = 3$ and $N_G = 2$ should be set for the channel estimator. However, for error performance evaluation purposes in this thesis, we are interested in SNRs below the operating SNR. Therefore, the settings of $P_G = 3$ and $N_G = 1$ are chosen for the evaluation of overall system error performance for the SUI-5 channel. Furthermore, for $L_T \geq 156$, the performance of the estimator varies only slightly up to 20dB. Hence, the number of initial training symbols for the overall system is set to 156. At a SNR of 20dB for $L_T = 156$, the estimator MSD is approximately 5×10^{-3}. Table. 4.2 summarizes the estimator parameters chosen for the simulation of the overall system to evaluate the error performance.

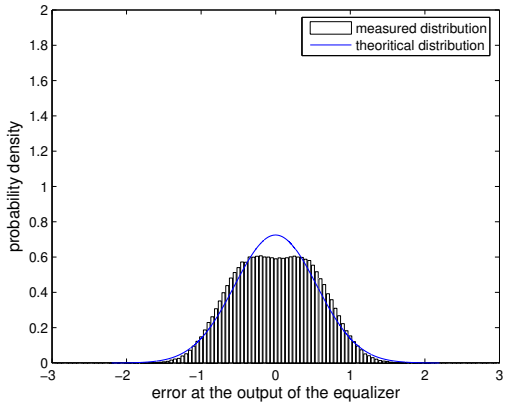
Table 4.2 GLMS estimator parameters for the SUI-5 channel.

Parameter	Value
No. of offline recursions	500
Forget factor (λ)	0.975
ζ	0.1
Predictor length (P_G)	3
Predictor order (N_G)	1
No. of training symbols (L_T)	156

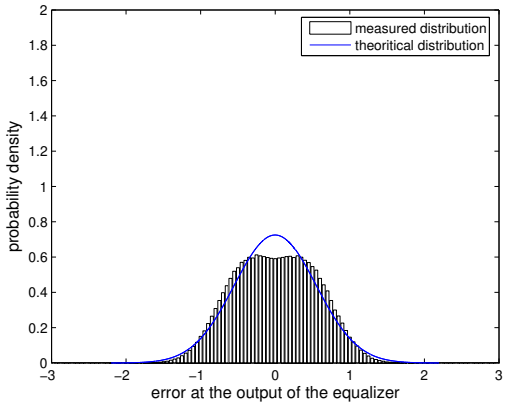
4.2.3 Overall System Error Performance

Before simulating the overall system performance for the SUI-5 channel, we conducted simulations to investigate the distribution of the equalization error similar to what was done in Section 4.2.1. The histograms of the equalization error at various received SNRs are shown in Fig. 4.25, 4.26, 4.27 and 4.28. Comparing the histogram and to a theoretical Gaussian distribution, we can see that there is a close resemblance especially at higher SNRs. For QPSK at low SNR, the distribution of the equalization error

deviates slightly from the theoretical Gaussian distribution in the vicinity of the mean. However, it is still very similar overall around the tails. This shows that the assumption that the equalization error is Gaussian distributed is reasonable when estimated CSI is used in the equalizer.

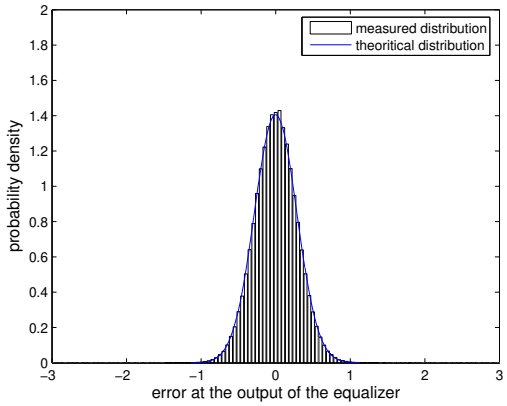


(a) In-phase with MSE = 0.3026.

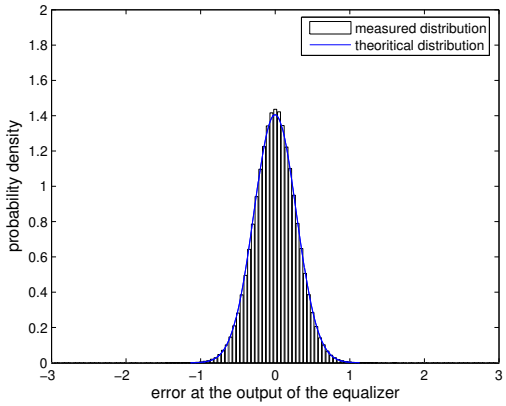


(b) Quadrature with MSE = 0.3026.

Figure 4.25 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 2dB and QPSK signal constellation.

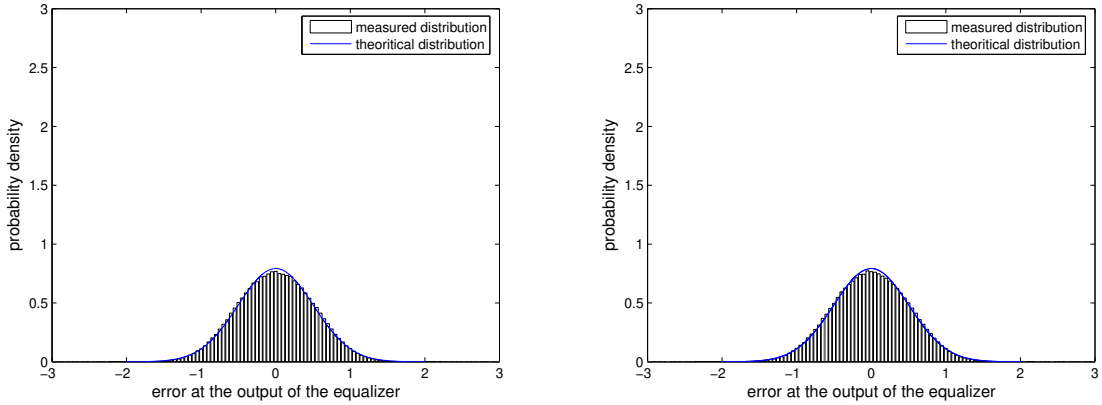


(a) In-phase with MSE = 0.0805.



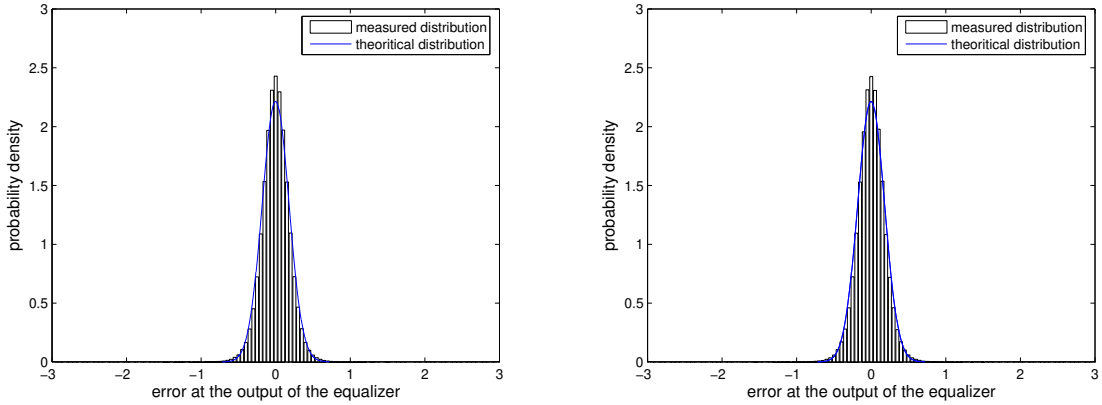
(b) Quadrature with MSE = 0.0805.

Figure 4.26 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 10dB and QPSK signal constellation.



(a) In-phase with MSE = 0.2533. (b) Quadrature with MSE = 0.2533.

Figure 4.27 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 2dB and 16-QAM signal constellation.



(a) In-phase with MSE = 0.0325. (b) Quadrature with MSE = 0.0325.

Figure 4.28 Distribution of the equalization error for the SUI-5 channel with estimated CSI after the initial iteration at a received SNR of 16dB and 16-QAM signal constellation.

Overall system error performance for the channel estimator settings of Table 4.2 for the SUI-5 channel and employing the 4-state $(5, 7)$ or 64-state $(133, 171)$ convolutional code is shown in Fig. 4.29 and Fig. 4.30, respectively. Similar to the results obtained for the 11-TAP channel, 156 initial training symbols were used, which corresponds to a delay of $156 \times 1\mu s = 156\mu s$ before the transmission of data.

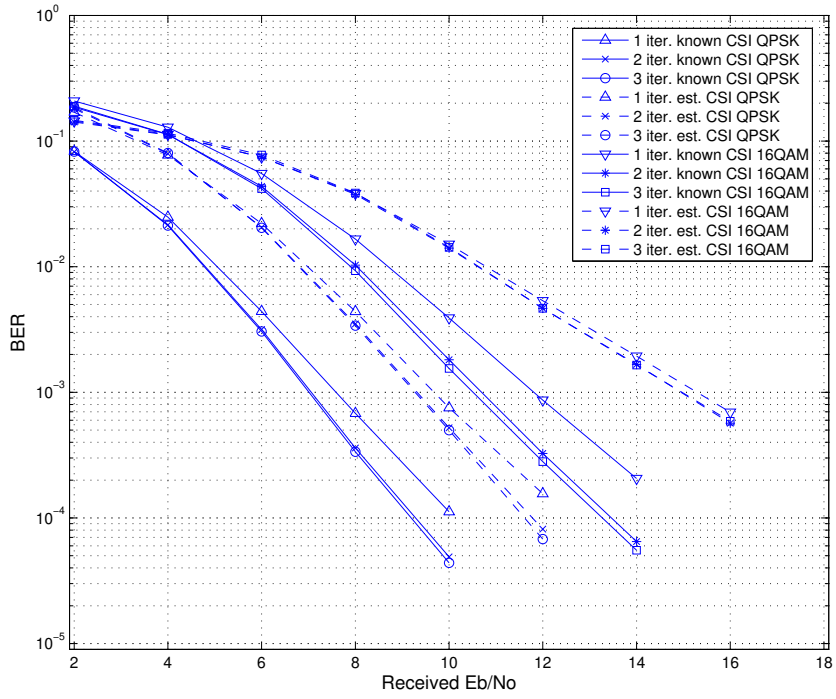


Figure 4.29 Overall system performance for the SUI-5 channel with the 1/2-rate 4-state $(5, 7)$ convolutional code.

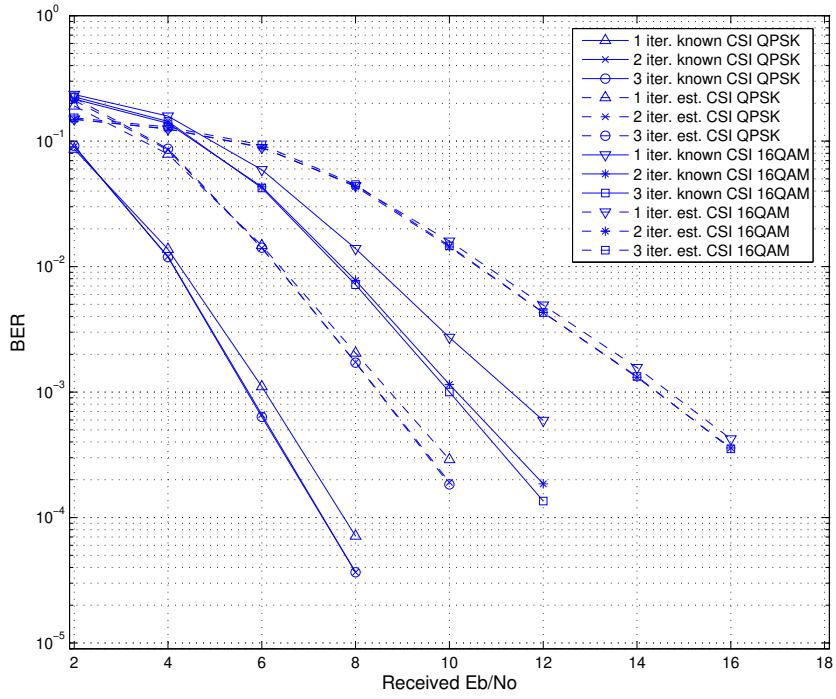


Figure 4.30 Overall system performance for the SUI-5 channel with the 1/2-rate 64-state $(133, 171)$ convolutional code.

The results shown in Fig. 4.29 and Fig. 4.30 illustrate that with estimated CSI, the FD-TLE receiver is still able to improve performance using 2 iterations, but not much additional performance can be gained for the third iteration. When the 4-state $(5, 7)$ convolutional code is used, there is a performance degradation of approximately 2dB at a BER of 10^{-4} for QPSK, and 4dB at a BER of 5×10^{-4} for 16-QAM, after 3 iterations. When the 64-state (131,171) convolutional code is used, there is a performance degradation of approximately 3dB at a BER of 10^{-4} for QPSK, and 5dB at a BER of 5×10^{-4} for 16-QAM, after 3 iterations. Similar to the results obtained for the 11-TAP channel, the amount of gain, when using estimated CSI decreases in comparison to the case with perfect CSI. We note that gain in performance is very small for 16-QAM with estimated CSI in comparison to that observed with QPSK. This suggests that the channel estimation error is sufficiently significant in the turbo equalization process that not much gain can be obtained by passing the symbol APPs back to the equalizer. It strongly suggests that the channel estimation error has to be decreased in order to realize significant gains from turbo equalization when using larger modulations schemes.

4.3 SUMMARY

GLMS channel estimator performance and overall system error performance for both the 11-TAP and SUI-5 channels using QPSK and 16-QAM signal constellations have been investigated in this chapter by means of computer simulation. It has been found that not much can be gained in the turbo equalization process from using a third iteration. Most of the gain is obtained from the second iteration for both the ideal or estimated CSI cases. There is a performance degradation of approximately $2-4$dB when estimated CSI is used in the receiver. This is caused by the channel estimation error influencing the equalization process. Furthermore, the amount of gain from the turbo equalization process decreases in comparison to that when perfect CSI is available.

Chapter 5

CONCLUSION

5.1 SUMMARY

This thesis began with a discussion of wireless channels, focusing on the 11-TAP and SUI-5 channels. These channels represents two different wireless environments as the 11-TAP channel represents a mobile environment and the SUI-5 channel represents a fixed access environment.

Multipath propagation of signals in a wireless environment introduces ISI at the receiver. In order to recover the transmitted signals correctly, equalization is required at the receiver. Usually a channel estimator is also required at the receiver as the equalizer requires some knowledge of the channel response.

In Chapter 2, a background literature search of various turbo equalization schemes and channel estimation algorithms was presented. The frequency-domain turbo equalization scheme of [32] and the GLMS channel estimation algorithm of [37] were emphasized as they are used in the proposed receiver structure.

In Chapter 3 various components of the proposed system were defined and the integration of the channel estimator with the equalizer was discussed. The channel estimator requires knowledge of the maximum delay spread of the channel, and uses pilot sequences embedded in a UW prefix to update the channel estimate. Following the derivation of the equalizer filter matrix of [32], a detailed derivation of the equalizer filter matrix using estimated CSI was presented. It explicitly showed the effect of channel estimation error.

In Chapter 4 system error performance was evaluated using computer simulation

for both the 11-TAP and SUI-5 channels with QPSK and 16-QAM signal constellations. It was found that three iterations between the equalizer and the decoder is sufficient. In fact, most of the gain due to the iterative process is from the second iteration. In addition, the gain due to turbo equalization using estimated CSI was observed to be smaller than the gain obtained with perfect CSI. Simulation results also indicated that good bit error performance is achievable using estimated CSI, with a loss ranging from 2dB to 5dB SNR in comparison to the ideal case. The amount of loss is dependent on the channel and the signal constellation of interest. Lastly, the results clearly demonstrate that improved performance of the turbo equalizer requires more accurate channel estimation. This likely to result in increased complexity. For example, if the generalize recursive least squares (GRLS) algorithm rather than the current GLMS approach were to be used, increased accuracy in channel estimation would be obtained at the cost of an increase in system complexity.

A point worth mentioning is that the overall performance could be improved through the use of more powerful coding techniques. The (5,7) and (133,171) convolutional codes chosen for the proposed system are relatively basic. They could be replaced with either a turbo code or a low density parity check code. Again this would lead to increased system complexity.

As a final remark, the proposed system is highly flexible. It may be applied to other channels provided that the maximum delay spread of the channel is upper bounded at the receiver. Increasing the signal constellation size can increase the throughput, but reduces the coding gain due to errors not sufficiently separable in the interleaving process. Reducing the frame length can reduce the computational complexity for each transmission block, but the amount of overhead for each block will increase. On the other hand, if the frame length is set too long, the assumption of a quasi-static channel is no longer valid. All these factors pose tradeoff in the design of the system and should be carefully considered.

5.2 FUTURE WORK

Block equalization approaches such as the one presented in this thesis assumes perfect timing synchronization at the receiver. Therefore, the effect of timing errors would be an interesting area for further research.

Actual measurements of the channel medium could be used to improve the accuracy of computer simulations. The simulation results obtained from actual channel measurements would be useful for someone who wants to apply this system in a particular scenario.

The work presented has been theoretical in nature involving analysis and computer simulation. The next logical development would be its implementation in DSP hardware.

As seen from the simulation results, the GLMS channel estimator performance varies for different predictor lengths and orders at different SNR values. So, for optimum system performance, it would be beneficial to look at ways of determining the channel estimator parameters while the system is in operation.

In the work presented in [52], the turbo equalization scheme of [32] has been extended to multiple-input multiple-output (MIMO) systems under the assumption that perfect CSI is available at the receiver. Therefore, it is possible to extend the work presented in this thesis to MIMO systems where the CSI is estimated. However, such a study is beyond the scope of the present project.

Appendix A

SUPPLEMENTARY SIMULATION RESULTS

Computer simulations were set up to ensure correct implementation of the turbo equalization algorithm of [32] and the GLMS channel estimator of [37] before incorporating them in the proposed system. Following the simulation parameters specified in [32] and [37], Fig. A.1 and Fig. A.2 show the simulation results in comparison to the published results for the turbo equalization and the GLMS channel estimator, respectively.

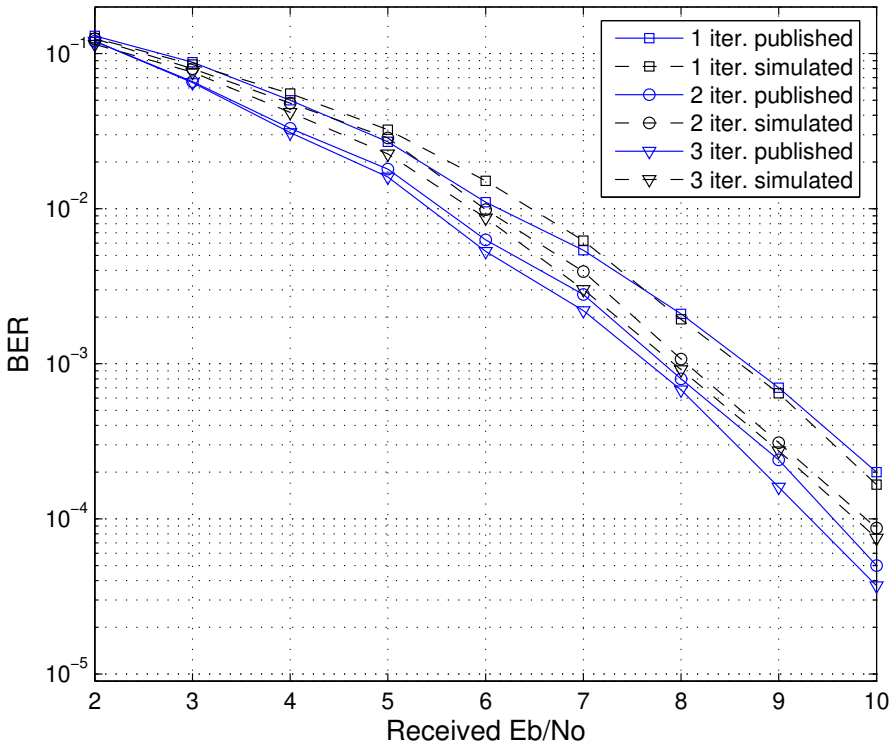


Figure A.1 FD-TLE error performance comparison.

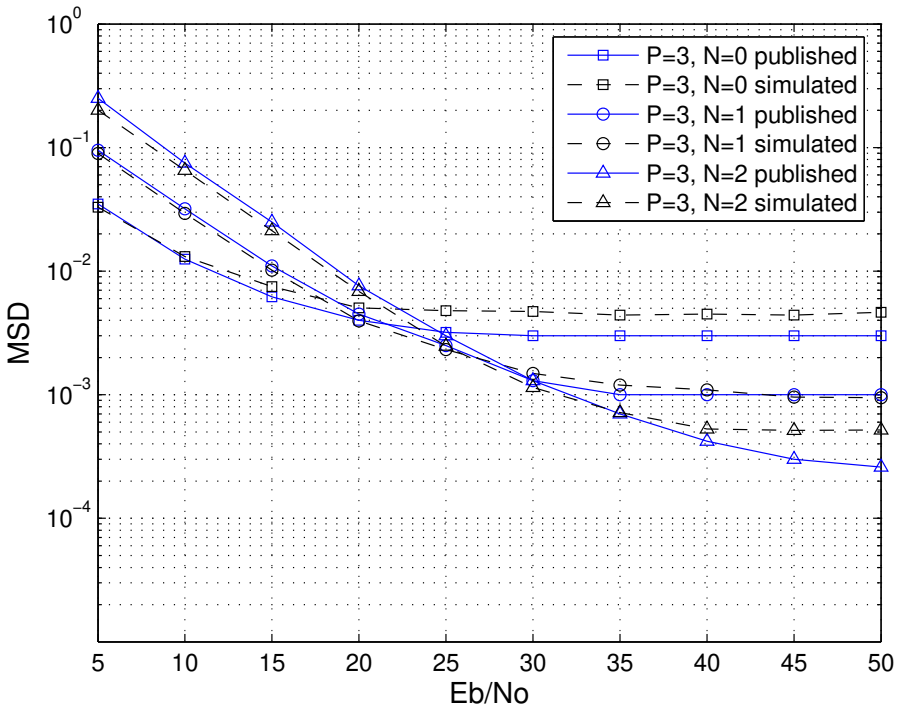


Figure A.2 GLMS estimator performance comparison with 26 training symbols.

Observing Fig. A.1 and Fig. A.2, we see that although the simulation results differ slightly to the published results, the overall performance curves are of close proximity to the published performance curves. It is possible that the slight difference in performance curves occurred due to the variation of simulation tools and random number generators employed.

REFERENCES

[1] A. Goldsmith, *Wireless Communications*. New York: Cambridge University Press, 2005.

[2] T. Rappaport, *Wireless Communications: Principles and Practice*. New Jersey: Prentice Hall, 2002.

[3] S. K. Majumdar, I. Vogelsang, and M. E. Cave, *Handbook of Telecommunications Economics Volume 2 - Technology Evolution and the Internet*. The Neatherlands: Elsevier B.V., 2002.

[4] B. P. Lathi, *Modern Digital and Analog Communication Systems*. New York: Oxford University Press, 1998.

[5] A. A. Huuredeman, *The Worldwide History of Telecommunications*. New Jersey: John Wiley and Sons, 2003.

[6] M. E. Cave, S. K. Majumdar, and I. Vogelsang, *Handbook of Telecommunications Economics Volume 1 - Structure, Regulation and competition*. The Neatherlands: Elsevier B.V., 2002.

[7] M. C. Jeruchim, P. Balaban, and K. S. Shanmugan, *Simulation of Communication Systems*. New York: Plenum Press, 1992.

[8] S. Haykin, *Communication Systems*. New York: John Wiley and Sons, 2001.

[9] H. Stark and J. Woods, *Probability and Random Processes with Applications to Signal Processing*. New Jersey: Prentice Hall, 2002.

[10] E. Arthurs and H. Dym, "On the optimum detection of digital signals in the presence of white gaussian noise - a geometric interpretation and a study of three basic data transmission systems," *IEEE Trans. Commun.*, vol. 10, pp. 336–372, Dec. 1962.

[11] R. Clarke, "A statistical theory of mobile-radio reception," *Bell Syst. Tech. J.*, vol. 47, pp. 957–1000, Dec. 1968.

[12] K. Pahlavan and A. H. Levesque, *Wireless Information Networks*. New York: John Wiley and Sons, 1995.

[13] P. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Commun.*, vol. CS-11, pp. 360–393, Dec. 1963.

[14] W. Jakes, *Microwave Mobile Communications*. New York: IEEE Press, 1974.

[15] M. Luise, R. Reggiannini, and G. Vitetta, "Blind equalization/detection for OFDM signals over frequency-selective channels," *IEEE Commun. Magazine*, vol. 40, pp. 58–66, Apr. 2002.

[16] D. Verdin and T. Tozer, "Generating a fading process for the simulation of land-mobile radio communications," *IEE Electronics Letters*, vol. 29, pp. 58–66, Nov. 1993.

[17] V. Erceg and et al, "Channel models for fixed wireless applications," *IEEE 802.16a cont. IEEE 806.16.3c-01/29r1*, Feb. 2001.

[18] D. Falconer, S. Ariyavisitakul, A. Benjamin-Seeyar, and A. Edison, "Frequency domain equalization for single-carrier broadband wireless systems," *IEEE Commun. Magazine*, vol. 40, pp. 58–66, Apr. 2002.

[19] V. Erceg, L. Greenstein, S. Tjandra, S. Parkoff, A. Gupta, B. Kulic, A. Julius, and R. Bianchi, "An empirically based path loss model for wireless channels in suburban environments," *IEEE J. Select. Areas Commun.*, vol. 17, pp. 1205–1211, Jul. 1999.

[20] U. W. Pooch, D. Machuel, and J. McCahn, *Telecommunications and Networking*. Florida: CRC Press, 1991.

[21] J. Proakis, *Digital Communications*. New York: McGraw-Hill, 2001.

[22] S. Chennakeshu and G. Saulnier, "Differential detection of $\pi/4$-shifted-dqpsk for digital cellular radio," *IEEE Trans. Vehicular Technology*, vol. 42, pp. 46–57, Feb. 1993.

[23] A. Ambardar, *Analog and Digital Signal Processing*. ITP: Books/Cole Publishing Company, 1999.

[24] C. Douillard, A. Picart, M. Jezequel, P. Didier, C. Berrou, and A. Galvieux, "Iterative correction of intersymbol interference: Turbo-equalization," *Euro. Trans. Commun.*, vol. 6, pp. 507–511, Sep. 1995.

[25] M. T*ü*chler and J. Hagenauer, "Turbo equalization using frequency domain equalizers," in *Proc. Allerton Conf.*, vol. 4, pp. 1234–1243, Oct. 2000.

[26] M. Tüchler and J. Hagenauer, "Linear time and frequency domain turbo equalization," in *Proc. VTC*, vol. 4, pp. 2773–2777, Oct. 2001.

[27] M. Tüchler, A. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Processing*, vol. 50, pp. 673–683, Mar. 2002.

[28] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, pp. 754–767, May. 2002.

[29] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization," *IEEE Signal Processing Magazine*, vol. 20, pp. 67–80, Oct. 2004.

[30] R. Otnes and M. Tüchler, "On iterative equalization, estimation, and decoding," *in Proc. ICC*, vol. 4, pp. 1918–1923, May. 2004.

[31] R. Otnes and M. Tüchler, "Iterative channel estimation for turbo equalization of time-varying frequency-selective channels," *IEEE Trans. Commun.*, vol. 3, pp. 2958–2962, Nov. 2003.

[32] F. Pancaldi and G. Vitetta, "Block channel equalization in the frequency domain," *IEEE Trans. Commun.*, vol. 53, pp. 463–471, Mar. 2005.

[33] S. Haykin, *Adaptive Filter Theory, 3rd Edition*. Prentice-Hall: Addison-Wesley, 1996.

[34] J. J. Shynk, "Frequency-domain and multirate adaptive filtering," *IEEE Signal Processing Magazine*, vol. 9, pp. 15–37, Jan. 1992.

[35] S. Song, A. Singer, and K.-M. Sung, "Soft input channel estimation for turbo equalization," *IEEE Trans. Signal Processing*, vol. 52, pp. III–2805–III–2808, Oct. 2004.

[36] T. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 3, pp. 47–60, Nov. 1996.

[37] W. Leon and D. Taylor, "The polynomial-based generalized least mean squares estimator for rician and rayleigh fading channels," in *Proc. VTC-Fall*, vol. 6, pp. 3861–3865, 2004.

[38] W. Leon and D. Taylor, "The polynomial-based generalized recursive least squares estimator for rayleigh fading channels," in *Proc. GLOBECOM*, vol. 5, pp. 2401–2405, 2003.

[39] D. Borah and B. Hart, "Frequency-selective fading channel estimation with a polynomial time-varying channel model," *IEEE Trans. Commun.*, vol. 47, pp. 862–873, Jun. 1999.

[40] G. C. Reinsel, *Elements of Multivariate Time Series Analysis*. New York: Springer-Verlag, 1993.

[41] L. M. Davis, I. B. Collings, and R. J. Evans, "Coupled estimators for equalization of fast-fading mobile channels," *IEEE Trans. Commun.*, vol. 46, pp. 1262–1265, Oct. 1998.

[42] S. Lin and D. Costello, *Error Control Coding*. New Jersey: Prentice Hall, 2004.

[43] S. B. Wicker, *Error Control Systems for Digital Communication and Storage*. New Jersey: Prentice Hall, 1995.

[44] "QUALCOMM announces single-chip K=7 viterbi decoder device," *IEEE Cummun. Magazine*, vol. 25, p. 75, Apr. 1987.

[45] B. Sklar, "A primer on turbo code concepts," *IEEE J. Select. Areas Commun.*, vol. 16, pp. 1865–1578, Oct. 1998.

[46] M. Bossert, *Channel Coding for Telecommunications*. England: John Wiley and Sons, 1999.

[47] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT-13 No. 2, pp. 260–268, Apr. 1967.

[48] G. Forney, "Final report on a coding system design for advanced solar missions," *Codex Corp.*, vol. Contract NAS2-3637, p. Appendix A, Dec. 1967.

[49] C. Heegard and S. B. Wicker, *Turbo Coding*. Norwell: Kluwer Academic Publishers, 1999.

[50] J. Coon, M. Beach, J. McGeehan, and M. Sandell, "Channel and noise variance estimation and tracking algorithms for unique-word based signal-carrier systems," *in Proc. 1st Int. Symp. Wireless Communication Systems*, pp. 51–55, Sep. 2004.

[51] L. Bahl, J. Cocke, F. Jelinek, and J. Riviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. 20, pp. 284–287, Mar. 1974.

[52] F. Pancaldi and G. Vitetta, "Frequency-domain equalization for spacetime block-coded systems," *IEEE Trans. Wireless Commun.*, vol. 4, pp. 2907–2916, Nov. 2005.