CPM Equalization to Compensate for ISI due to Band Limiting Channels

By

Andres Moctezuma

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

Dr. A.A. (Louis) Beex, Chairman
Dr. William Tranter
Dr. Virgilio Centeno

August 4, 2006
Blacksburg, Virginia

KEYWORDS: CPM, Equalization, Channel Truncation, Group-delay, MLSE

CPM Equalization to Compensate for ISI due to Band Limiting Channels

Andres Moctezuma

(Abstract)

In modern wireless communication systems, such as satellite communications and wireless networks, the need for higher data rates without the need for additional transmit power has made Continuous Phase Modulation (CPM) one of the most attractive modulation schemes in band limited channels. However, as the data rates keep increasing, the spectral width of the CPM signal increases beyond the channel bandwidth and performance becomes constrained by the intersymbol interference (ISI) that results from band-limiting filters.

We propose two approaches to the problem of equalization of band-limited CPM signals. First, our efforts are focused on shortening the channel impulse response so that we can use a low complexity MLSE equalizer. We implement the channel truncation structure by Falconer and Magee and adapt it to work with CPM signals. This structure uses a, a more derivable, pre-filter to shape the overall response of the channel, so that its impulse response is of shorter duration. Simulation results show that near-MLSE performance can be obtained while dramatically reducing MLSE equalizer complexity.

In our second approach, we focus on eliminating the group-delay variations inside the channel passband using an FIR pre-filter. We assume the channel to be time-invariant and provide a method to design an FIR filter so that – when convolved with the band limiting filter – it results in more constant group-delay over the filter passband. Results show that eliminating the group-delay variations in the band limiting filter passband reduce the amount of ISI and improve bit error rate performance.

Acknowledgments

I would like to express my sincere gratitude to my academic advisor Dr. A.A. (Louis) Beex for extending me the opportunity to be part of the DSPRL at Virginia Tech. I feel fortunate to have had the opportunity to work under his supervision and have had him as my advisor. For all his confidence, help and support I will be always grateful.

I would also like to thank my lab mate Takeshi Ikuma for his endless support and for being a great lab mate and friend. Team working with him in several DSPRL projects has been a great honor. His valuable insight and advice were very useful not only in the completion of this thesis, but also in my general understanding of many aspects of signal processing and communications.

I would like to extend my gratitude to Dr. William Tranter and Dr. Virgilio Centeno who kindly agreed to serve as Members of my Graduate Committee and for taking the time to review this work. In addition, I would like to thank Dr. Centeno and Dr. De La Ree for their refreshing encouragement, support and advice throughout my stay at Virginia Tech.

I would further like to thank the following people I had the opportunity of working with: Areg Baghdasaryan, Gleb Tcheslavski, Drs. Brandon and Jim Zeidler.

Lastly, I would also like to thank, my parents, my brother Rudy, and Lauren for their encouragement, support and unconditional love.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BLF	Band Limiting Filter
BPSK	Binary Pulse Shift Keying
CPFSK	Continuous Phase Frequency Shift Keying
CPM	Continuous Phase Modulation
DIR	Desired Impulse Response
FFT	Fast Fourier Transform
FIR	Finite Impulse Response
FMS	Falconer and Magee Structure
GDC	Group-delay Compensator
GMSK	Gaussian Minimum Shift Keying
ISI	Intersymbol Interference
MHCPM	Multi-H Continuous Phase Modulation
ML	Maximum Likelihood
MLSE	Maximum Likelihood Sequence Estimation
MSE	Minimum Square Error
MSK	Minimum Shift Keying
PF	Pre-Filter
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
VA	Viterbi Algorithm

1 Introduction

1.1 Research Motivation and Objective

In modern wireless communication systems, such as satellite communications and wireless networks, the need for higher data rates without the need for additional transmit power has made Continuous Phase Modulation (CPM) one of the most attractive modulation schemes. The reason for this is that not only do CPM signals have superior bandwidth characteristics (narrower bandwidth) over other modulation schemes, but they also exhibit superior bit-error-rate (BER) performance. The superior bandwidth characteristics of CPM signals arise from the phase being constrained to be continuous. The BER improvements come from the fact that – by careful selection of the CPM signal parameters – the Euclidean distance between possibly transmitted symbols can be increased and as a consequence the probability of making the correct decision also increases.

Even with the advantages of CPM signals, as the data rates keep increasing, the spectral width of the signal also increases. As the signal spectrum spreads, the performance becomes constrained by the intersymbol interference (ISI) that results from band-limiting filters. A band-limiting filter passes frequencies inside a limited frequency range and rejects the frequencies outside that frequency range. Band limiting filters are used on multiple-channel signals when each channel is to be processed individually to prevent adjacent channel interference or to prevent spectral leakage, as specified in some standards.

The relation between the spectrum of the CPM signal and the response of the channel as the data rate increases is shown in Figure 1.

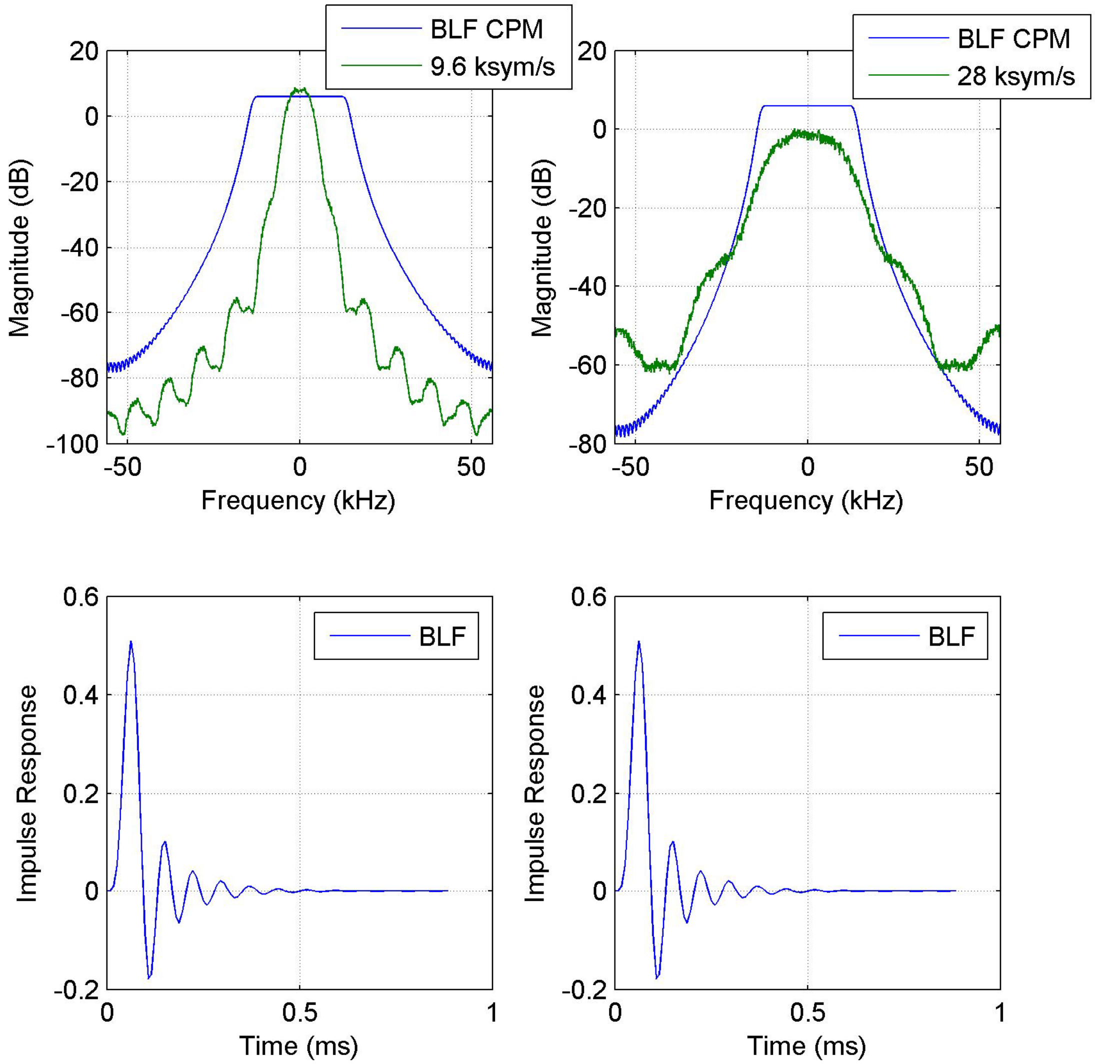

Figure 1. Channel response and CPM spectra for two different modulation rates.

The slower modulation rate (9.6 ksym/s) fits entirely within the channel passband. For this rate, the channel impulse response becomes essentially zero after 4 symbol periods. In contrast, for the higher modulation rate (28 ksym/s) the signal spectrum of the signal extends beyond the channel passband. The out-of-band portions of the signal become attenuated and thus some of the power in the signal is lost. The severity of the ISI, for the high data rates, is best illustrated by the extent of the channel impulse response. For the 28 ksym/s CPM, the channel impulse response extends over 10 symbol periods before becoming practically zero.

Figure 2 shows the distortion of the CPM signal, as produced by the band limiting filter, for both data rates presented in Figure 1. Since retrieving the phase information is critical for the decoding of CPM signals, the phase distortion is also shown.

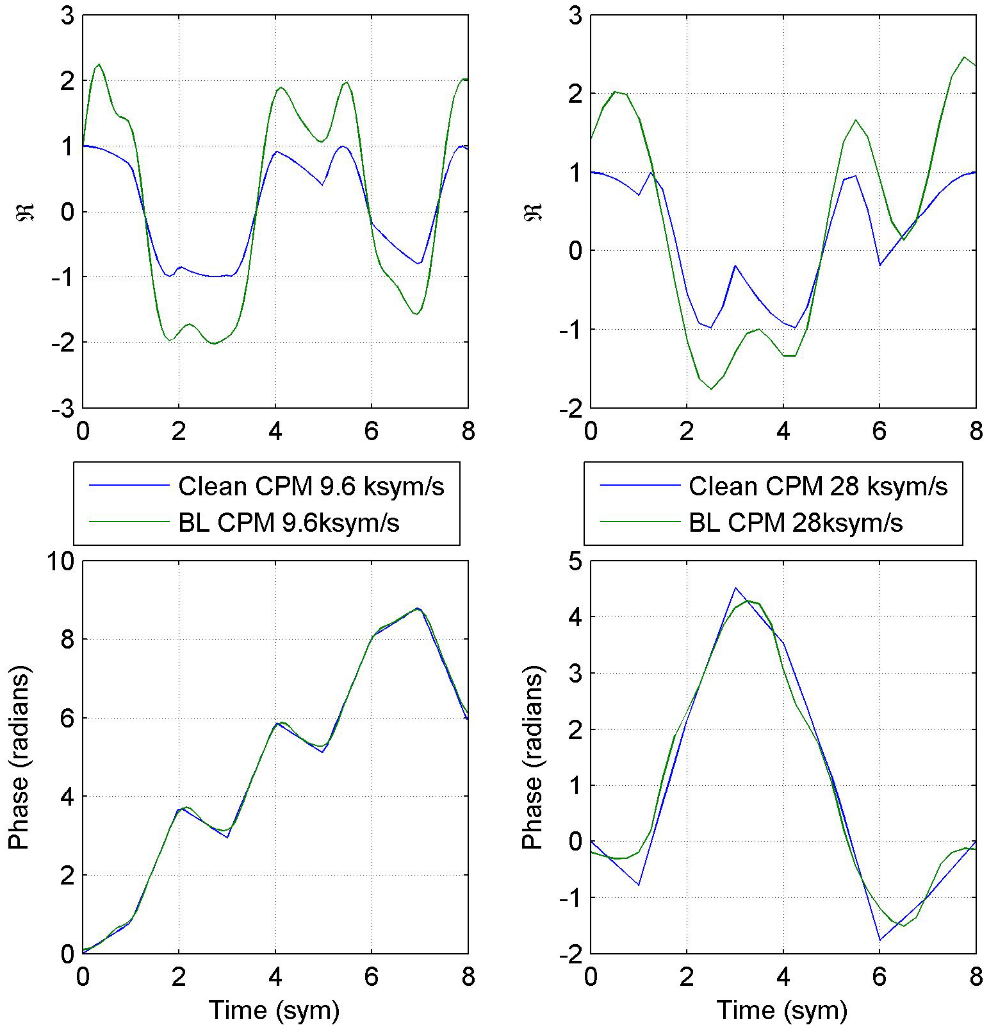

Figure 2. Signal distortion produced by band limiting filter at different modulation rates.

The distortion produced at the lower modulation rate is pretty much amplitude distortion. However as shown in the bottom left plot, the phase information is preserved. Contrary, at the higher modulation rates, the signal is distorted in both shape and magnitude. Even more critical is the loss in phase information as shown in the bottom right plot. The loss of phase information, in particular the phase slope, severely affects the performance of the CPM receiver when decoding the signal. The BER performance for the two modulation rates is shown in Figure 3.

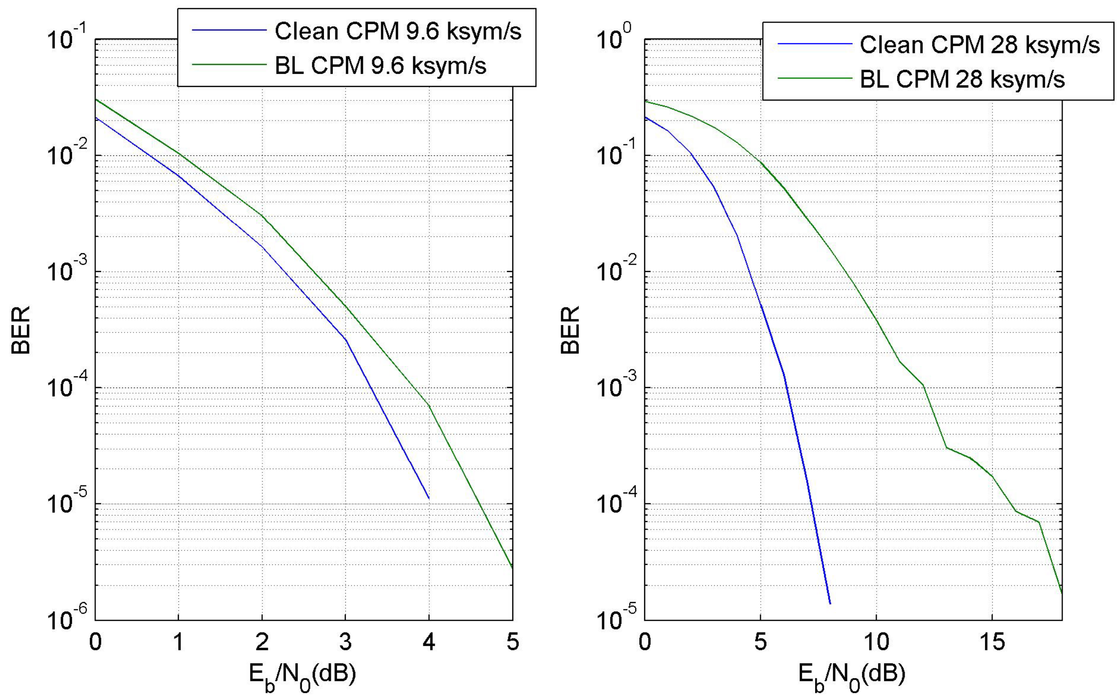

Figure 3. BER performance for different data rates.

At the low modulation rates, the signal is still decoded at an acceptable BER (less than 3dB from clean CPM performance), losing only half a dB relative to the clean signal. However, when the signal is transmitted at the faster rate, the BER loss is greater than 10 dB (for bit error rate of 10^{-4}), making the system unusable.

The goal of this thesis is to design, by different approaches, equalizers that will alleviate some of the loss in BER performance produced by the band limiting filter.

1.2 Literature Review

The research on equalization of CPM signals has been focused on finding methods to achieve MLSE performance without the need of a purely MLSE CPM Equalizer. The reason for this is that the complexity of a purely MLSE CPM equalizer increases exponentially with the amount of symbol memory used. Thus, for channels whose impulse response extends over several symbols, implementation of a MLSE CPM receiver is not practically feasible.

To reduce the complexity of a MLSE receiver, Forney [1] introduced the idea of using a linear equalizer as a pre-filter to shorten the impulse response of the channel. By shortening the impulse response of the channel, the memory requirements decrease and – consequently – the complexity of the MLSE decreases. Later, Falconer and Magee [2] introduced an adaptive channel truncation method for MLSE detection for PAM signals. Their structure uses a linear pre-filter to shape the overall response of the channel to have a – more desirable – impulse response of short duration. The desirable short duration impulse response is then computationally more efficiently equalized by the MLSE receiver.

In separate approaches, Lee and Hill [3], and Duel-Hallen and Heegard [4] implemented similar decision feedback structures to use in combination with the MLSE receiver. Both structures used a decision feedback equalizer as a pre-filter, prior to detection by the MLSE receiver. These structures also allow control of the complexity of the MLSE receiver by controlling the number of taps in the equalizer filter.

The advantage of the linear pre-filter approach is that it is simple to implement. In addition, this approach works well when the impulse response of the augmented channel is desirably short. On the negative side, a linear pre-filter may enhance the frequencies outside the band of interest, and thus enhance the noise.

The decision feedback approach is somewhat harder to implement and it is susceptible to error propagation. Nonetheless, even when degraded due to error propagation, decision feedback structures almost always outperform linear pre-filter equalizers.

The decision feedback approach has been applied to CPM signals by several authors. Cheung and Steele [5] proposed a feedback equalizer which allowed incorrect symbol decisions to be exchanged for correct symbol decisions when the feedback occurs, reducing the error propagation. Guren and Holte [6] implemented a decision feedback structure that reduces the complexity of the MLSE receiver by reducing the number of CPM states, with some tradeoff in performance.

In this work, we provide two approaches to the problem of equalization of CPM signals in band-limited channels. The first approach is to shorten the channel impulse response. We adapt the model of Falconer and Magee [2] for CPM signals and use it to equalize the ISI produced by the band limiting filter. This structure is chosen for two reasons. First, it provides the best possible pre-filter – based on the minimum mean square error criterion – to shape the channel impulse response. Second, the length of the desired impulse response can be changed to control the complexity of the MLSE receiver. In the second approach – for equalizing ISI in CPM signals – we analyze the source of the ISI, in terms of non-constant group-delay in the filter passband. We then design a pre-filter to act as a group-delay compensator, so that the overall response of the channel has a more constant group-delay.

2 Channel and Signal Characteristics

This chapter introduces the communications system that will be under study in this thesis. The system model consists of a transmitter, a channel, and a receiver and is shown in Figure 4.

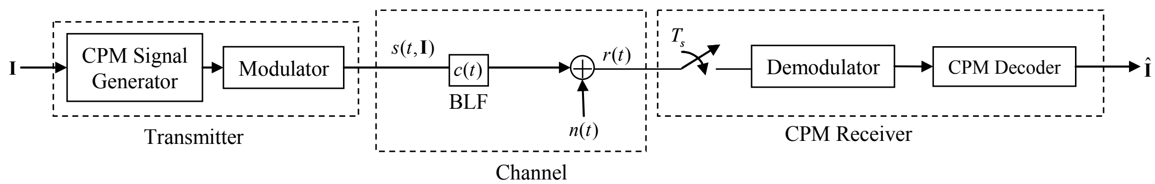

Figure 4. System block diagram.

The transmitter consists of a CPM signal generator that takes as input the symbol sequence **I**. From the symbol input sequence, a continuous phase modulated (CPM) signal is generated. The generation and properties of CPM signals are covered in Section 2.1. The output of the signal generator is modulated to around f_c Hz to create the transmitted signal $s(t,\mathbf{I})$.

The channel is modeled by a band limiting filter (BLF) and additive white Gaussian noise (AWGN). The BLF filter is characterized by its impulse response $c(t)$. The AWGN component is represented by $n(t)$. A more detailed description of the BLF is provided in Section 2.2. The output of the channel is represented as $r(t)$, where

$$r(t) = \int_{-\infty}^{\infty} s(\tau,\mathbf{I})c(t-\tau)d\tau + n(t) \tag{1}$$

At the receiver, $r(t)$ is sampled and demodulated before going through the CPM decoder, which produces the symbol sequence estimate $\hat{\mathbf{I}}$. At the receiver, the signal is sampled by f_s Hz. The different alternatives for receiver structures (CPM decoder) are the core of this thesis, and they will be discussed in Chapter 3.

2.1 Continuous Phase Modulation Overview

This section serves as an introduction to continuous phase modulated (CPM) signals. We first provide a general description of CPM signals followed by a description of a specific type of

CPM signal known as Multi-h CPM.

2.1.1 Signal Description

CPM is a class of modulation schemes for which the instantaneous phase is a continuous signal. CPM spectra do not exhibit the large spectral side lobes that result from discontinuities in the phase and thus CPM signals have better bandwidth efficiency characteristics than other modulation schemes such as QPSK, FSK, BPSK, and QAM. The instantaneous phase of CPM signals depends on the current symbol being transmitted as well as on previously transmitted symbols, so that we can say that CPM transmitters require memory [7]. A common way of expressing a CPM signal at complex baseband is

$$x(t) = e^{j\phi(t,\mathbf{I})} \tag{2}$$

where $\phi(t,\mathbf{I})$ is the instantaneous phase signal and $\mathbf{I} = \{I_k : k \in \mathbb{Z}\}$ represents the transmitted *M*-ary symbols. The symbol elements I_k are chosen from an *M*-ary alphabet (where *M* is a positive integer power of 2), in which symbols are defined for $\{\pm1, \pm3, \ldots, \pm(M-1)\}$. Each symbol of the alphabet can represent $\log_2 M$ bits. For instance, the quaternary ($M = 4$) CPM alphabet $\{+3,\ +1,\ \text{-}1,\ \text{-}3\}$ can be mapped to the set of bit-pairs $\{00,\ 01,\ 10,\ 11\}$.

The instantaneous phase signal $\phi(t;\mathbf{I})$ of the CPM signal is defined as

$$\phi(t;\mathbf{I}) = 2\pi \sum_{k=-\infty}^{\infty} I_k h q(t - kT) \tag{3}$$

with modulation index *h*, phase function *q*(*t*), and symbol period *T*. The modulation index *h* is a rational number $\frac{p}{q}$, where *p* and *q* have no common factors and $p < q$. Both *p* and *q* are integers to ensure a finite number of possible phase transitions. The modulation index determines the extent of phase rotation (as a fraction of π) per symbol unit. The phase function *q*(*t*) describes how the phase changes due to a symbol. The phase function is defined to be zero for all $t < 0$ and to have a value of ½ at time $t \geq LT$, where *T* is the symbol duration and *L* is the number of symbol periods it takes for the phase function to reach ½. The multiplier *L* is an integer quantity to guarantee that the frequency pulse shape spans an integer number of symbol periods. When $L = 1$, we refer to the CPM signal as being *full response* (the signal over one symbol period fully and only represents one symbol); when $L > 1$, we refer to the CPM signal as being *partial response* (the signal over one symbol period only partially represents one symbol, and multiple symbols influence the signal over one symbol period).

The frequency phase function is typically defined as the integral of a frequency pulse shape function $g(t)$

$$q(t) = \int_{-\infty}^{t} g(\tau)\, d\tau \tag{4}$$

The pulse shape function, along with the modulation index and the transmitted symbol, determine the instantaneous angular velocity during of the symbol transition. The frequency pulse shape function is non-zero only for $0 \le t \le LT$ and – to fulfill the final value property of $q(t)$ – it integrates to ½. Some common frequency pulse shape functions are rectangular (*L*REC), raised cosine (*L*RC), and Gaussian minimum-shift keying (GMSK) [7]. The letter *L* in the acronyms for the frequency pulse shape function is the same variable *L* that determines the pulse duration, i.e., a 2REC pulse means a rectangular pulse with $L = 2$. For example, an *L*REC pulse shape $g(t)$ is defined as

$$g(t) = \begin{cases} \dfrac{1}{2LT}, & 0 \le t < LT \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

An *L*REC frequency pulse shape $g(t)$ and its corresponding phase function $q(t)$ are shown in Figure 5.

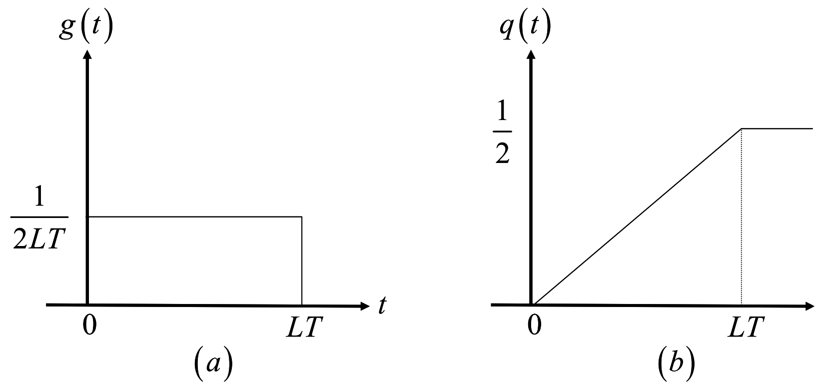

Figure 5. (a) Frequency pulse shape function and (b) corresponding phase function.

Figure 6 illustrates a full-response, 1REC quaternary CPM signal with modulation index $h = 1/4$. The real and imaginary components of the baseband signal are formed from piece-wise sinusoids. In other words, each symbol in the alphabet is represented by a sinusoid of a distinctive frequency. Hence, this 1REC single-*h* CPM is also referred to as continuous phase frequency shift keying (CPFSK) modulation. On a related note, minimum shift keying (MSK) is a special form of CPFSK for which h = ½ and M = 2.

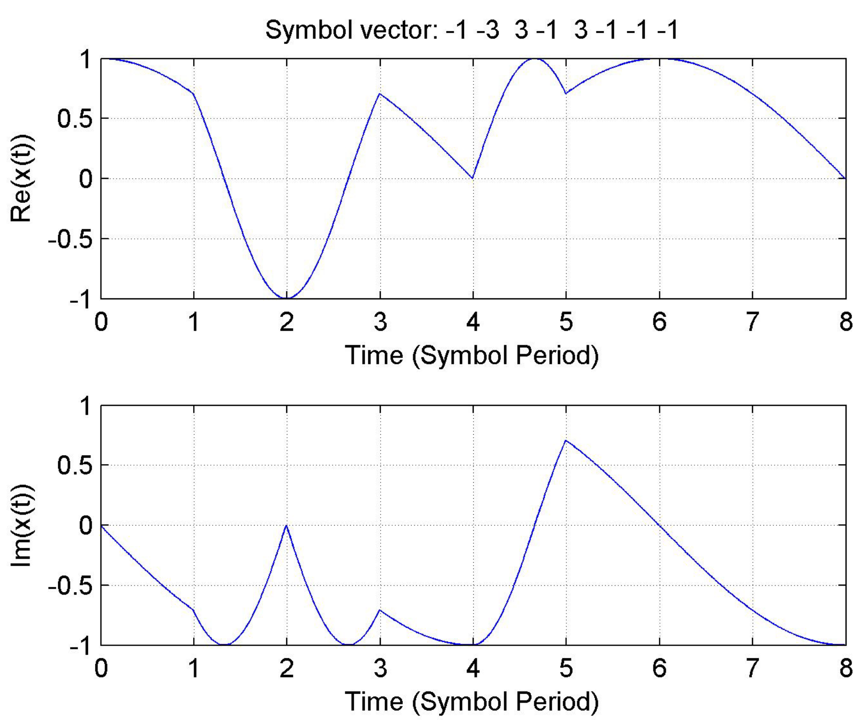

Figure 6. Baseband signal for quaternary CPM signal with 1REC frequency pulse shape, with modulation index *h*=1/4 and symbol sequence: [-1,-3,+3,-1,+3,-1,-1,-1].

The symbol transitions of a CPM signal can be represented by a state trellis diagram. The states of the trellis represent all attainable terminal phases, modulo 2π. The terminal phases are defined as the phase at $t = nT \;\forall\; n \in \mathbb{Z}$ (i.e., the phases at the symbol boundaries). A rational h (= p/q) guarantees that the phase trellis has a finite number of possible states. For full-response CPM signals, if p is odd, the trellis has a total of $2q$ possible phase states; if p is even, it has q possible phase states. Having a finite number of states is an important characteristic of CPM as it allows the use of the Viterbi algorithm in the receiver.

To illustrate a CPM state trellis diagram, Figure 7 shows the phase trellis (in black) and the path (in red) taken by the CPM baseband signal shown in Figure 6. With the modulation index $h = \frac{1}{4}$ and a quaternary alphabet {+1, –1, +3, –3}, the phase changes by $\frac{\pi}{4}$, $-\frac{\pi}{4}$, $\frac{3\pi}{4}$, or $-\frac{3\pi}{4}$ radians, respectively, during a symbol period. The symbols I_k are indicated above each phase transition interval.

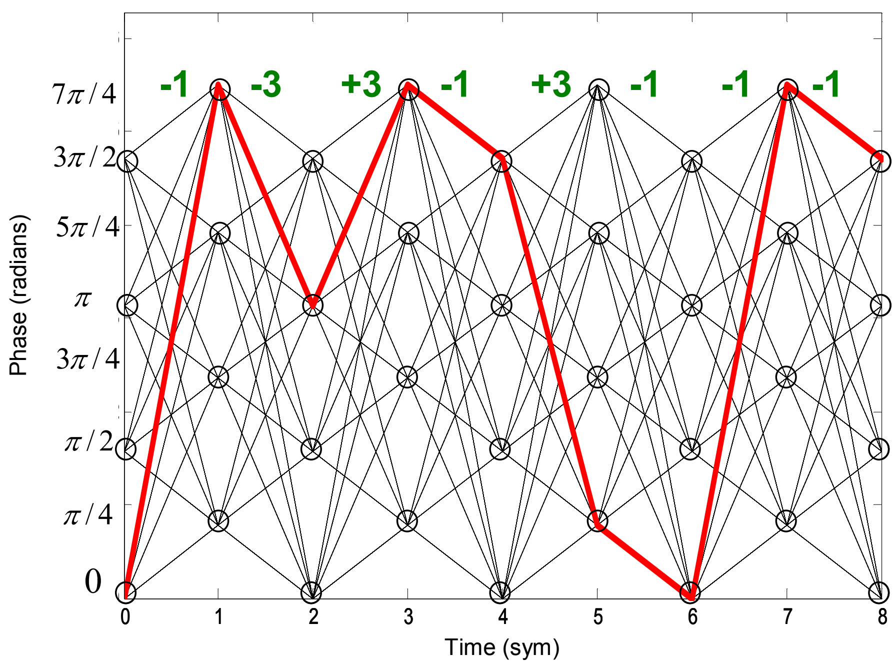

Figure 7. Trellis path (red) for the CPM signal shown in Figure 6 with all possible phase paths (black).

The state trellis in Figure 7 exhibits a total of 8 states (recall that an odd p results in $2q$ states):

$$\Theta = \left\{0, \quad \frac{\pi}{4}, \quad \frac{\pi}{2}, \quad \frac{3\pi}{4}, \quad \pi, \quad \frac{5\pi}{4}, \quad \frac{3\pi}{2}, \quad \frac{7\pi}{4}\right\} \tag{6}$$

However, for a given symbol index, there are only 4 possible states. More specifically, there are two unique sets of states for even and odd indices:

$$\Theta_{even} = \left\{0, \quad \frac{\pi}{2}, \quad \pi, \quad \frac{3\pi}{2}\right\} \tag{7}$$

and

$$\Theta_{odd} = \left\{\frac{\pi}{4}, \quad \frac{3\pi}{4}, \quad \frac{5\pi}{4}, \quad \frac{7\pi}{4}\right\} \tag{8}$$

This characteristic of CPM signals can be exploited to reduce the number of metric computations in the Viterbi decoder, by only computing the metrics for a set of q states. The remaining q states are not considered valid phase transitions and therefore they are ignored [8].

The error rate performance for the optimum CPM receiver for an AWGN channel, is dominated by the minimum Euclidean distance $\delta^2_{\min}$ and can be expressed as [7]

$$P_e = K_{\delta_{\min}} Q\left(\sqrt{\frac{\varepsilon_b}{N_b}\delta^2_{\min}}\right) \tag{9}$$

where $K_{\delta_{\min}}$ is the number of paths having $\delta^2_{\min}$, and $\frac{\varepsilon_b}{N_0}$ is the ratio between the energy per bit and the spectral noise density. The minimum Euclidean distance $\delta^2_{\min}$ is the normalized minimum squared Euclidean distance between any two paths through the trellis which split apart at time $t = 0$ and merge later. The normalization of $\delta^2_{\min}$ is with respect to the energy per bit ε_b. The minimum Euclidean distance is related to the *constraint length* [8], which is the least number of symbol intervals over which any two paths will remain unmerged. The longer the constraint length, the larger the minimum Euclidean distance between two paths, which results in better error rate performance. In Figure 8, we show two merging paths in the phase trellis for a full-response CPM signal with h = ½. The constraint length is two symbol periods and is indicated in the plot. Indeed, the constraint length for single-*h* CPM signals is always two symbol periods because CPM symbols come in signed pairs, i.e. $\pm 1, \pm 3,$ etc. As will be shown later, the constraint length can be increased by using multiple modulation indices.

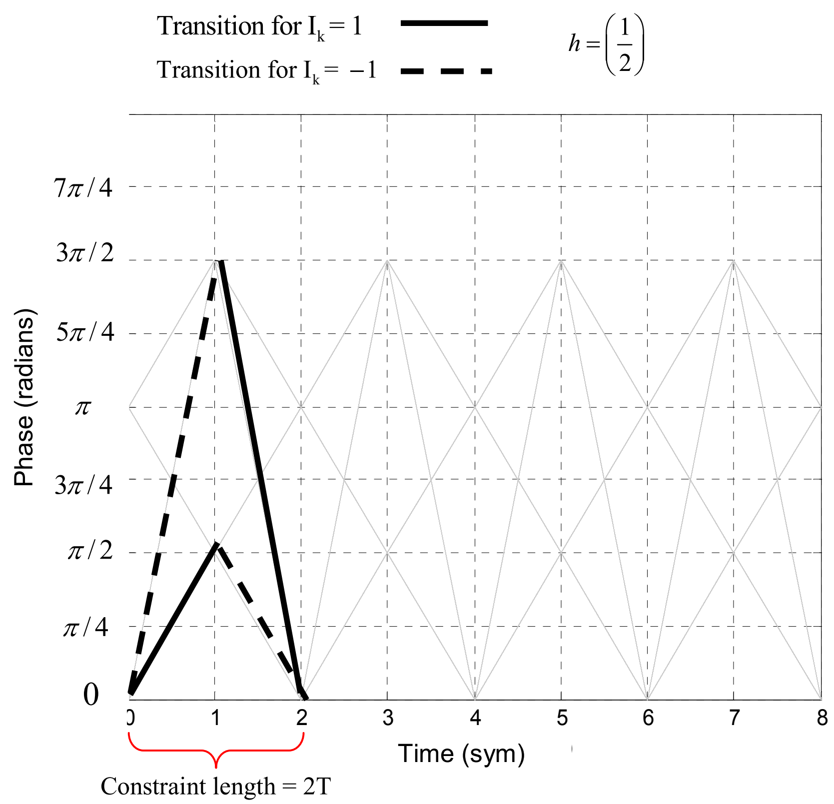

Figure 8. Phase trellis for a full-response binary CPM, h=1/2.

2.1.2 Multi-h CPM

CPM signals that use more than one modulation index are called multi-*h* CPM (MHCPM) signals. In general, MHCPM signals have a fixed number K of modulation indices that are varied periodically in successive symbol duration intervals. The CPM phase signal definition in (3) can be generalized to include the MHCPM signals as follows:

$$\phi(t;\mathbf{I}) = 2\pi \sum_{k=-\infty}^{\infty} I_k h_{k \bmod K} q(t-kT) \qquad (10)$$

where mod represents the modulus operation. The modulation indices generally form a set of rational numbers with common denominator q :

$$\mathbf{h} = \{h_0, h_2, \ldots, h_{K-1}\} = \left\{\frac{p_0}{q}, \frac{p_1}{q}, \cdots, \frac{p_{K-1}}{q}\right\} \tag{11}$$

This method of selecting modulation indices ensures a periodic phase trellis, which is a trellis condition necessary for employing the VA [8].

The basic idea of MHCPM is to delay the first merge for the purpose of increasing the minimum Euclidean distance between paths in the phase trellis. The first merge will inevitably occur at $t \le (L+K)T$. Hence, we denote $(L+K)$ as the maximum attainable constraint length, provided $q \ge M^K$. In the example for full-response single-*h* CPM shown in Figure 8, the observed constraint length of 2 is the maximum attainable constraint length. Two merging paths of the phase trellis for a full-response (*L*=1) binary MHCPM signal with $\mathbf{h} = \left(\frac{2}{4}, \ \frac{3}{4}\right)$ (*K*=2) are shown in Figure 9.

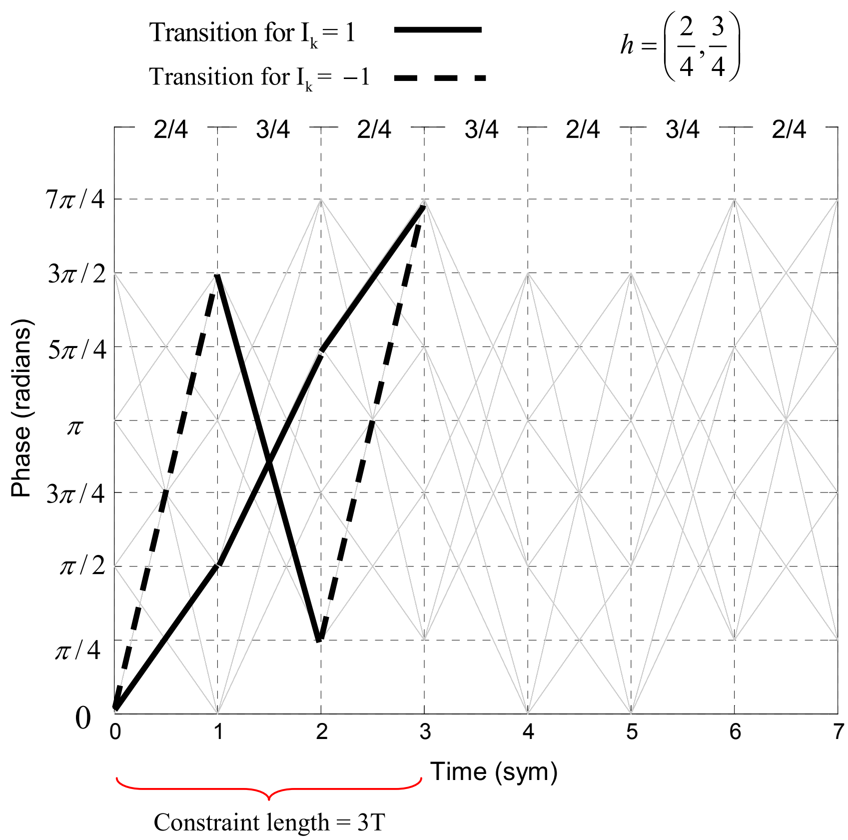

Figure 9. Phase trellis full-response MHCPM signal, h = [2/4, 3/4].

The constraint length of $3T$ as shown in the figure agrees with the maximum attainable constraint length of 3. Clearly, K – the number of modulation indices – has a direct effect on the constraint length, although certain conditions must be met to obtain the maximum attainable constraint length. For M-ary full-response MHCPM signals, apparently the necessary and sufficient condition on **h** – to obtain the maximum constraint length – is that $I_0 h_0 + I_1 h_1 + \cdots + I_{K-1} h_{K-1}$ must not be integer-valued for any possible symbol sequence $\{I_j : j = \{0,1,\ldots,(K-1)\}\}$. To illustrate the need for this condition consider a binary full-response CPM signal with modulation indices $\mathbf{h} = (1/4, 3/4)$ whose trellis is shown in Figure 10. Even though $K = 2$, because of poor index selection, the constraint length for this signal is only 2, which is short of the maximum attainable constraint length of 3.

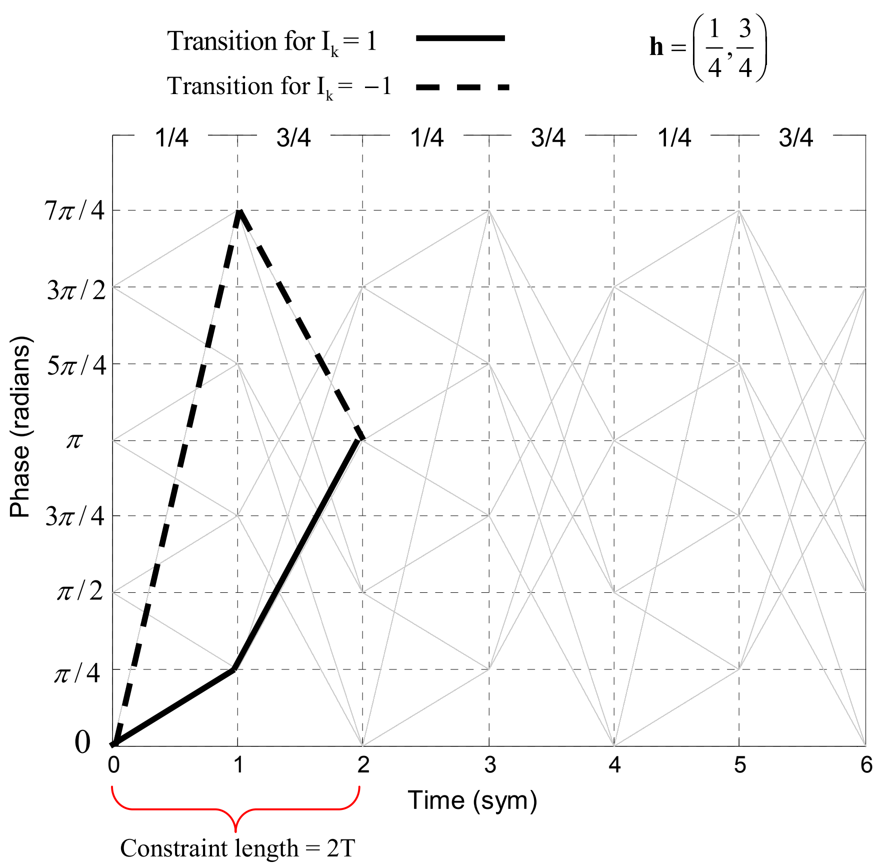

Figure 10. Phase trellis for binary full-response CPM signal, h= [1/4 3/4].

2.2 Channel Model

In this thesis, the channel is characterized as a band-pass filter having an impulse response $c(t)$ and frequency response characteristic $C(f)$. The channel is referred to as band limited because it limits the transmitted signal to W Hz, where $W < f_s$. In other words, $C(f) = 0$ for all frequencies outside the passband frequency range $f_c - 0.5W < f < f_c + 0.5W$, where f_c is the signal carrier frequency.

Within the bandwidth of the channel, the frequency response is expressed as

$$C(f) = |C(f)| e^{j\theta(f)} \tag{12}$$

where $|C(f)|$ is the magnitude response characteristic and $\theta(f)$ is the phase response characteristic. Additionally, the group-delay characteristic is defined as

$$\tau(f) = -\frac{1}{2\pi}\frac{\partial\theta(f)}{\partial f} \tag{13}$$

A channel is non-distorting if $|C(f)|$ is constant and if $\theta(f)$ is a linear function of frequency over the passband frequency range. If $\theta(f)$ is a linear function then $\tau(f)$ is constant throughout the passband. On the contrary, if the amplitude response $|C(f)|$ is not constant in the passband, the channel distorts the signal in amplitude. In addition, if the group-delay $\tau(f)$ is non constant over the passband, the channel distorts the signal in delay.

From the work of Peterson et al.[9], in the context of satellite communications, a good model to approximate an analog 25 kHz bandpass filter is a 12-pole Chebyshev filter. In this thesis we adopt this model and use a 25 kHz bandpass as a benchmark for all experiments. The selectivity or magnitude response, along with the phase and group-delay responses for the bandpass filter model are shown in Figure 11.

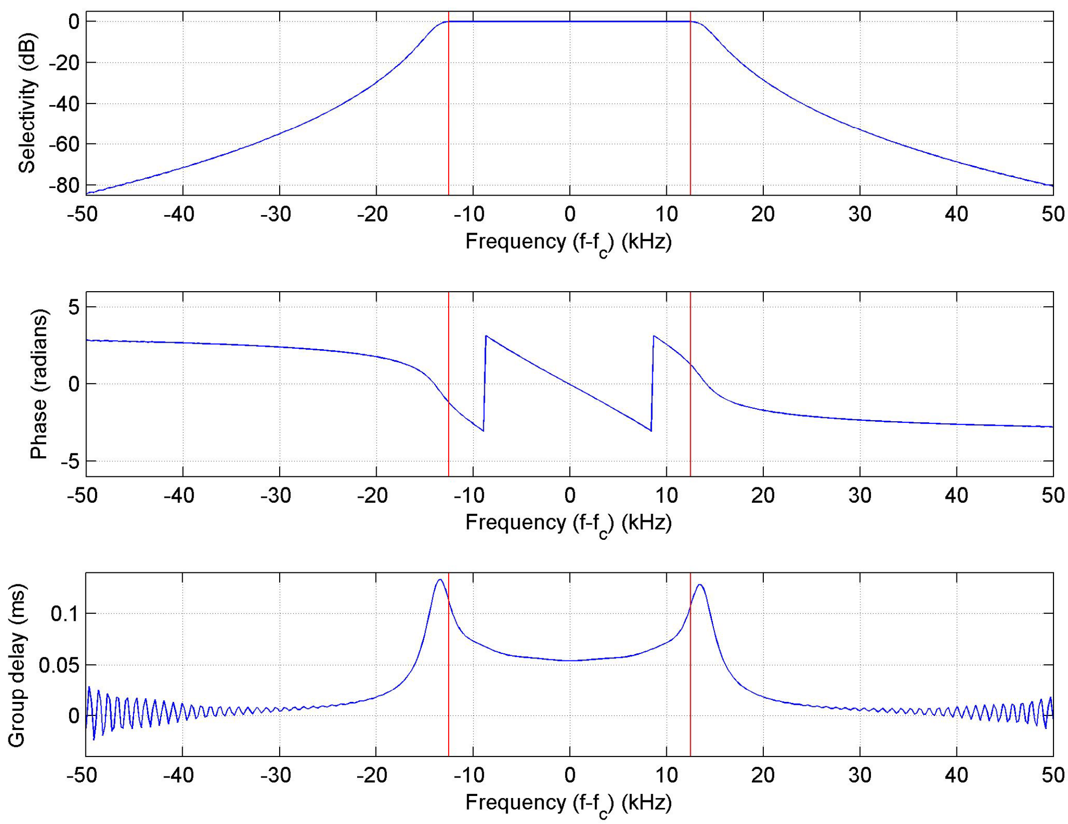

Figure 11. Chevyshev bandpass filter model.

The vertical lines delimit the ±12.5 kHz cutoff frequencies. Note that, although the filter has constant magnitude in the passband, the phase response exhibits non-linear behavior, especially at the passband edges. The phase non-linearity becomes more evident in the group-delay response which reveals a higher delay for frequencies close to the passband edges. As a result of delay distortion, pulses with varying frequency content get delayed more than others and as a consequence pulses overlap introducing intersymbol interference (ISI).

From a time domain perspective, the impulse response of the channel is shown in Figure 12.

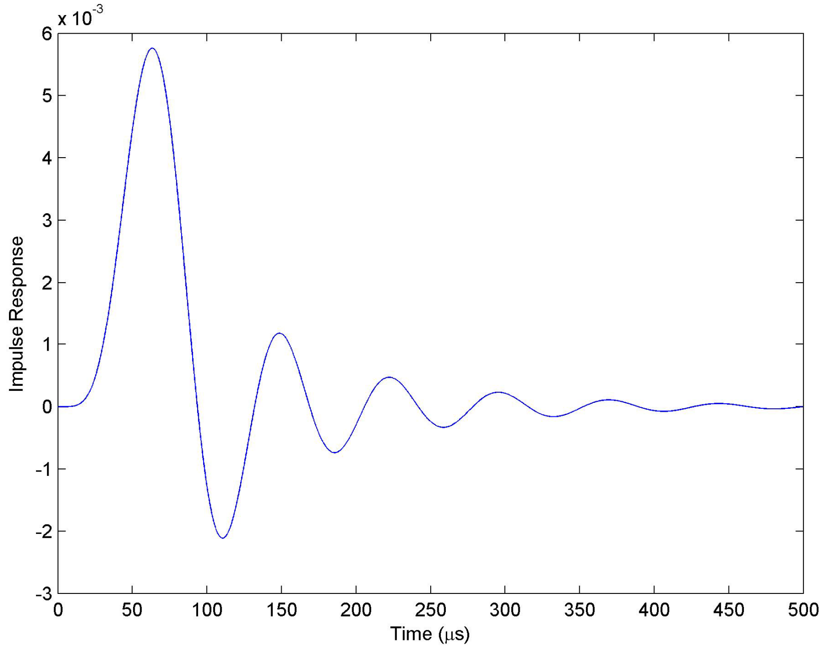

Figure 12. Channel Impulse Response.

In the context of equalization for CPM signals, it is useful to characterize the duration of the channel impulse response in terms of the integer number of symbols over which the channel impulse response extends. If we let the channel impulse response $c(t)$ be a band-limited (free of aliasing) continuous complex function that essentially vanishes after some time T_{ch}, then – when the channel is approximated in the digital domain – by sampling with period T_s, we obtain

$$c_k = c(kT_s) \tag{14}$$

We let N_{ch} be the number of samples of c_k before its impulse response becomes essentially zero, and we then find the number of symbols over which the channel impulse response extends as

$$L_{ch} = \left\lceil \frac{N_{ch} T_s}{T} + 0.5 \right\rceil \tag{15}$$

where $\lceil \; \rceil$ represents rounding up. When the channel is a single pulse or delta function with complex gain, so that there is no ISI, we find $L_{ch} = 1$. The latter is also found when the extent of the channel impulse response is non-zero but less than half of a symbol period; this case corresponds to non-resolvable ISI. From (15), the faster the modulation rate, the more symbols the channel spreads over. Table 1 shows the number of symbols over which the channel impulse response is spread. The results shown in Table 1 ignore the effect of the channel impulse response after 400μs.

Table 1. Extent of ISI by BPF model for different modulation rates.

Modulation Rate (sym/s)	L_{ch}
16,000	7
24,000	11
28,000	12
32,000	14

We can see from Table 1 that the severity of the ISI increases with the modulation rate, which makes equalization a challenge.

3 Receiver Structures

This chapter presents several receiver structures designed to equalize band limited CPM signals, and is divided in two major subsections. The first subsection introduces the optimal (in terms of performance) CPM equalizer whereas the second subsection provides receiver alternatives of reduced complexity but sub-optimal performance.

The optimal way of receiving CPM signals in a band-limited channel is to use the maximum-likelihood sequence-estimator (MLSE). This receiver is referred to as the MLSE CPM equalizer and it is implemented using the Viterbi Algorithm (VA). The MLSE is implemented using the Viterbi algorithm (VA) because the CPM signal can be described as a finite state machine (in this report, the terms MLSE and VA are used interchangeably). To facilitate the description of the VA, a MLSE CPM receiver for an AWGN channel is introduced first. Thus, details of the functions related to the VA – such as metric computation and template signal generation – are discussed first with respect to the simpler MLSE CPM receiver for the AWGN channel and then later adapted for the MLSE CPM equalizer.

In the second major subsection, two suboptimal receiver structures based on the MLSE CPM equalizer are presented. The first of the sub-optimal receivers implements the *channel truncation* method, which is designed to shorten the length of the impulse response of the channel by means of a pre-filter and thus reduces some of the complexity of the receiver. The second suboptimal receiver consists of a group-delay compensator (GDC) designed to eliminate the inter-symbol interference (ISI) that results from the variation in the group-delay response of a channel.

3.1 MLSE CPM Equalizer

In this section we present the optimal MLSE CPM receiver for the AWGN channel followed by the MLSE CPM equalizer for band limited channels. For simplicity, the concepts of maximum likelihood as well as processes related to the VA will first be described in the context of an MLSE CPM receiver in an AWGN channel and later expanded and applied in the more complex MLSE CPM equalizer. For clarity in analysis, the models used in this section are assumed to be operating at (complex) baseband.

3.1.1 MLSE Receiver

In this section, we introduce the optimum CPM receiver for the AWGN channel. The optimum CPM receiver consists of an MLSE block preceded by a metric computation block connected to a template signal generator. Figure 13 shows the structure of the optimum CPM receiver.

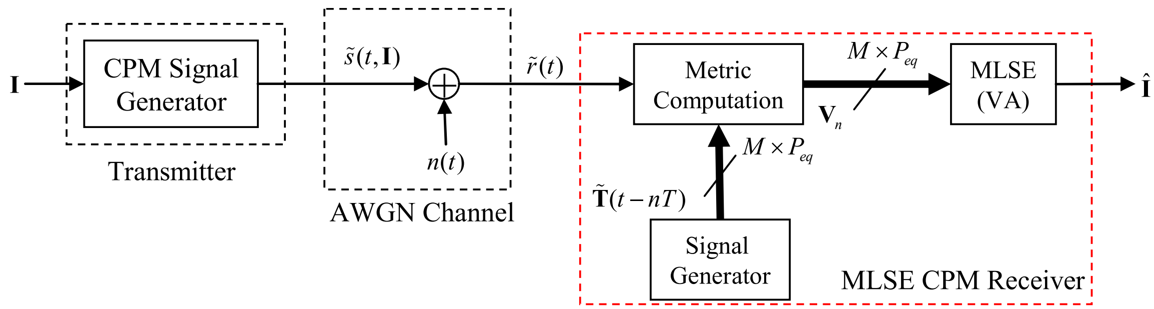

Figure 13. Optimum CPM receiver.

The received signal is given by

$$\tilde{r}(t) = \tilde{s}(t,\mathbf{I}) + n(t) \tag{16}$$

where $\tilde{s}(t,\mathbf{I})$ is the M-ary (M is the size of the alphabet) CPM signal associated with the transmitted symbol sequence I_n — denoted as a vector $\mathbf{I}$ — and $n(t)$ is AWGN. The metric computation block computes the correlation between the template signal generator and the received signal, and outputs the matrix of metrics**:**

$$\mathbf{V}_n = \begin{bmatrix} v_n(\mathcal{I}_1,\Theta_1) & v_n(\mathcal{I}_1,\Theta_2) & \cdots & v_n(\mathcal{I}_1,\Theta_P) \\ v_n(\mathcal{I}_2,\Theta_1) & v_n(\mathcal{I}_2,\Theta_2) & & v_n(\mathcal{I}_2,\Theta_P) \\ \vdots & & \ddots & \vdots \\ v_n(\mathcal{I}_M,\Theta_1) & v_n(\mathcal{I}_M,\Theta_2) & \cdots & v_n(\mathcal{I}_M,\Theta_P) \end{bmatrix} \tag{17}$$

where $v_n(\mathcal{I}_m,\Theta_p)$ represents the branch metric for the n-th received symbol to be the symbol $\mathcal{I}_m$ (one of the M symbols in the alphabet) arriving at the trellis state with the associated signal phase Θ_p. In a CPM receiver, the number of trellis states P corresponds to the discrete number of phase values that $\tilde{s}(t,\mathbf{I})$ can take on at each symbol interval $t = nT$, where $n = 0,1,\ldots,N$.

The branch metric $v_n\left(\mathcal{I}_m, \Theta_p\right)$ is then computed by correlating $r\left(t-nT\right)$, over the interval $nT \leq t \leq \left(n+1\right)T$, and the template waveform $\tilde{t}_{\mathcal{I}_m,\Theta_p}\left(t\right)$. The symbol template signal $\tilde{t}_{\mathcal{I}_m,\Theta_p}\left(t\right)$—which is non-zero only over the symbol interval $0 \leq t \leq T$—provides the expected waveform for the received symbol to be $\mathcal{I}_m$ and for the underlying finite state trellis to be at the *p*-th state at the end of the symbol period. In analogy to the branch metrics, these symbol template signals are compactly expressed as a matrix of signals:

$$\tilde{\mathbf{T}}\left(t\right)=\begin{bmatrix} \tilde{t}_{\mathcal{I}_1,\Theta_1}\left(t\right) & \tilde{t}_{\mathcal{I}_1,\Theta_2}\left(t\right) & \cdots & \tilde{t}_{\mathcal{I}_1,\Theta_P}\left(t\right) \\ \tilde{t}_{\mathcal{I}_2,\Theta_1}\left(t\right) & \tilde{t}_{\mathcal{I}_2,\Theta_2}\left(t\right) & & \tilde{t}_{\mathcal{I}_2,\Theta_P}\left(t\right) \\ \vdots & & \ddots & \vdots \\ \tilde{t}_{\mathcal{I}_M,\Theta_1}\left(t\right) & \tilde{t}_{\mathcal{I}_M,\Theta_2}\left(t\right) & \cdots & \tilde{t}_{\mathcal{I}_M,\Theta_P}\left(t\right) \end{bmatrix} \tag{18}$$

A more complete discussion of the symbol template signals $\tilde{t}_{\mathcal{I}_m,\Theta_p}\left(t\right)$ and the branch metrics $v_n\left(\mathcal{I}_m, \Theta_p\right)$ is provided in Section 3.1.1.2. The optimum CPM receiver produces a sequence of symbol estimates $\hat{I}_n$ (or $\hat{\mathbf{I}}$) which correspond to its best estimate for the transmitted sequence I_n. In addition to the branch metric discussion, Section 3.1.1.1 covers the operation of the MLSE block.

3.1.1.1 Finite-State MLSE (Viterbi Algorithm)

The MLSE (VA) block in Figure 13 performs MLSE using the Viterbi algorithm. The use of the Viterbi algorithm is a plausible solution to detect the symbols in the CPM signal because the behavior of CPM signals can be described with a finite-state trellis. This section begins with the general definition of the MLSE and shows that the Viterbi algorithm performs MLSE in detecting a finite-duration CPM signal.

The maximum-likelihood (ML) criterion produces decisions [7] based on the conditional discrete PDF $p\left(\tilde{s}(t,\mathbf{I}) \,|\, \tilde{r}(t)\right)$ of *a posteriori* probabilities. The a posteriori probabilities are defined as $\Pr\left(\tilde{s}(t)=\hat{s}_k(t,\tilde{\mathbf{I}}_k) | \tilde{r}(t)\right)$, or the probability that $\hat{s}_k\left(t,\tilde{\mathbf{I}}_k\right)$—one of the M_S possible transmitted signals based on the *k*-th symbol sequence $\tilde{\mathbf{I}}_k \triangleq \left[\tilde{I}_{k,0} \quad \tilde{I}_{k,1} \quad \ldots \quad \tilde{I}_{k,N-1}\right]$— is transmitted given the received signal $\tilde{r}\left(t\right)$. The ML criterion evaluates the conditional PDFs of a posteriori probabilities (or any monotonic function thereof) and finds an estimate $\hat{s}(t,\mathbf{I})$ based on the maximum of the conditional PDFs, such that

$$\hat{s}\left(t,\hat{\mathbf{I}}\right)=\underset{\left\{\widehat{s}_k(t),\ k=1:M_S\right\}}{\arg\max}\ p\left(\widehat{s}_k(t,\tilde{\mathbf{I}}_k)\,|\,\tilde{r}(t)\right) \tag{19}$$

The number of sequence template signals M_S increases exponentially with the duration of the signal. Assuming that all template signals $\widehat{s}_k\left(t,\tilde{\mathbf{I}}_k\right)$ are equi-probable, the conditional probability $\Pr\left(\widehat{s}_k(t,\tilde{\mathbf{I}}_k)|\tilde{r}(t)\right)$ is proportional to the cross-correlation metric [7, 10]

$$Z\left(\tilde{\mathbf{I}}_k\right)=\int_{-\infty}^{\infty}\mathrm{Re}\left[\tilde{r}(t)\,\widehat{s}\left(t,\tilde{\mathbf{I}}_k\right)^*\right]dt \tag{20}$$

All possible sequence template signals $\breve{s}\left(t,\tilde{\mathbf{I}}_k\right)$ are correlated with the received signal $r(t)$, and the received symbol sequence $\hat{I}_n$ (or $\hat{\mathbf{I}}$) is determined to be that data sequence $\tilde{\mathbf{I}}_k$ that maximizes (20). In other words,

$$\hat{\mathbf{I}}=\underset{\left\{\tilde{\mathbf{I}}_k,\ k=1:M_S\right\}}{\arg\max}\ Z\left(\tilde{\mathbf{I}}_k\right) \tag{21}$$

Because the generation of a CPM signal is a causal process (i.e., the signal during $nT\leq t\leq(n+1)T$ only depends on up to the *n*-th symbol), we define the cross-correlation metric up to the *n*-th symbol as

$$Z_n\left(\tilde{\mathbf{I}}_k\right)=\int_{-\infty}^{(n+1)T}\mathrm{Re}\left[\tilde{r}(t)\,\widehat{s}\left(t,\tilde{\mathbf{I}}_k\right)^*\right]dt \tag{22}$$

Note that $Z_N\left(\tilde{\mathbf{I}}_k\right)=Z\left(\tilde{\mathbf{I}}_k\right)$. The metric $Z_n\left(\tilde{\mathbf{I}}_k\right)$ requires knowledge of up to the *n*-th symbol in $\tilde{\mathbf{I}}_k$ only. We use the notation $\tilde{\mathbf{I}}_k(n)$ to indicate the subset $\left[\tilde{I}_{k,0}\quad \tilde{I}_{k,1}\quad \ldots\quad \tilde{I}_{k,n}\right]$ of $\tilde{\mathbf{I}}_k$. Then, (22) can be written as

$$Z_n\left(\tilde{\mathbf{I}}_k(n)\right)=\int_{-\infty}^{(n+1)T}\mathrm{Re}\left[\tilde{r}(t)\,\widehat{s}\left(t,\tilde{\mathbf{I}}_k(n)\right)^*\right]dt \tag{23}$$

Equation (23) can then be computed in recursive fashion by

$$Z_n\left(\tilde{\mathbf{I}}_k(n)\right) = Z_{n-1}\left(\tilde{\mathbf{I}}_k(n-1)\right) + v_n\left(\tilde{\mathbf{I}}_k(n\ )\right) \tag{24}$$

where

$$v_n\left(\tilde{\mathbf{I}}_k(n)\right) = \int_{nT}^{(n+1)T} \mathrm{Re}\left[\tilde{r}(t)\,\hat{s}\left(t, \tilde{\mathbf{I}}_k(n)\right)^*\right] dt \tag{25}$$

updates the cross-correlation metric when the *n*-th symbol is included.

As we previously showed in Section 2.1, a finite state trellis can be used to represent a CPM signal with a finite alphabet and rational modulation index(es). The Viterbi algorithm efficiently performs the MLSE for the finite-duration CPM signal (or can be an excellent approximate realization of the MLSE for a perpetual CPM signal). For full-response CPM signals, the states in the trellis correspond to the possible terminal phases. The metric of the *p*-th trellis state at time *n* can then be defined as

$$Z_{p,n}\left(\hat{\mathbf{I}}_p(n-1)\right) = \int_{-\infty}^{nT} \mathrm{Re}\left[\tilde{r}(t)\,\hat{s}\left(t, \hat{\mathbf{I}}_p(n-1)\right)^*\right] dt \tag{26}$$

where $\hat{\mathbf{I}}_p(n-1)$ is the best length-$(n-1)$ sequence that leads up to the *p*-th state. Now, the number of possible state transitions during each symbol period is given by *PM*. The potential state metric at time *n*, given that the *n*-th symbol is $\mathcal{I}_m$, can be computed recursively from

$$Z_{p,n+1}\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1), \mathcal{I}_m\right) = Z_{q(\mathcal{I}_m,p),n}\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1)\right) + v_n\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1), \mathcal{I}_m\right) \tag{27}$$

where $q(\mathcal{I}_m, p)$ is the trellis state from which symbol $\mathcal{I}_m$ will get to the *p*-th trellis state, and $v_n\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1), \mathcal{I}_m\right)$ is the corresponding branch metric for that state transition. Consequently, the best estimate for the *n*-th transmitted symbol of a sequence arriving at state *p* at time *n*+1 is given by

$$\hat{I}_{p,n} = \underset{\mathcal{I}_m \in A}{\arg\max}\left\{Z_{p,n+1}\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1), \mathcal{I}_m\right)\right\} \tag{28}$$

where *A* is the *M*-ary CPM alphabet. As a result, the best symbol sequence estimate $\hat{\mathbf{I}}_p(n)$, up to the n-th symbol ending at the *p*-th state, can be found by extending one of the previous estimates with $\hat{I}_{p,n}$,

$$\hat{\mathbf{I}}_p(n) = \left[\hat{\mathbf{I}}_{q(\hat{I}_{p,n},p)}(n-1) \;\vdots\; \hat{I}_{p,n}\right] \quad (29)$$

From this follows the evaluation of $Z_{p,n+1}\left(\hat{\mathbf{I}}_p(n)\right)$, the p-th state metric at time n+1. Alternatively we can obtain the state metric while computing (26) or

$$Z_{p,n+1}\left(\hat{\mathbf{I}}_p(n)\right) = \max_{\mathcal{I}_m \in A}\left\{Z_{p,n+1}\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1), \mathcal{I}_m\right)\right\} \quad (30)$$

For a finite-duration CPM signal of N symbols, the sequence estimate $\hat{\mathbf{I}} \triangleq \left[\hat{I}_0 \quad \hat{I}_1 \quad \dots \quad \hat{I}_{N-1}\right]$ is selected as that sequence which arrives at the p-th state with the highest state metric

$$\hat{\mathbf{I}} = \hat{\mathbf{I}}_{p_{\max}}\left(N-1\right) \quad (31)$$

where

$$p_{\max} = \arg\max_{p \in 1:M}\left\{Z_{p,N}\left(\hat{\mathbf{I}}_p(N-1)\right)\right\} \quad (32)$$

For a perpetual CPM signal (or very long signal, i.e. one too long for the receiver to wait until its end), the notion of trace-back is introduced to the Viterbi algorithm. With the trace-back length N_{tb} (in symbols) the Viterbi algorithm estimates the (n–N_{tb})-th symbol at time n by selecting the (n–N_{tb})-th symbol of $\hat{\mathbf{I}}_{p_{\max}}\left(n\right)$ as its estimate. This allows the receiver to produce $\hat{I}_n$ at a constant rate (although the output is subject to the delay of N_{tb} symbols) and reduces the memory requirements for the receiver, as only the sequence information over the newest N_{tb} symbols is required.

3.1.1.2 Branch Metric Computation for the Optimum CPM Receiver

When MLSE for CPM detection is implemented using the Viterbi algorithm (VA), equation (30) can be interpreted as the updating of the state metric for state p, with $Z_{q(\mathcal{I}_m,p),n}\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1)\right)$ representing the state metrics for a state q up to time $(n-1)$ and $v_n\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1), \mathcal{I}_m\right)$ representing the branch metric for a transition from state q to state p when symbol $\mathcal{I}_m$ was transmitted at time *n*. The symbol $\mathcal{I}_m$ is one of the symbols in the *M*-ary CPM alphabet.

Computing the branch metric $v_n\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1),\mathcal{I}_m\right)$ for full-response CPM does not actually require explicit knowledge of the previous symbols (i.e., $\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1)$). Instead, the departing phase state $\Theta_{q(\mathcal{I}_m,p)}$ (which does not depend on time *n*) is the only information from the past necessary to compute the metric. Moreover, as $q\left(\mathcal{I}_m,p\right)$ suggests, the departing state depends on the arriving state (or vice versa, depending on the perspective). We choose to make the branch metric a function of the arriving phase state Θ_p because it facilitates the updating of the state metrics at the VA end. Hence, we use the following notation [7]

$$v_n\left(\mathcal{I}_m,\Theta_p\right)=v_n\left(\hat{\mathbf{I}}_{q(\mathcal{I}_m,p)}(n-1),\mathcal{I}_m\right) \tag{33}$$

The branch metric in (11) can be equivalently defined as

$$v_n\left(\mathcal{I}_m,\Theta_p\right)=\int_{nT}^{(n+1)T}\mathrm{Re}\left[r\left(t\right)\tilde{t}_{\mathcal{I}_m,\Theta_p}\left(t-nT\right)^*\right]dt \tag{34}$$

where $\tilde{t}_{\mathcal{I}_m,\Theta_p}\left(t\right)$ is a symbol template signal, for the optimum CPM receiver, that arrives at the *p*-th state Θ_p when symbol $\mathcal{I}_m$ is transmitted. Thus, the symbol interval metric $v_n\left(\mathcal{I}_m,\Theta_p\right)$ is the real part of the complex correlation (equivalently the sum of correlations of the in-phase and quadrature components) between the received signal $r\left(t\right)$ and the aligned symbol template signal $\tilde{t}_{\left(I_m,\Theta_p\right)}\left(t-nT\right)$ over the interval $nT\le t\le\left(n+1\right)T$. During each symbol interval, there are *PM* possible state transitions; therefore, we need *PM* symbol template signals $\tilde{t}_{\mathcal{I}_m,\Theta_p}\left(t\right)$. For example, Figure 14 shows the periodic state trellis for MSK—for which *P* = 4 and *M* = 2—and the symbol template signals for all the transitions.

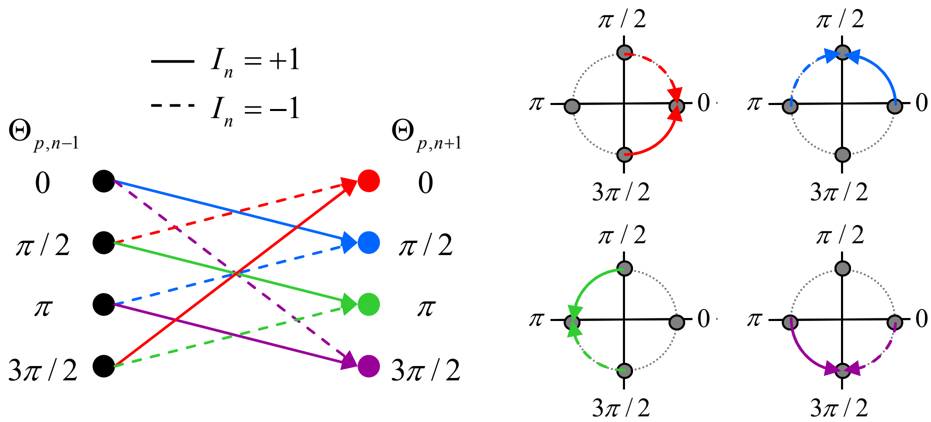

Figure 14. Trellis state transitions for MSK (left) and its 8 symbol template signals on the complex plane (right).

The trellis transitions are color-coded according to the arriving state. There are 2 state transitions arriving at each state. All signals in one color arrive at the same state, e.g., the state transitions in red both arrive at state $\Theta_1 = 0$. This organization facilitates the evaluation of (16).

The correlation operation in (34) is performed for all symbol template signals in $\tilde{\mathbf{T}}(t)$ and thereby generates $\mathbf{V}_n$. The structure of the metric computation block is shown in Figure 15.

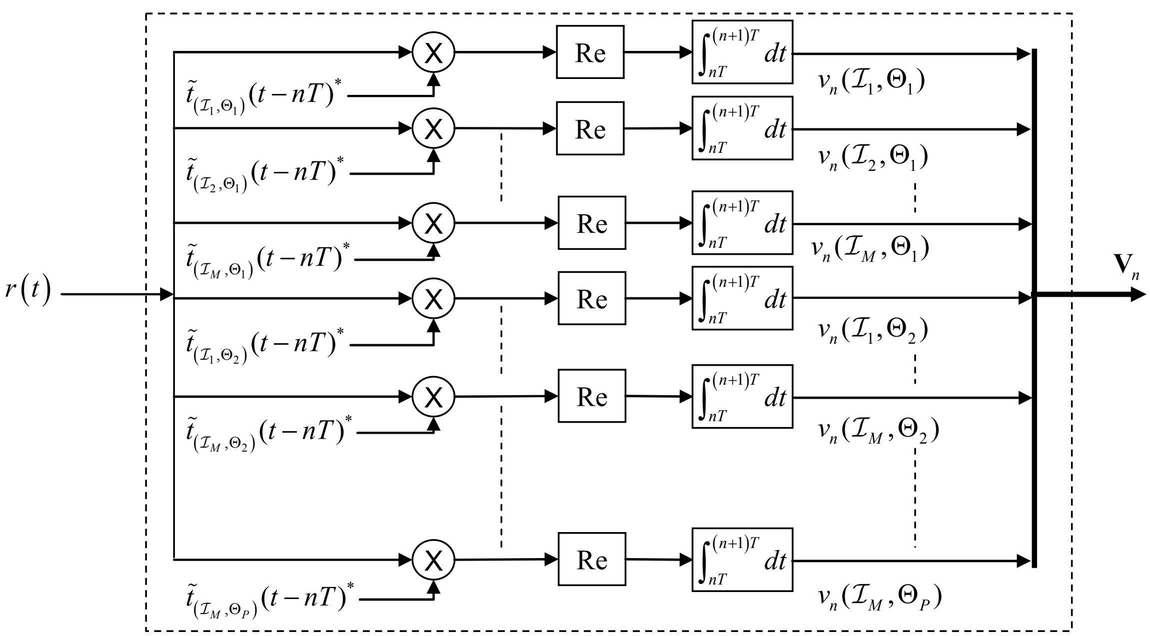

Figure 15. Metric computation block.

The metric computation block operates on the received signal $r(t)$ on a symbol interval basis, i.e. integration over $nT < t \le (n+1)T$. The received signal is correlated with the conjugate of all signals in $\tilde{\mathbf{T}}(t)$, which are stacked by column, to form a bank of correlators. Since the correlators are matched to the complex conjugate of the channel, the correlator output is maximized when it is entirely real. Thus, the real component of the complex product is extracted ("Re" block) and then passed through integrators. Finally, all the branch metrics are multiplexed to produce as output $\mathbf{V}_n$, the metric matrix.

3.1.2 MLSE Equalizer

This section builds on the optimum MLSE CPM receiver for AWGN channels presented in Section 3.1.1. Using knowledge of the channel, the MLSE CPM receiver is modified to receive and equalize the received signal. The structure for the MLSE CPM Equalizer is shown in Figure 16. The single addition to the MLSE CPM receiver structure consists of a channel estimator block, the output of which is used to reduce the effects of ISI introduced by a frequency selective channel.

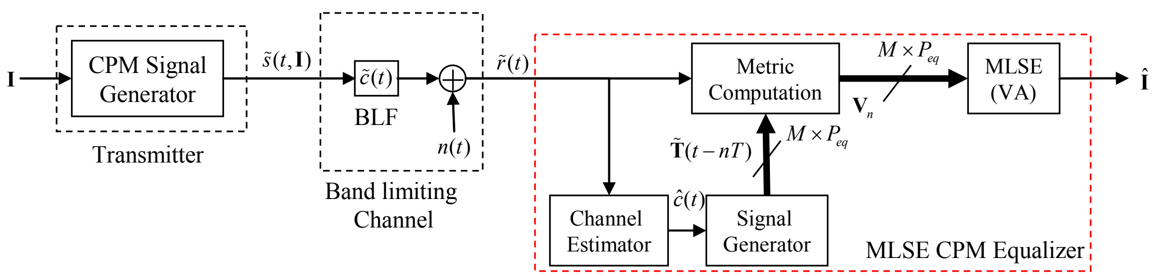

Figure 16. MLSE equalizer structure.

Although transparent to the structure, the MLSE CPM receiver presented in Section 3.1.1 and the MLSE equalizer presented here differ in the method by which the template signals are generated and in the complexity of the MLSE trellis. These two differences are examined in subsequent subsections.

The received signal is now given by $\tilde{r}(t)$ and is defined as

$$\tilde{r}(t) = \int_{-\infty}^{\infty} \tilde{s}(t,\mathbf{I})\tilde{c}(t-\tau)d\tau + n(t) \tag{35}$$

where $\tilde{s}(t,\mathbf{I})$ is the M-ary CPM signal associated with the symbol sequence **I,** $\tilde{c}(t)$ is the baseband impulse response for the band limiting channel, and n(t) is AWGN. $\tilde{\mathbf{T}}(t)$ is the matrix of template signals generated using the channel estimate $\hat{c}(t)$, and $\mathbf{V}_n$ is the matrix of branch metrics that correspond to the n-th symbol. The dimensions for $\tilde{\mathbf{T}}(t)$ and $\mathbf{V}_n$ are specified as M x P_{eq} where M is the size of the CPM alphabet and P_{eq} is the total number of states in the equalizer trellis. The operation of the *Metric Computation* and *MLSE* blocks is the same as for the MLSE CPM receiver discussed in Section 3.1.1. The *Channel Estimator* block refers to the process of estimating the channel impulse response. A condition for true maximum likelihood is that the channel estimate $\hat{c}(t)$ be an accurate model of the actual channel $\tilde{c}(t)$, otherwise the performance is sub-optimal. The method by which the impulse response of the channel is estimated lies outside the scope of this thesis and will not be considered here. For now, the channel estimate $\hat{c}(t)$ will be considered known a priori.

The two key differences between the equalizer that is presented here and the MLSE CPM receiver are the configuration of the MLSE trellis and the template signal generator. The MLSE equalizer trellis is more complex because each state contains memory of previously transmitted symbols. The template signal generator for the MLSE equalizer differs from the MLSE CPM receiver in that it now accounts for the channel effects (ISI). The template signals use the channel estimate $\hat{c}(t)$ to account for the effect of ISI. More detailed explanations of the MLSE equalizer trellis and the generation of the template signals follow.

3.1.2.1 MLSE equalizer periodic state trellis

This section investigates the trellis construction of the MLSE equalizer. The MLSE equalizer trellis differs from the MLSE receiver trellis in that it incorporates memory of previously transmitted symbols. Assuming a channel that produces ISI spanning over L symbols, the MLSE criterion is equivalent to estimating the state of a discrete-time finite state machine [11]. The finite-state machine is the discrete-time channel and the state is composed of the L most recent inputs. Thus, the trellis states of the MLSE equalizer are a combination of a terminal phase value called the *phase state* and a vector of $L-1$ memory symbols, referred to as the *correlative state*. In the context of MLSE equalizers, $L-1$ is associated with the depth of the correlative state and/or the number of symbols taken into account to compensate for ISI. Thus, the p-th trellis state is denoted as

$$\Theta_p = \{ \underbrace{\tilde{\theta}_p}_{\textit{Phase state}}, \underbrace{\tilde{I}_{-1,p}, \tilde{I}_{-2,p}, \ldots, \tilde{I}_{-L+1,p}}_{\textit{Correlative State}} \} \tag{36}$$

At time k, the phase state represents the present phase value $\tilde{\theta}_p$ and the correlative state represents the $L-1$ most recent symbols, i.e. $\{\tilde{I}_{k-1}, \tilde{I}_{k-2}, \ldots, \tilde{I}_{k-L+1}\}$, which – due to the signal spreading caused by the channel – produce ISI on I_k.

The number of trellis states in a CPM MLSE equalizer is given by $P_{eq} = PM^{L-1}$, where P is the number of terminal phase values for the M-ary CPM signal. In addition, the number of possible state transitions at any given symbol interval is given by PM^L. Note that L affects the complexity (number of states) of the trellis exponentially. As illustration, a periodic state trellis for a dual-h binary CPM MLSE equalizer with h = [1/4 2/4] and $L = 3$ is shown in Figure 17.

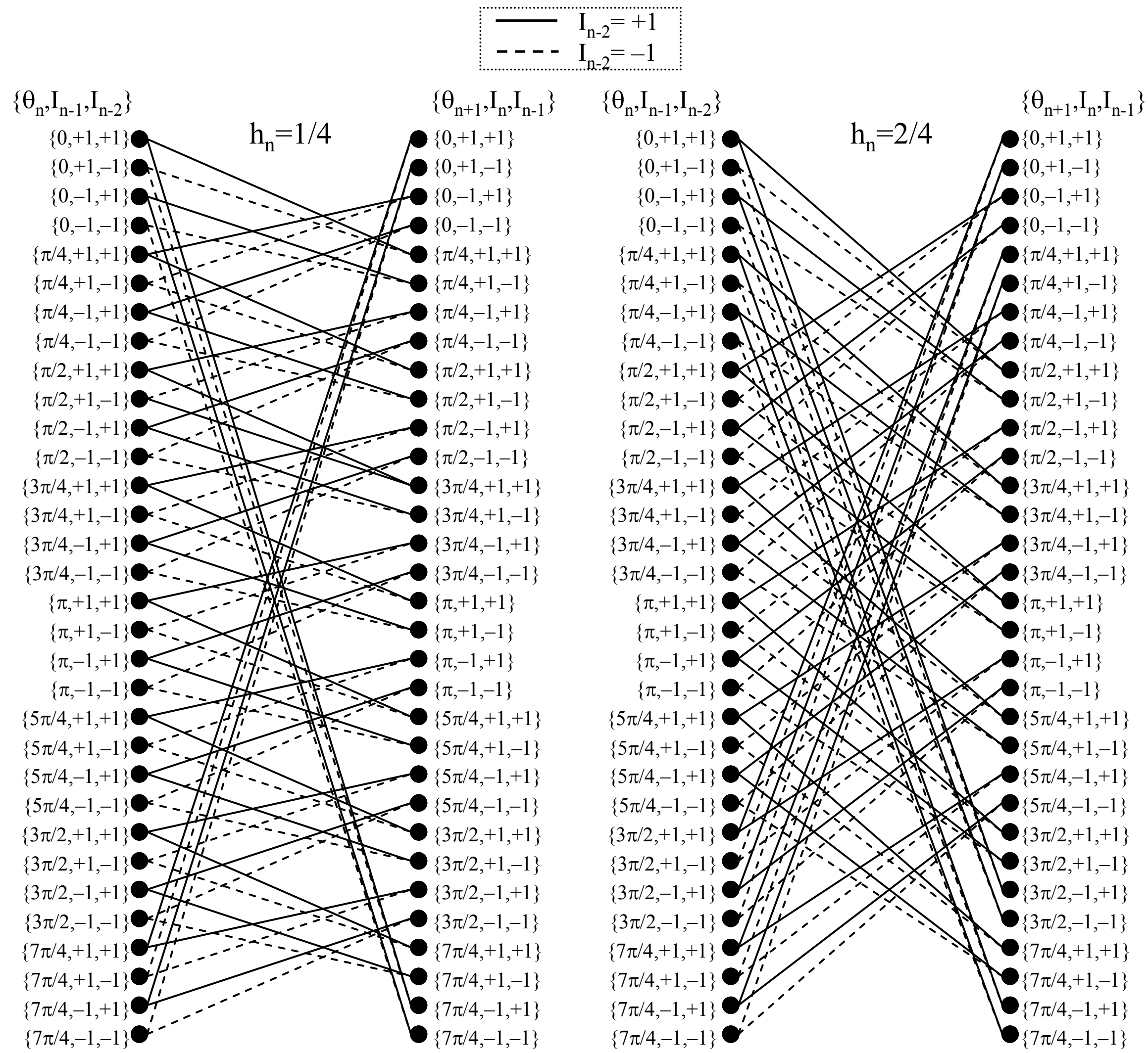

Figure 17. Periodic state trellis for dual-*h* binary CPM MLSE Equalizer; h = [1/4 2/4], *L*=3.

The left trellis corresponds to modulation index h =1/4 and the right trellis corresponds to modulation index h = 2/4. For multi-h CPM signals, we require as many state trellises as the number of modulation indices. The MLSE receiver will alternate successively between trellises for each symbol interval. For the given modulation indexes, we have, P = 8. The total number of states in the trellis is given by $8 \cdot 2^{3-1} = 32$ and the number of state transitions by $8 \cdot 2^3 = 64$. Note that the decision made for each state transition is determined on the basis of the L-th previously transmitted symbol. For instance, in the left trellis (h =1/4), one can get to state #5 $\{\pi/4, +1, +1\}$ by departing from either state #1 $\{0, +1, +1\}$ or state #2 $\{0, +1, -1\}$. Both departing states have the same I_{n-1} symbol, but they differ in symbol I_{n-2}. In general, two departing states that arrive at the same state will have the same correlative state except for the last element, I_{n-L+1}, which is the criterion used to select one state transition over the other. The state transitions shown with a solid line correspond to $I_{n-2} = +1$, whereas the transitions shown with a dotted line correspond to $I_{n-2} = -1$.

3.1.2.2 Template signal generation

The template signal generation for the MLSE equalizer is different from that for the MLSE CPM receiver presented in Section 3.1.1.2. While the template signals for the MLSE receiver are only associated with a phase transition, the template signals for the MLSE equalizer are associated with a phase transition, a symbol history, and a channel estimate. This section describes the procedure by which the template signals for the MLSE equalizer are generated.

The generation of the template signals consists of two steps. The first step involves the generation of a bank $\mathbf{G}(t)$ of PM^L symbol template signals that correspond to each one of the state transitions of the CPM MLSE equalizer trellis. These transition signals must include the correlative state information, and therefore each signal has duration *LT*. In the second step, all symbol template signals, in $\mathbf{G}(t)$, are convolved with the channel estimate $\hat{h}(t)$ to obtain the template signals $\tilde{\mathbf{T}}(t)$.

For ease of notation, an alternative state notation $\Theta_{k(\mathcal{I}_m,p)}$ is introduced in addition to (4). The trellis state $\Theta_{k(\mathcal{I}_m,p)}$ is defined as the trellis state to depart from in order to arrive at the *p*-th state. The correlative state element $\tilde{I}_{-L+1,k}$ for $\Theta_{k(\mathcal{I}_m,p)}$ is defined by $\mathcal{I}_m$. The symbol template signal associated with the state transition $\Theta_{k(\mathcal{I}_m,p)} \rightarrow \Theta_p$ at time *n* is denoted as $g(\tau;\mathcal{I}_m,\Theta_p,n)$, where

$$g(t;\mathcal{I}_m,\Theta_p,n) = \exp\left[j\left\{ \tilde{\theta}_p - 2\pi\left(\mathcal{I}_m h_{n-L} q(t-LT) + \sum_{v=-L+1}^{-1} \tilde{I}_{v,p} h_{n-v} q(t-vT) \right)\right\}\right] \quad (37)$$

To illustrate the process of (37), consider the trellis shown in Figure 17 for the case *h* = ¼. The generation of the symbol template signal for the transition from the 2nd state to the 5th state is illustrated in Figure 18.

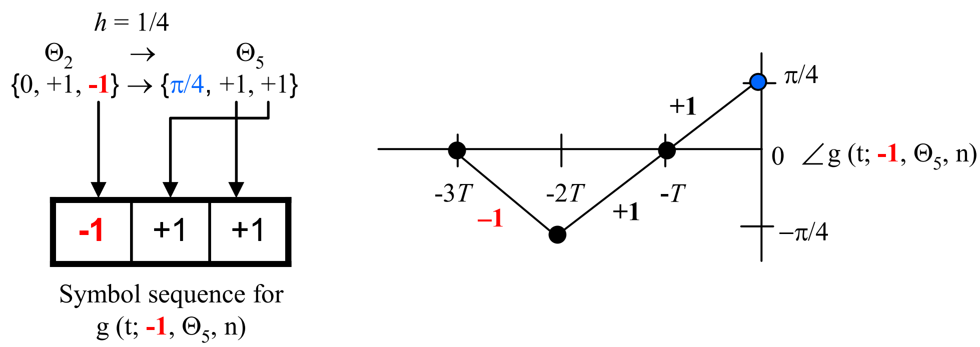

Figure 18. Example of symbol template signal.

The CPM symbol shown in red corresponds to $\mathcal{I}_m$ in (37) and it is the first element of the symbol sequence of L elements that will have final phase value $\tilde{\theta}_p$ (π/4 in the example). Note also that the symbol template signal is defined only from -LT (-3T for the example) to 0 and that the final phase value corresponds to the phase state of the arriving state.

The template signals $\tilde{t}_{m,p}(t;\Theta_p,\mathcal{I}_m,n)$ used for the computation of the branch metrics are defined as

$$\tilde{t}_{m,p}(t;\Theta_p,\mathcal{I}_m,n)=\begin{cases}\int_{-\infty}^{\infty} g(\tau;\mathcal{I}_m,\Theta_p,n)\hat{h}(t-\tau)d\tau & -T\le t<0\\ 0 & \text{otherwise}\end{cases} \tag{38}$$

where $\hat{h}(t)$ is the channel estimate of duration $(L-1)T$. The convolution is defined strictly over $-T\le t<0$ since it is in this interval that $\tilde{t}_{m,p}(t;\Theta_p,\mathcal{I}_m,n)$ contains information of all L symbols in $g(t;\mathcal{I}_m,\Theta_p,n)$. The template signals are normalized to have unit energy over the symbol interval, i.e.

$$\int_{-T}^{0}\left|\tilde{t}_{m,p}(t;\Theta_p,\mathcal{I}_m,n)\right|^2=1 \tag{39}$$

To show the interdependence of the previous transmitted symbols in the template signal, we show in Figure 19 diagrams that indicate the influence that each of the L previously transmitted symbols has on $\tilde{t}_{m,p}(t;\Theta_p,\mathcal{I}_m,n)$. In Figure 19a we represent an abstract symbol template signal for $L=3$, in which the influence of the L previous symbols is indicated with different colors. In Figure 19b we represent the channel estimate which is defined from 0 to $(L-1)T$. We let the channel be white, representing a neutral color.

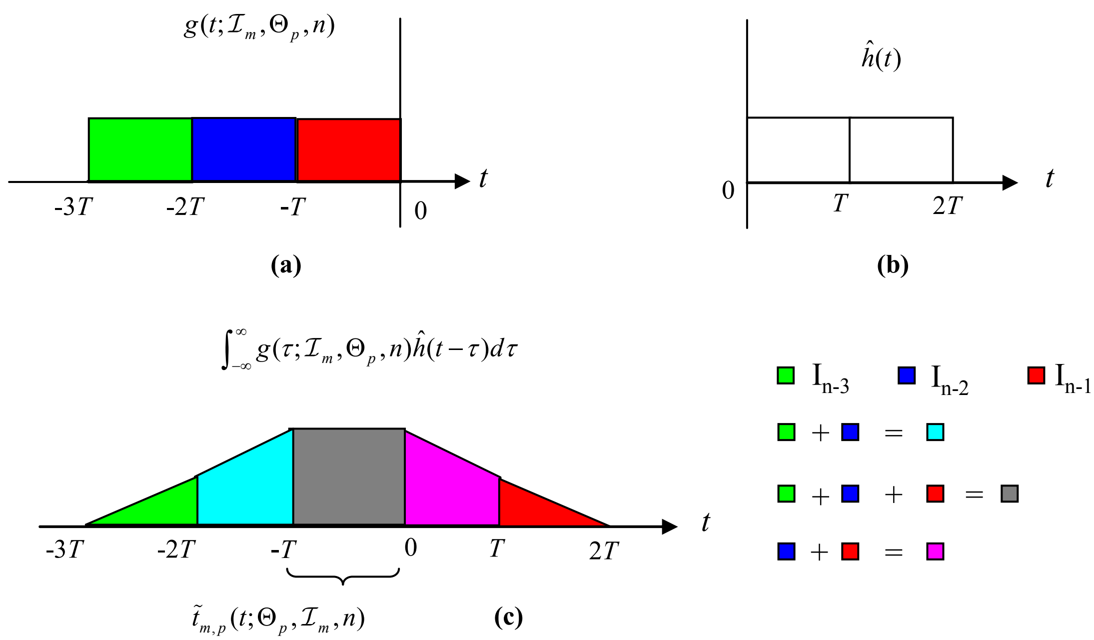

Figure 19. Symbol influence in template signal generation.

In Figure 19c we show the convolution between the symbol template signal and the channel estimate. The influence that each previously transmitted symbol has on the convolution result is indicated through the color combinations indicated in the figure. The template signal $\tilde{t}_{m,p}(t;\Theta_p,\mathcal{I}_m,n)$ is chosen to be the convolution signal in the interval $-T \le t < 0$, since this interval contains information on the L previously transmitted symbols.

The collection of all template signals is contained in the matrix of signals $\tilde{\mathbf{T}}(t)$, such that

$$\tilde{\mathbf{T}}(t)=\begin{bmatrix} \tilde{t}_{m,p}(t;\Theta_1,\mathcal{I}_1,n) & \tilde{t}_{m,p}(t;\Theta_2,\mathcal{I}_1,n) & \cdots & \tilde{t}_{m,p}(t;\Theta_{P_{eq}},\mathcal{I}_1,n) \\ \tilde{t}_{m,p}(t;\Theta_1,\mathcal{I}_2,n) & \tilde{t}_{m,p}(t;\Theta_2,\mathcal{I}_2,n) & & \tilde{t}_{m,p}(t;\Theta_{P_{eq}},\mathcal{I}_2,n) \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{t}_{m,p}(t;\Theta_1,\mathcal{I}_M,n) & \tilde{t}_{m,p}(t;\Theta_2,\mathcal{I}_M,n) & \cdots & \tilde{t}_{m,p}(t;\Theta_{P_{eq}},\mathcal{I}_M,n) \end{bmatrix} \tag{40}$$

The template signal matrix $\tilde{\mathbf{T}}(t)$ is sent to the Branch Metric computation block where each one of the template signals $\tilde{t}_{m,p}(t;\Theta_p,\mathcal{I}_m,n)$ gets convolved with the received signal $r(t)$ thus generating the branch metrics. A more thorough description of the computation of the branch metrics was presented in Section 3.1.1.2.

3.1.2.3 Simulation performance of MLSE Equalizer under ISI AWGN channel

This sub-section presents the performance of the MLSE equalizer MATLAB implementation equivalent to those reported by Peterson et al. [9]. The performance of the MLSE equalizer is tested using three different channel estimates obtained by assigning the length of the equalizer memory L to be 1 (no equalization case), 3 and 5, and thus varying the complexity by 32, 512, and 8192 states respectively.

The first case, for which $L=1$, corresponds to the case where the receiver makes decisions based on measurements of the present symbol only. Hence, there is no symbol memory and therefore no equalization takes place. In the second case, $L=2$, the channel impulse response estimate is considered only over $0 \le t \le 2T$. Similarly, for the case when $L=5$, the channel impulse response estimate is assumed to be non-zero over $0 \le t \le 4T$. Figure 20 shows the channel impulse response estimates that are used by the MLSE equalizer for each value of L. The actual channel is shown in the background of each plot to give an indication of the unresolved ISI for each case.

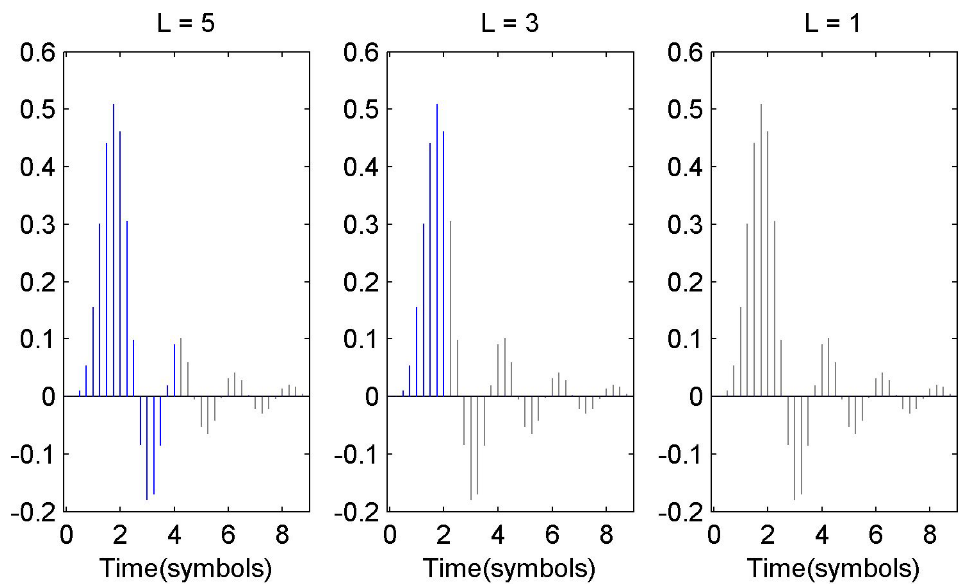

Figure 20. Channel estimates.

All cases shown in Figure 20 correspond to partial channel estimation, because only a portion of the actual channel is considered by the equalizer. The performance of the MLSE equalizer with partial channel estimation is sub-optimal. The bigger the symbol memory L, the closer the performance to MLSE will be since more ISI can be counteracted. Unfortunately, increasing L

increases the complexity of the MLSE equalizer exponentially.

Finally, we show the simulation results for each of the symbol memory lengths for MLSE equalizers. The simulation consisted of 3 trials of 10,000 symbols of the transmitted signal. The MLSE Viterbi trace-back length was set to 31, while the number of samples per symbol was set to 4. The bit-error rate P_b was obtained from the simulated symbol error rate P_e as follows

$$P_b = \frac{2^{\log_2 M - 1}}{2^{\log_2 M} - 1} P_e \tag{41}$$

The results are shown in Figure 21.

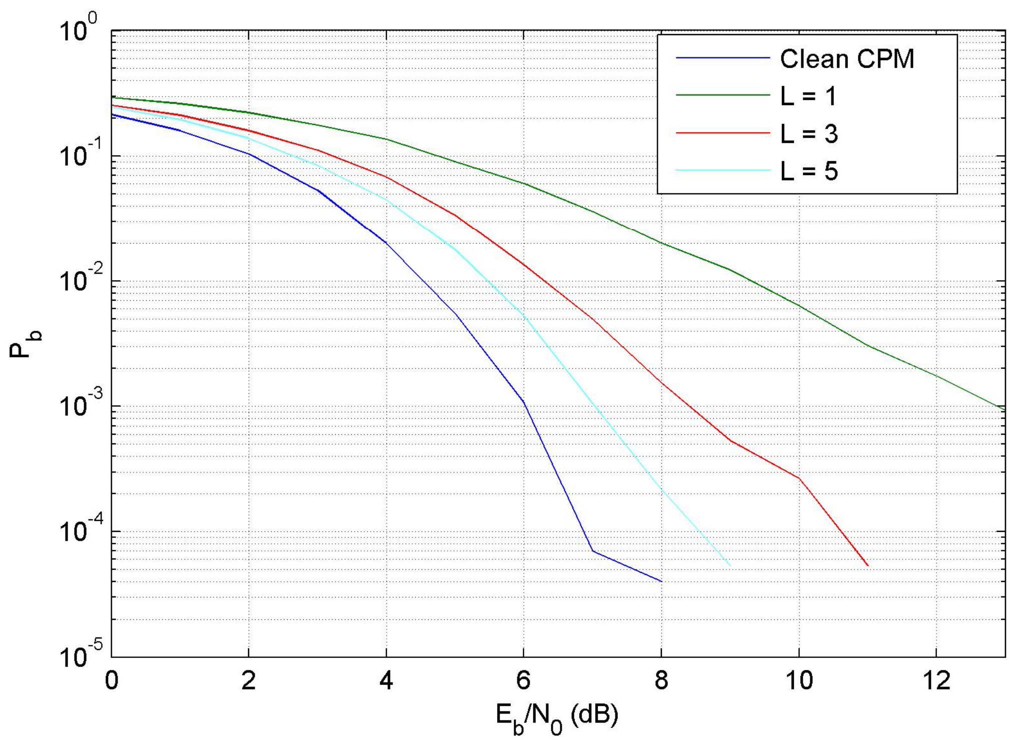

Figure 21. Simulation results for CPM MLSE equalizer with different symbol memory length *L*.

As expected, the MLSE equalizer that compensated the most ISI was the equalizer using $L = 5$ (cyan), outperforming the equalizer using $L = 3$ (red). In this case, the performance gain between $L = 5$ and $L = 3$ was roughly 1 dB for low E_b/N_0 and seemed to increase for larger E_b/N_0. In general, the amount of performance loss due to partial channel estimation depends strictly on the characteristics of the channel and/or on how much of the ISI is left unresolved. The case with no equalization (green) performed poorly compared to the other two cases, and corresponds to the detection of the band limited CPM signal without equalization. These results are consistent with the results obtained by Peterson et al.[9, Figure 7].

3.2 Channel Memory Truncation for MLSE of CPM signals

In this section we present the adaptive channel memory truncation MLSE model of Falconer and Magee [2] to be used for CPM signals which are sampled faster than the symbol rate T (i.e. there are multiple samples per symbol). In the Falconer and Magee model for performing equalization, a short duration desired impulse response (DIR) filter q_n is used by the Viterbi Algorithm as a model for the channel estimate. The DIR filter is designed to approximate the combined response of the BLF $c(t)$ and a pre-filter p_n, and to have a shorter impulse response than $c(t)$. Hence, the objective is to shorten the overall response of the channel (as seen by the receiver) so that the signal can be fully equalized with a less complex MLSE equalizer. Thus, the DIR filter q_n and the pre-filter p_n, with frequency responses $Q(f)$ and $P(f)$ respectively, are designed such that

$$\frac{Q(f)}{P(f)} \approx C(f) \tag{42}$$

Even though in this thesis we are concerned only with steady state behavior, it is insightful to provide an adaptive structure in order to find the optimum p_n and q_n. The communication model for adaptive channel memory truncation of Falconer and Magee adapted for CPM signals is shown in Figure 22.

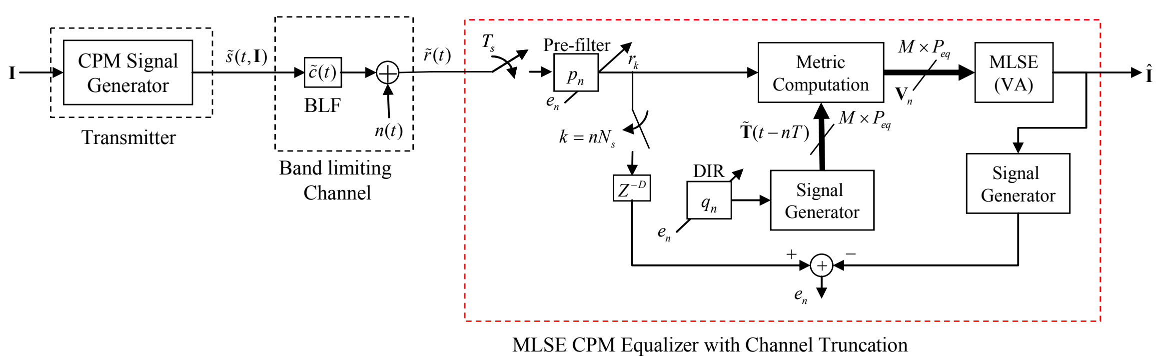

Figure 22. Adaptive channel memory truncation model.

The received signal $\tilde{r}(t)$, and output of the channel, is sampled at the receiver at rate T_s. Each sample of the received signal, is given by

$$\tilde{r}_k = \sum_{m=0}^{N_c} c_m^* \tilde{s}_{k-m} + n_k \tag{43}$$

where c_m is the m-th tap of the sampled version of the BLF impulse response, $\tilde{s}_k$ is the sampled version of $\tilde{s}(t, \mathbf{I})$, and n_k is a complex zero-mean Gaussian distributed with variance σ_n^2.

The received sampled signal $\tilde{r}_k$ is sent through the FIR pre-filter p_n, with the set of tap coefficients $\mathbf{p}$. The output of the pre-filter is denoted by

$$r_k = \sum_{l=0}^{Pl} p_l^* \tilde{r}_{k-l} = \mathbf{p}^H \tilde{\mathbf{r}}_k \tag{44}$$

where

$$\mathbf{p} = \begin{bmatrix} p_0 & p_1 & \cdots & p_{N_p-1} \end{bmatrix}^T, \tag{45}$$

$$\tilde{\mathbf{r}}_k = \begin{bmatrix} \tilde{r}_k & \tilde{r}_{k-1} & \cdots & \tilde{r}_{k-N_p+1} \end{bmatrix}^T, \tag{46}$$

and $(.)^H$ denotes Hermitian transpose operation. The pre-filter output samples $\tilde{r}_k$ are fed into the MLSE, which implements the Viterbi Algorithm, resulting in the sequence estimate $\hat{\mathbf{I}}$, with

$$\hat{\mathbf{I}} = \begin{bmatrix} \hat{I}_0 & \hat{I}_1 & \cdots & \hat{I}_{N_{sym}-1} \end{bmatrix}^T \tag{47}$$

For each symbol estimate $\hat{I}_n$, the template signal that minimized the Euclidean distance is chosen as the desired signal $\tilde{t}_k$, as described in Section 1.1.2, and is given by

$$\tilde{t}_n = \sum_{m=0}^{Dl} q_m^* \hat{s}_{k-m} = \mathbf{q}^H \hat{\mathbf{s}}_k \tag{48}$$

where $\mathbf{q}$ is a vector containing the tap coefficients of the DIR filter,

$$\mathbf{q} = \begin{bmatrix} q_0 & q_1 & \cdots & q_{N_d-1} \end{bmatrix}^T \tag{49}$$

and $\hat{\mathbf{s}}_k$ is the sampled version of the symbol template signal.

$$\hat{\mathbf{s}}_k = \begin{bmatrix} \hat{s}_k & \hat{s}_{k-1} & \cdots & \hat{s}_{k-N_d+1} \end{bmatrix}^T \tag{50}$$

For each symbol, (N_s samples) the template signal $\tilde{t}_n$ is combined with the delayed pre-filter output $\tilde{r}_n$ to find the error e_n,

$$e_n = \tilde{r}_n - \tilde{t}_n \tag{51}$$

The error e_n is used to adapt the pre-filter and the DIR filter using a least mean square (LMS) algorithm. The pre-filter and DIR filter are updated by

$$\begin{aligned} \mathbf{p}_{n+1} &= \mathbf{p}_n - \mu_1 \tilde{\mathbf{r}}_k e_n^* \\ \mathbf{q}_{n+1} &= \mathbf{q}_n - \mu_2 \hat{\mathbf{s}}_k e_n^* \end{aligned} \tag{52}$$

Where μ_1 and μ_2 are the step size parameters and * denotes the complex conjugate operator.

3.2.1 Optimum pre-filter and DIR filter coefficients to minimize MSE

We wish to find the pre-filter and DIR filter coefficients that will minimize the power in e_n. We rewrite equation (51) in terms of (44) and (48).

$$e_n = \mathbf{p}^H \tilde{\mathbf{r}}_k - \mathbf{q}^H \hat{\mathbf{s}}_k \tag{53}$$

The MSE expression for e_n is then given by

$$\begin{aligned} E\left[e_k e_k^*\right] &= E\left[\left(\mathbf{p}^H \tilde{\mathbf{r}}_k - \mathbf{q}^H \hat{\mathbf{s}}_k\right)\left(\mathbf{p}^H \tilde{\mathbf{r}}_k - \mathbf{q}^H \hat{\mathbf{s}}_k\right)^*\right] \\ &= \mathbf{p}^H E\left[\tilde{\mathbf{r}}_k \tilde{\mathbf{r}}_k^H\right]\mathbf{p} - \mathbf{q}^H E\left[\hat{\mathbf{s}}_k \tilde{\mathbf{r}}_k^H\right]\mathbf{p} - \mathbf{p}^H E\left[\tilde{\mathbf{r}}_k \hat{\mathbf{s}}_k^H\right]\mathbf{q} + \mathbf{q}^H E\left[\hat{\mathbf{s}}_k \hat{\mathbf{s}}_k^H\right]\mathbf{q} \end{aligned} \tag{54}$$

We express $\tilde{\mathbf{r}}_k$ as the matrix product

$$\tilde{\mathbf{r}}_k = \mathbf{H}^H \tilde{\mathbf{s}}_k \tag{55}$$

where $\mathbf{H}$ contains the taps of the sampled impulse response of the BLF $c(t)$, such that

$$\mathbf{H} = \begin{bmatrix} c_0^* & 0 & \cdots & 0 \\ c_1^* & c_0^* & & \vdots \\ \vdots & \ddots & \ddots & 0 \\ c_{Nh-1}^* & & \ddots & c_0^* \\ 0 & & & c_1^* \\ & \ddots & & \vdots \\ 0 & & & c_{Nh-1}^* \end{bmatrix}_{N_H x N_p} \tag{56}$$

with $N_H = N_p + N_c - 1$, and $\tilde{\mathbf{s}}_k$ is given by

$$\tilde{\mathbf{s}}_k = \begin{bmatrix} \tilde{s}_k & \tilde{s}_{k-1} & \cdots & \tilde{s}_{k-M+1} \end{bmatrix}^T \tag{57}$$

Using (55), we substitute in the expectation matrices in (54) and make the following definitions,

$$\begin{aligned} E\left[\tilde{\mathbf{r}}_k \hat{\mathbf{s}}_k^H\right] &= E\left[\mathbf{H}^H \tilde{\mathbf{s}}_k \hat{\mathbf{s}}_k^H\right] \\ &= \mathbf{H}^H \mathbf{C} \end{aligned} \tag{58}$$

Where

$$\mathbf{C} = E\left[\tilde{\mathbf{s}}_k \hat{\mathbf{s}}_k^H\right] \tag{59}$$

and

$$\begin{aligned} E\left[\tilde{\mathbf{r}}_k \tilde{\mathbf{r}}_k^H\right] &= E\left[\mathbf{H}^H \tilde{\mathbf{s}}_k \tilde{\mathbf{s}}_k^H \mathbf{H}\right] \\ &= \mathbf{H}^H \mathbf{D} \mathbf{H} \end{aligned} \tag{60}$$

where

$$\mathbf{D} = E\left[\tilde{\mathbf{s}}_k \tilde{\mathbf{s}}_k^H\right] \tag{61}$$

In addition we let

$$\mathbf{G} = E\left[\hat{\mathbf{s}}_k \hat{\mathbf{s}}_k^H\right] \tag{62}$$

Assuming the signal estimate $\hat{I}_n = I_n$, the matrices **C**, **D,** and **G** are merely autocorrelation matrices of different dimensions. Equation (54) is written using the definitions in (58) and (60) to obtain

$$E\left[e_k e_k^*\right] = \mathbf{p}^H \mathbf{H}^H \mathbf{DHp} - \mathbf{q}^H \mathbf{C}^H \mathbf{Hp} - \mathbf{p}^H \mathbf{H}^H \mathbf{Cq} + \mathbf{q}^H \mathbf{Gq} \tag{63}$$

Taking the first gradient with respect to the pre-filter taps **p** and setting it equal to zero we obtain

$$\frac{\partial e_k e_k^*}{\partial \mathbf{p}} = \left(\mathbf{H}^H \mathbf{DHp}\right)^* - \left(\mathbf{H}^H \mathbf{Cq}\right)^* = 0 \tag{64}$$

The optimum pre-filter taps are found to be

$$\mathbf{p}_{opt} = \left(\mathbf{H}^H \mathbf{DH}\right)^{-1} \mathbf{H}^H \mathbf{Cq} \tag{65}$$

We now find the optimum taps for the DIR filter that would further minimize the MSE, the mean-square value of the error signal e_n. We substitute the optimum filter in (65) for the pre-filter in (63) and obtain

$$E\left[e_k e_k^*\right] = \mathbf{q}^H \left(\mathbf{G} - \mathbf{C}^H \mathbf{H} \left(\mathbf{H}^H \mathbf{DH}\right)^{-1} \mathbf{H}^H \mathbf{C}\right)\mathbf{q} \tag{66}$$

To avoid the trivial case of no MSE ($\mathbf{q} = \mathbf{0}$), and to satisfy implementation of the MLSE equalizer, the DIR filter taps are constrained to have unit energy

$$\mathbf{q}^H \mathbf{q} = 1 \tag{67}$$

To account for this constraint in (66), we use the method of Lagrange multipliers. The constraint is written as

$$\mathcal{C}(\mathbf{q}) = 1 - \mathbf{q}^H \mathbf{q} = 0 \tag{68}$$

and the Lagrangian becomes

$$\mathcal{L} = \mathbf{q}^H \left(\mathbf{G} - \mathbf{C}^H \mathbf{H} \left(\mathbf{H}^H \mathbf{D} \mathbf{H} \right)^{-1} \mathbf{H}^H \mathbf{C} \right) \mathbf{q} + \lambda (1 - \mathbf{q}^H \mathbf{q}) \tag{69}$$

where λ is a real Lagrangian multiplier. We then take the gradient with respect to the DIR filter taps and set it equal to zero to obtain,

$$\frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \left(\mathbf{G} - \mathbf{C}^H \mathbf{H} \left(\mathbf{H}^H \mathbf{D} \mathbf{H} \right)^{-1} \mathbf{H}^H \mathbf{C} \right) \mathbf{q} - \lambda \mathbf{q} = 0 \tag{70}$$

Consequently we have

$$\left(\mathbf{G} - \mathbf{C}^H \mathbf{H} \left(\mathbf{H}^H \mathbf{D} \mathbf{H} \right)^{-1} \mathbf{H}^H \mathbf{C} \right) \mathbf{q} = \lambda \mathbf{q} \tag{71}$$

This result shows that $\mathbf{q}$, the vector of tap coefficients for the DIR filter, must be an eigenvector of the matrix $(\mathbf{G} - \mathbf{C}^H \mathbf{H} \left(\mathbf{H}^H \mathbf{D} \mathbf{H} \right)^{-1} \mathbf{H}^H \mathbf{C})$ and that λ is the corresponding eigenvalue. To minimize (66), the optimum $\mathbf{q}$ is then

$$\mathbf{q}_{opt} = \text{eigenvector of } \left[\mathbf{G} - \mathbf{C}^H \mathbf{H} \left(\mathbf{H}^H \mathbf{D} \mathbf{H} \right)^{-1} \mathbf{H}^H \mathbf{C} \right] \text{ corresponding to the minimum eigenvalue} \tag{72}$$

3.2.2 Simulation Results

In this section we present simulation results for the MLSE Equalizer structure that implements the channel truncation method. The signal under study is a quaternary ($M = 4$) multi-h CPM signal with modulation indices {4/16, 5/16}, a symbol rate of 28 ksym/s, and sampled at

96 kHz. This configuration allows an integer number of samples per symbol such that $N_s = 4$. The channel is the BLF with impulse response $c(t)$.

The PF length N_p is set to extend over 10 symbols (41 taps) whereas the DIR filter is set to extend over one symbol period (5 taps). The pre-filter coefficients p_n and the DIR filter coefficients q_n are calculated a priori according to (65) and (72) respectively and assuming perfect knowledge of the channel. The in-phase and quadrature components for the impulse responses of the BLF, the PF, and the combined response between the BLF and the PF are shown in Figure 23.

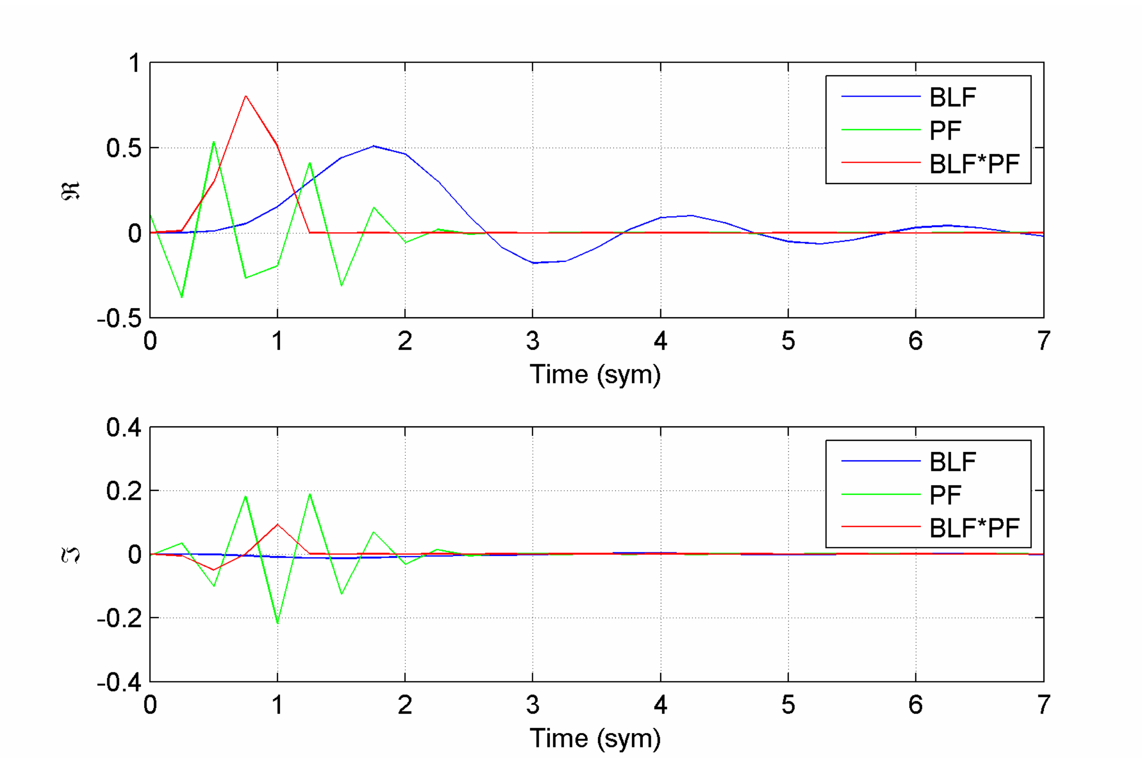

Figure 23. Impulse response for BLF, PF, and cascade of BLF and PF.

The BLF impulse response (blue) extends over more than 5 symbols, which would require a MLSE Equalizer with 32,768 correlators. However, when the BLF is combined with the PF (green), the combined response (red) concentrates most of the energy within the first symbol interval. To equalize the combined response of short duration, a MLSE CPM Equalizer of only 512 correlators would suffice. Thus, we can configure the MLSE CPM Equalizer to the DIR filter taps q_n, with a shorter impulse response duration than $c(t)$, and account for most of the energy in the channel. The magnitude, phase, and group-delay responses for the BLF $C(f)$, the pre-filter $P(f)$ and the combined response $C(f)P(f)$ are shown in Figure 24.

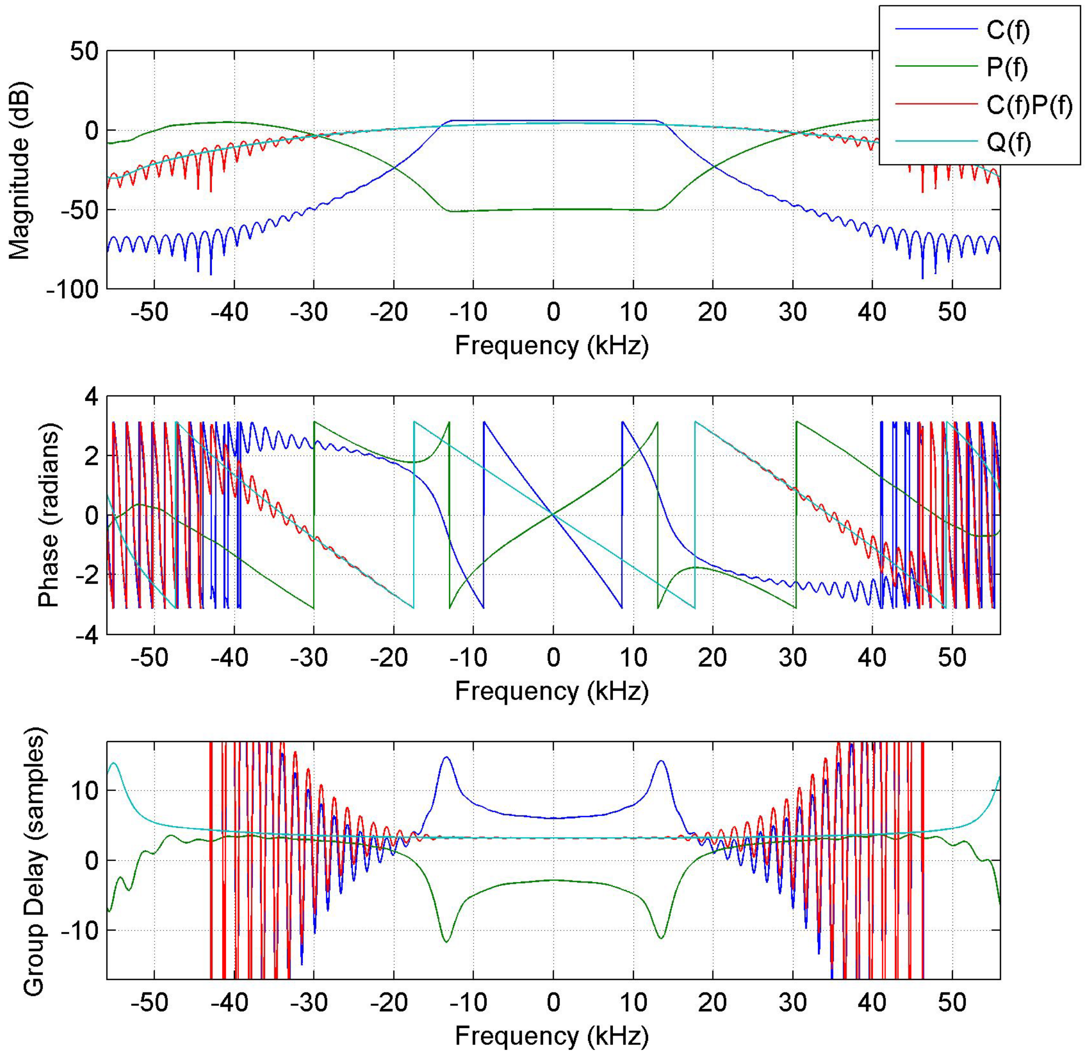

Figure 24. Magnitude, phase, and group-delay responses for the BLF C(f), pre-filter P(f), combined response C(f)P(f), and DIR Q(f).

When chosen optimally, the DIR filter $Q(f)$ (cyan) approximates the combined response of the channel and the PF $C(f)P(f)$ (red). Note that the PF response whitens the overall response (BLF + PF) of the channel by enhancing the BLF stopband frequencies. Furthermore, the PF also eliminates the phase response non-linearity at the edges of the BLF passband. In consequence the group-delay response inside the BLF passband is constant.

The performance of the channel truncation method using the Falconer and Magee structure is shown in Figure 25. The simulation is based on 100,000 symbols. The performance of the clean CPM signal and the band limited CPM signal are also shown, for reference.

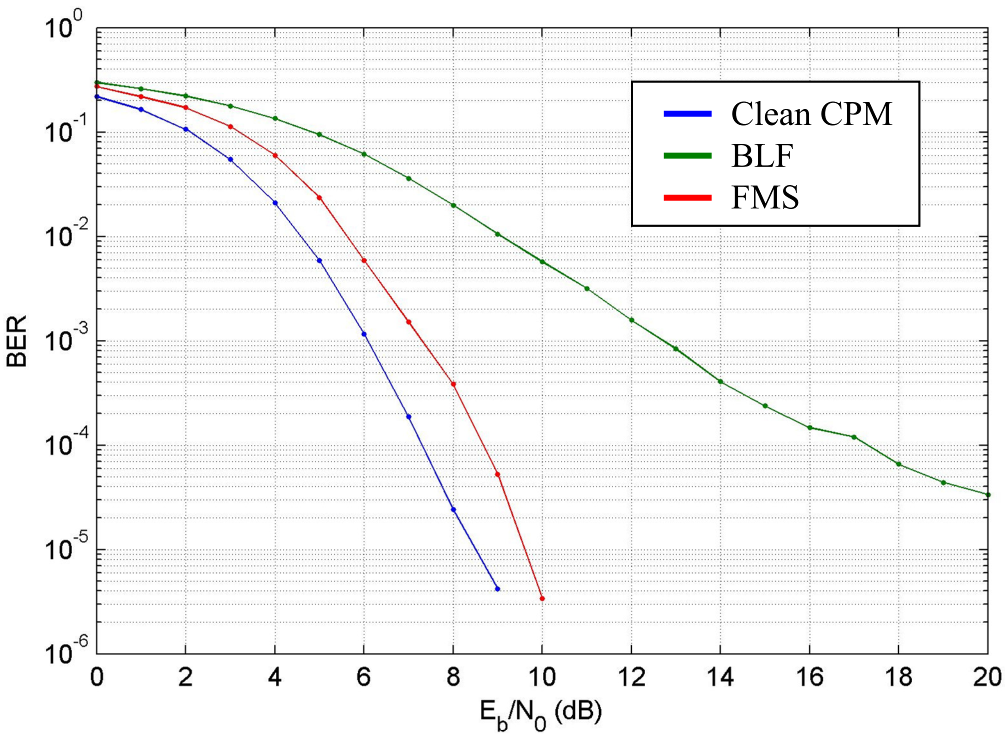

Figure 25. Performance of band limited CPM signal with Falconer and Magee channel truncation structure.

Figure 25 shows that the channel truncation method results in considerable performance gain. The performance gain is roughly 8 dB at BER of 10^{-4}, resulting in performance only 1.5 dB from the performance of the clean CPM signal. Most important is that this result is achieved with a memory requirement for the MLSE corresponding to only one symbol period.

3.3 Group-delay Compensator

This section presents an algorithm to obtain a group-delay compensator (GDC) aimed at reducing the amount of ISI produced by a band-limiting filter (BLF). This GDC is an FIR filter that reduces the group-delay variations occurring over the passband frequencies as produced by the BLF. The compensation is done by forcing the combined response of the baseband response of the BLF and the GDC to have nearly linear phase – and thus constant group-delay – over the passband. Note that the GDC compensates only for the phase variations and not the magnitude variations inside the passband. For reference, consider the system block diagram shown in Figure 26.

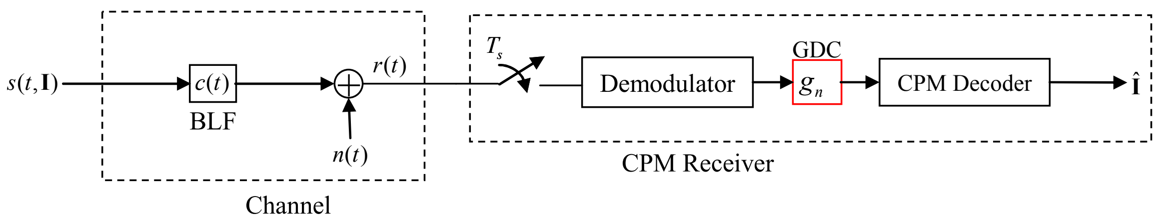

Figure 26. GDC system block diagram.

We let $\theta_C(f)$ be the demodulated phase response of the BLF with impulse response $c(t)$. The GDC g_n is designed to have a phase response $\theta_G(f)$ such that

$$\theta_C(f) + \theta_G(f) = af \tag{73}$$

where af is a linear function of frequency. By forcing an overall linear phase response, the group-delay (as seen by the receiver) is guaranteed to be constant and it will thus eliminate the phase distortions caused by the channel.

The algorithm for finding the GDC depends on having a reliable channel estimate. Thus, in addition to the algorithm for finding the group-delay we must develop a method to estimate the channel. For this reason, the remainder of this subsection is further divided into three subsections. The first subsection presents an algorithm for finding a channel estimate. The second subsection presents the algorithm for finding the GDC based on the channel estimate. Finally, in the third subsection, we present simulation results to illustrate the performance gain obtained by the addition of the GDC.

3.3.1 Finding the channel estimate

The algorithm to find the frequency response channel estimate is based on finding the ratio of spectral estimates associated with a portion of the input and its corresponding output signal of the channel. Due to the large number of points required to obtain a reliable frequency response estimate, the algorithm is executed offline. This is feasible since we assume the BLF to be time-invariant. The channel estimate obtained is then used to find the GDC which is subsequently implemented in the receiver.

The algorithm presented here uses the ratio of spectral estimates of the baseband versions of the input and output of the channel. To illustrate the origin of the data used in this algorithm with respect to the system model, Figure 27 shows the set-up for the data acquisition for the channel estimation algorithm.

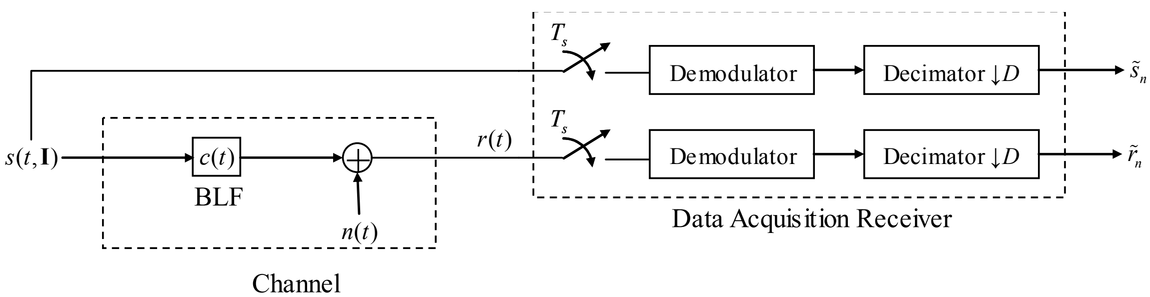

Figure 27. Model for data acquisition used in channel estimation.

The first step for obtaining the channel estimate is to demodulate and decimate the passband signals $s(t,\mathbf{I})$ and $r(t)$ which are the input and output of the channel respectively. Demodulation-decimation is done to ease the computational burden and to focus primarily on the passband of the signal. Both signals are sampled at f_s Hz and demodulated by f_c Hz, where f_c is the carrier frequency. Both signals are then decimated by the decimation factor D, which is given by

$$D = \left\lfloor \frac{f_s}{4\left(f_{co} + \Delta f\right)} \right\rfloor \tag{74}$$

where f_{co} is the passband cut-off frequency, Δf is the transition bandwidth and $\lfloor \bullet \rfloor$ denotes the rounding towards $-\infty$ operation (*floor* in MATLAB). The sampling frequency used in the analysis is then given by

$$f_{s,dec} = \frac{f_s}{D} \tag{75}$$

We denote the baseband-decimated versions of $s(t,\mathbf{I})$ and $r(t)$ as $\tilde{s}_n$ and $\tilde{r}_n$ respectively. A diagram representing the spectral content of $s(t,\mathbf{I})$ and $r(t)$, before and after demodulation-decimation, is illustrated in Figure 28.

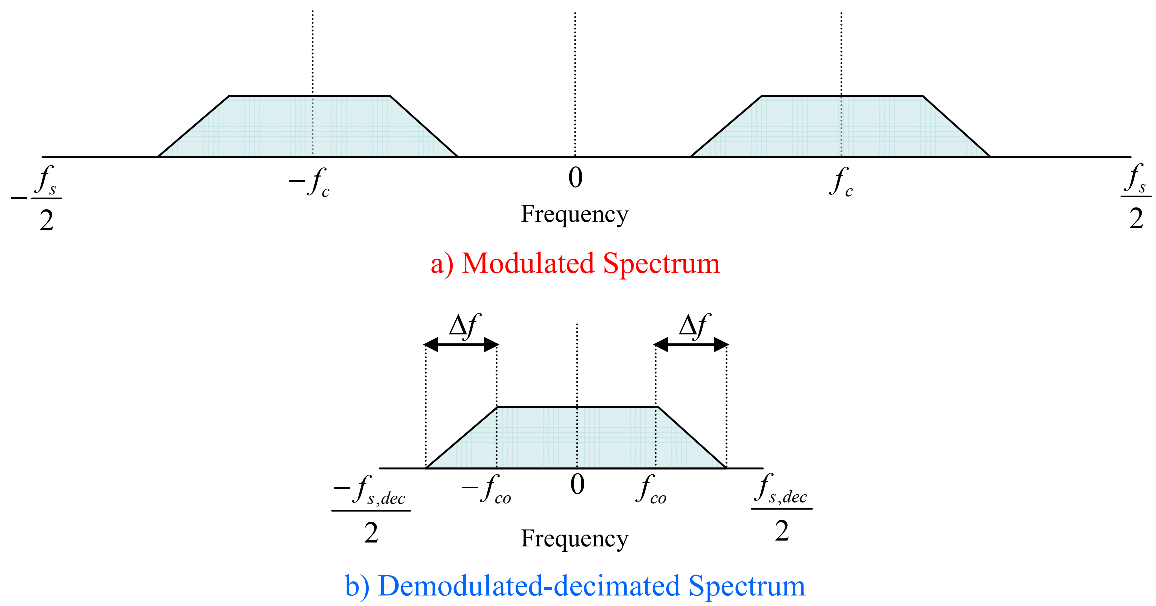

Figure 28. Representation for a) Modulated spectra and b) Demodulated-decimated spectra.

Before any computations are made, the signals $\tilde{s}_n$ and $\tilde{r}_n$ are aligned to compensate for the time delay resulting from the BLF. The alignment is performed by means of finding the lag n_0 for which the maximum absolute correlation occurs between the two signals. The output baseband signal $\tilde{r}_n$ is normalized in amplitude by multiplying by the constant gain G given by

$$G = \sqrt{\frac{\text{var}[s_n]}{\text{var}[r_n]}} \tag{76}$$

After the signals are aligned and have been normalized, both signals are windowed using a

Hanning window w_n. The spectral estimate is then computed using FFTs, with zero-padding to N_{FFT} points, where N_{FFT} is a large power of 2. The frequency response estimates for the baseband windowed signals $\tilde{s}_n$ and $\tilde{r}_n$ are then given by

$$\begin{aligned} \hat{S}(f) &= \mathrm{FFT}\{w_n \tilde{s}_n\} \\ \hat{R}(f) &= \mathrm{FFT}\{G w_n \tilde{r}_{n-n_0}\} \end{aligned} \tag{77}$$

To smooth both frequency response estimates, the number of FFT points is reduced from N_{FFT} to N_{FFT}/s, where s is the smoothing factor. As a result, each point in the smoothed frequency response vectors $S_s(f)$ and $R_s(f)$ is the average of s points of the original frequency response vectors. The channel frequency response estimate is obtained as the ratio of both smoothed spectral estimates, i.e.

$$\hat{C}(f) = \frac{R_s(f)}{S_s(f)} \tag{78}$$

To find the impulse response corresponding to the channel estimate, an inverse FFT is performed on the channel frequency response estimate so that the impulse response contains N_{FFT}/s samples.

$$\hat{c}(t) \underset{IFFT}{\rightarrow} \hat{C}(f) \tag{79}$$

The discrete coefficients of the impulse response are *fftshifted* to have the samples with the most energy adjacent to each other. For an N_{ce}-th order channel estimate, the N_{ce}+1 samples of the impulse response next to the maximum absolute value are selected with the maximum absolute value of the channel impulse response placed in the (N_{ce}/2+1)-st sample. This process is illustrated in Figure 29.

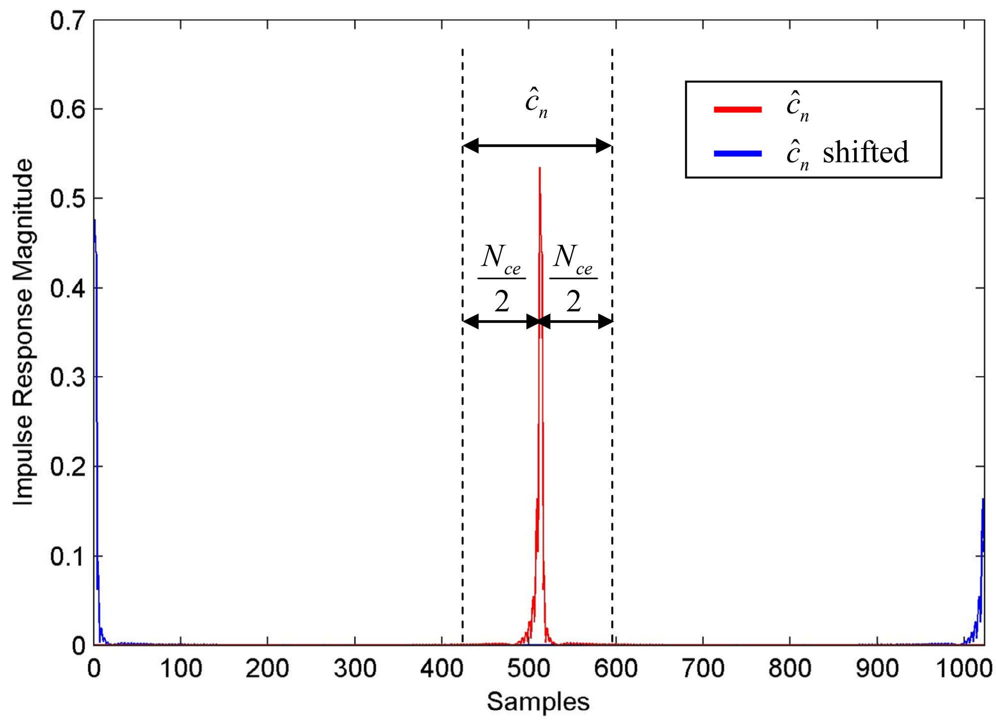

Figure 29. Impulse response magnitude for channel estimate.

3.3.2 Group-delay Compensator (GDC)

The GDC is designed to approximate a linear phase response for the cascade of the channel and the GDC. The GDC is based on the phase response of the channel estimate within the passband frequency range $|f| \le (f_{co} + \Delta f)$ with Δf Hz being the transition bandwidth. Given the phase response $\theta_{\hat{C}}(f)$ of the channel estimate, the phase response $\theta_G(f)$ of the GDC is chosen to satisfy

$$\theta_{\hat{C}}(f) + \theta_G(f) = af; \quad |f| \le (f_{co} + \Delta f) \tag{80}$$

where af is the best linear function of frequency that fits the phase response of the channel estimate $\theta_{\hat{C}}(f)$. The phase responses for the GDC, the channel estimate, and af are shown in Figure 30.

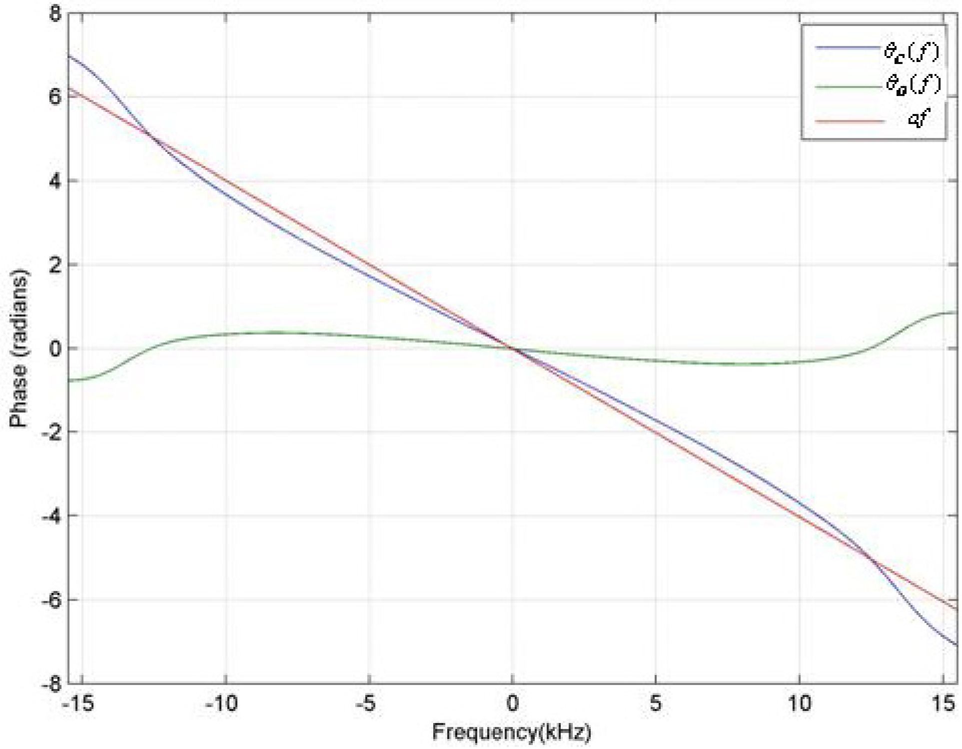

Figure 30. Phase responses for channel estimate, GDC, and the best linear estimate to the channel phase response.

To avoid phase discontinuities anywhere in the frequency range, the phase response of the GDC outside the frequency interval of interest is chosen so that the phase value and slopes at the boundary frequencies satisfy the phase values and slopes at the edges of the frequency interval of interest, as shown in Figure 31.

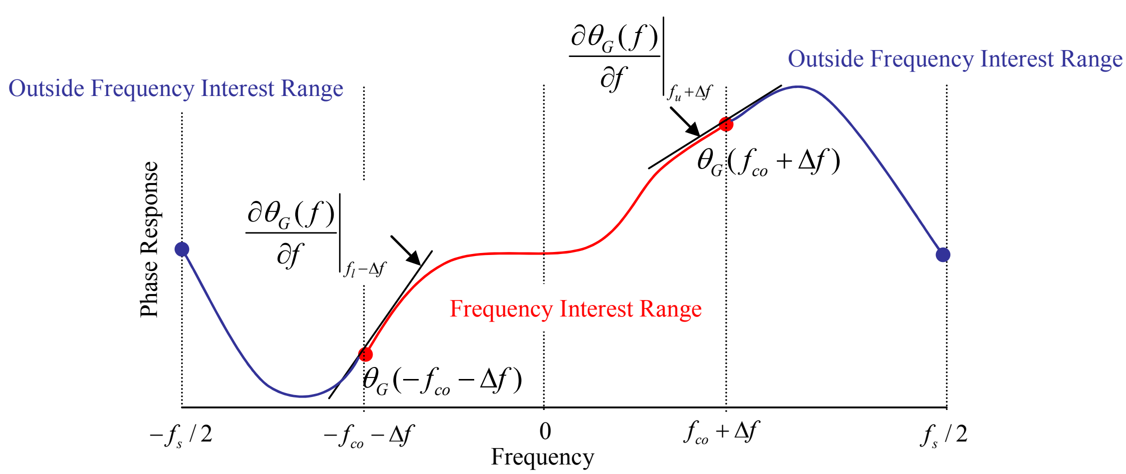

Figure 31. Phase Response for GDC.

The magnitude response of the compensator is chosen to be one for all frequencies since we wish to keep the magnitude response of the BLF. Thus, the frequency response of the GDC is given by

$$G(f) = \exp^{j\theta_G(f)} \tag{81}$$

For an N_{GDC}-th order GDC, the impulse response is obtained using the same procedure as that described to obtain the channel impulse response estimate (Figure 29). That is, from the desirable frequency response we compute the inverse FFT to obtain the impulse response. The impulse response is then *fftshifted* and the N_{GDC} samples around the main peak are selected as the discrete impulse response.

3.3.3 Simulation Results

This subsection presents results of the channel estimation algorithm and the GDC. Furthermore, simulation results are shown that validate the performance gain obtained by the addition of the GDC.

For ease in computation, this experiment is performed entirely at baseband. The signal $\tilde{s}_n$ is a clean baseband quaternary CPM signal with modulation indices {4/16 5/16}, with modulation rate 28 ksym/s. The sampling frequency is set to 112 kHz to allow an integer number of samples per symbol. The channel output signal $\tilde{r}_n$ is the result of filtering $\tilde{s}_n$ with the baseband model of the BLF. The magnitude and group-delay response of the BLF channel – in relation to the spectrum of the CPM signal – are shown in Figure 32.

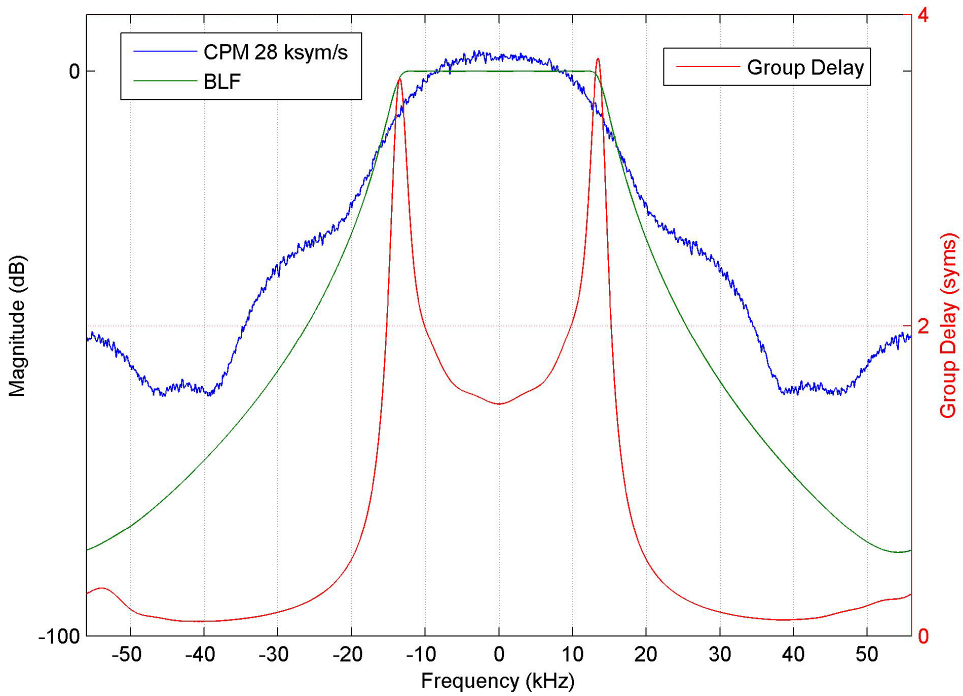

Figure 32. Magnitude and group-delay response of BLF w.r.t. spectrum of CPM signal.

For the channel frequency response estimation algorithm, 3.5 seconds of data (400k samples, 100k symbols) of $\tilde{s}_n$ and $\tilde{r}_n$ are used. Both signals are filtered by a Hanning window and the remaining parameters are set as follows. The total number of FFT points N_{FFT} is set to 2^{20} with a smoothing factor s of 1024. The order N_{CE} of the channel estimate is set to 60. A comparison between the channel estimate and the actual channel is shown in Figure 33.

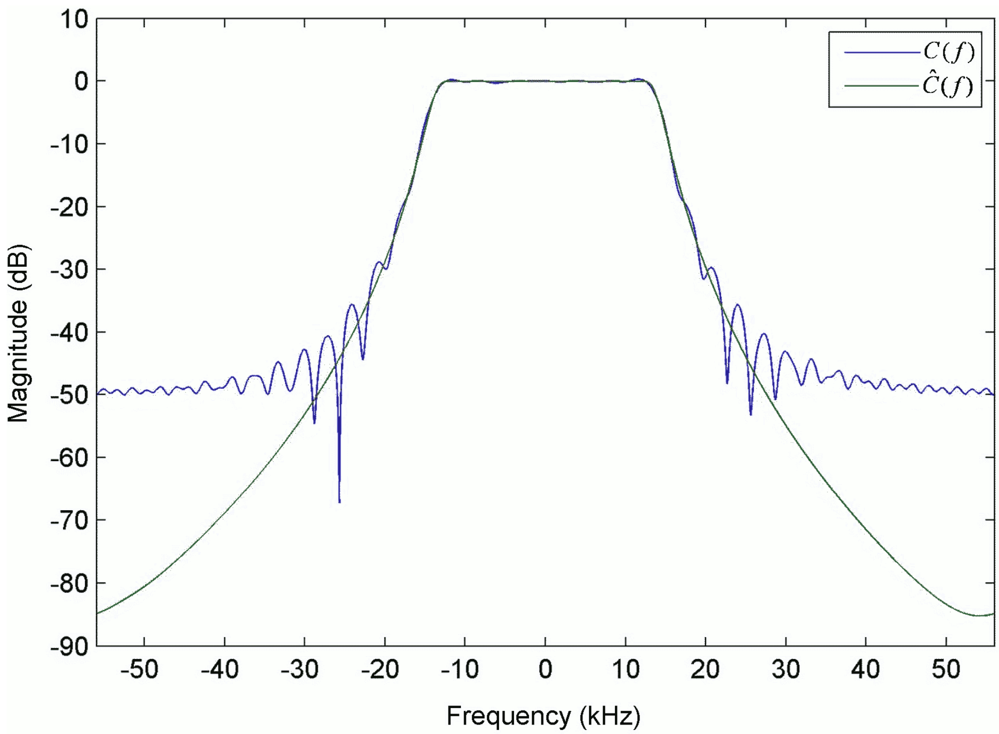

Figure 33. Comparison of the baseband magnitude response of the channel $c(t)$ and channel estimate $\hat{c}(t)$.

For the 60th order channel estimate, the estimate is relatively close down to -25 dB relative to the passband. For more accurate channel estimates, the order of the channel estimate can be increased.

The GDC is obtained based on the channel estimate and setting the transition bandwidth Δf to 3 kHz. The order of the GDC is set to 60. The group-delay response for the GDC, the BLF, and the channel frequency response estimate are shown in Figure 34.

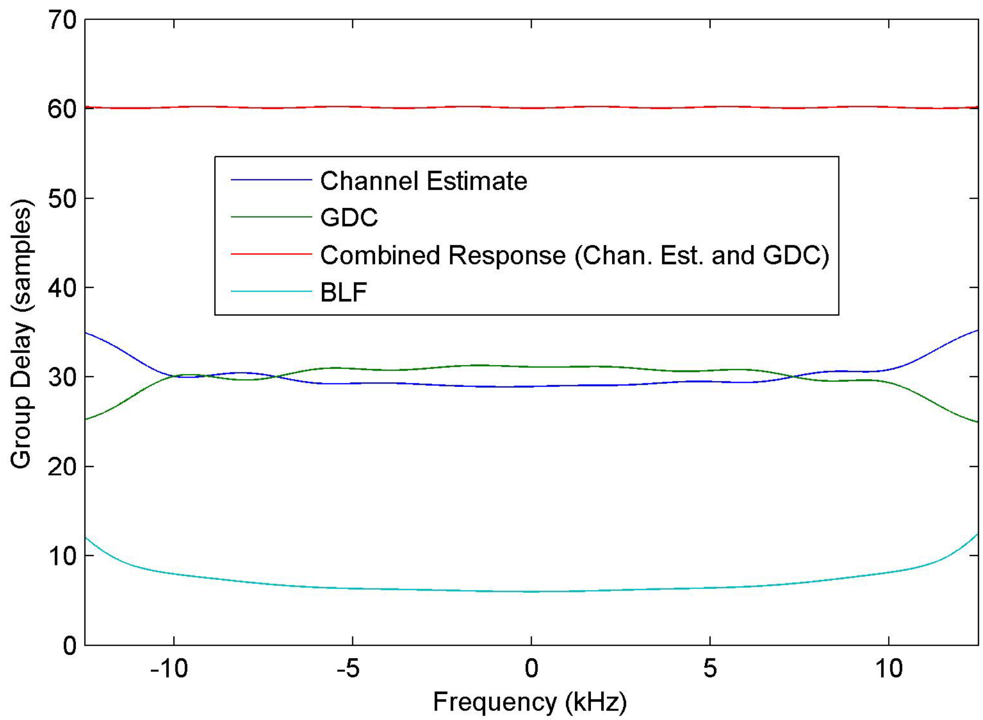

Figure 34. Group-delay responses for channel estimate and GDC.

The channel estimate group-delay response approximates that of the BLF with an offset of approximately 22 samples. The offset in the channel estimate is a result of the method by which the impulse response of the channel is computed. Since the maximum absolute value of the impulse response is placed at the ($N_{CE}/2+1$)-th sample, the assumed base group-delay is $N_{CE}/2+1$ samples.

The group-delay response of the GDC resembles the inverted group-delay response of the channel estimate. In this particular experiment, since the order of both the channel estimate and the GDC is the same, at 60, both group-delay responses exhibit the same base group-delay of 30 samples.

The combined response between the channel estimate and the GDC is relatively constant at 60 samples. The base group-delay from the combined response results from the addition of the group-delay response of the channel estimate and the GDC.

To assess the performance gained by the addition of the GDC, we measure the bit error rate (BER) for a system with and without GDC. In order to evaluate the loss due to non-constant group-delay, we provide the results of filtering the CPM signal with a linear phase (constant group-delay) 80th order FIR BPF filter $h(t)$ with similar frequency response characteristics as $c(t)$. The filter $h(t)$ has the same passband range but exhibits a steeper roll-off. The objective is to provide a lower bound on the performance of the GDC and to measure the performance loss due to band limiting of the signal without considering the negative effects of non-constant group-delay. A comparison between the magnitude and phase responses of $c(t)$ and $h(t)$ is shown in Figure 35.

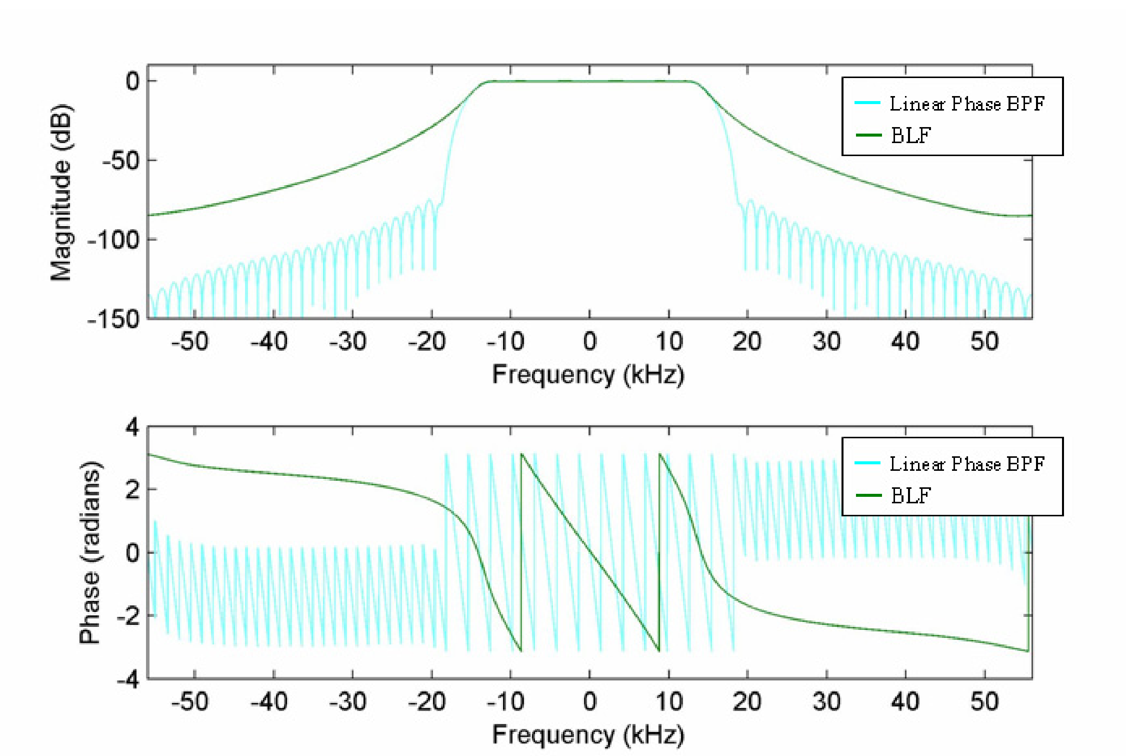

Figure 35. Comparison of frequency and phase response between BLF and linear phase BPF.

The performance results are obtained through Monte Carlo simulation and based on 1.5×10^{6} symbols (3 million bits). The results obtained are shown in Figure 36.

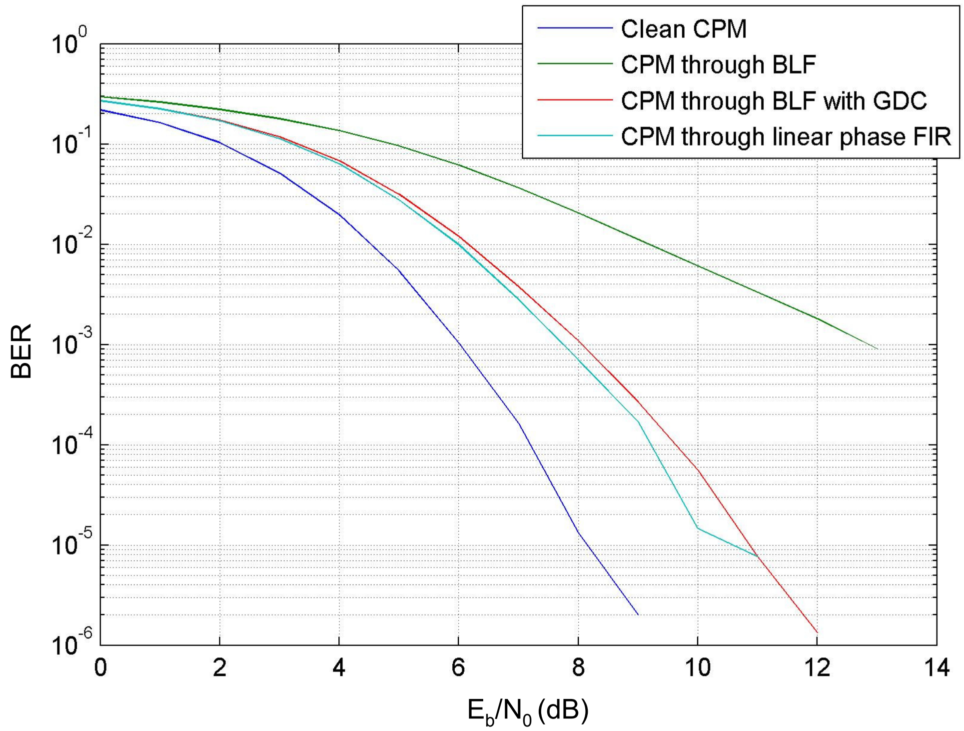

Figure 36. 60th order GDC performance in perspective.

The performance gain resulting from the presence of the GDC is substantial. The GDC reduces – by more than half – the loss incurred by the BLF and thus can be an important addition to the receiver. In addition, the performance of the GDC was within half a dB of the performance of the CPM signal transmitted through a linear phase bandpass filter, thereby indicating that most of the effect of the non-constant group-delay is being compensated by the GDC. Furthermore, since the GDC was implemented with a 60^{th}-order FIR, implementation is feasible at a low cost in terms of the number of clock cycles.

In Chapter 3 we presented several alternatives to the problem of equalization of CPM signals in band limited channels. In Section 3.1 we discussed the use of the pure MLSE CPM equalizer, which is the optimum form of decoding CPM signals in ISI channels, however, this requires a much bigger trellis. Later, in Section 3.2, we presented an adaptation of the Falconer and Magee structure (FMS) for CPM signals. This FMS used a pre-filter (PF) to shorten the effective length of the overall channel impulse response so that the MLSE CPM equalizer can then match a desired impulse response of shorter duration. The short duration impulse response naturally reduced the complexity of the MLSE required for near optimal performance. Finally, in Section 3.3, an FIR pre-filter was used to ensure a constant group-delay over the BLF passband.

4 Simulation Results - Comparison and Discussion

In this chapter we discuss the similarities between the FMS and the GDC system and compare them to one another in terms of performance, complexity, and versatility.

Performance is measured in terms of BER vs. E_b/N_0, where E_b/N_0 is a measure of the energy per bit E_b and the noise spectral density N_0. Complexity is measured in terms of the number of states required by the MLSE trellis to equalize the channel for near to optimal performance. Versatility expresses how easily the structure can be adapted to accommodate time-varying channels and accommodate the addition of further processing blocks.

4.1 Falconer and Magee Structure vs. Group-delay Compensator System

Both of the sub-optimum structures presented in this thesis have in common the use of a pre-filter to modify the overall impulse response of the channel. First, in the Falconer and Magee structure, a PF p_n is used to shorten the overall impulse response of the BLF discrete response c_n with the objective of reducing the symbol memory required by the MLSE process. In the GDC structure, a GDC FIR filter g_n is used to force the overall group-delay response of the channel to be constant. The combined response between the GDC and the BLF is obtained by convolving the GDC impulse response g_n with the BLF impulse response c_n Similarly, the combined response between the FMS pre-filter and the BLF is obtained from the convolution of their impulse responses, p_n and c_n respectively. The impulse responses for the GDC, the FMS pre-filter, and the BLF are shown in Figure 37.

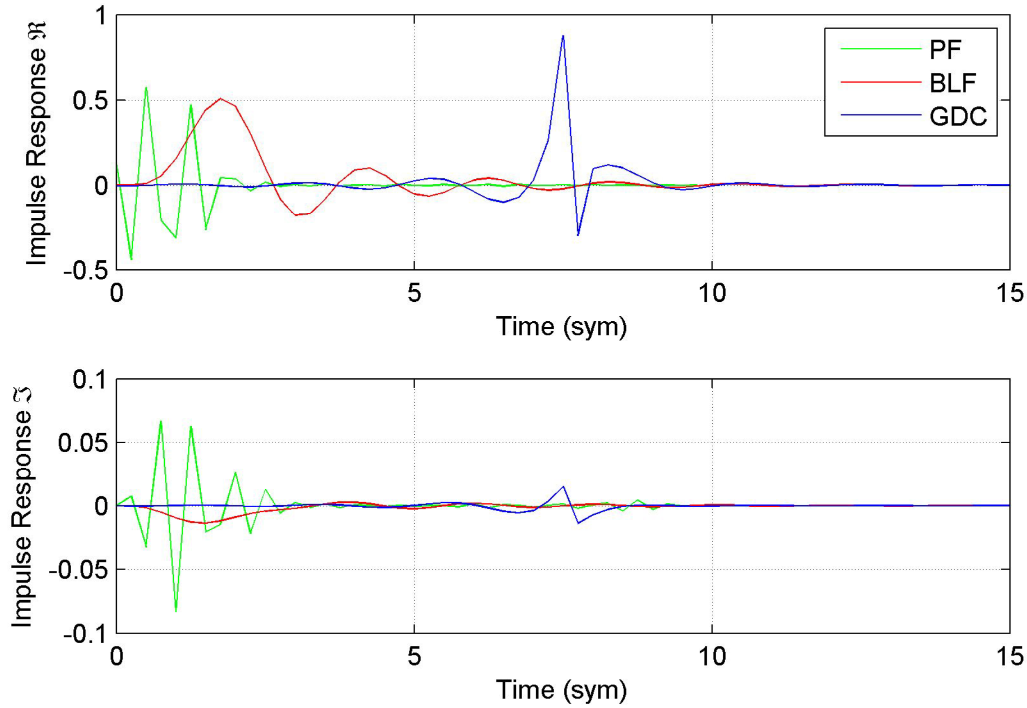

Figure 37. Impulse responses for FMS PF, GDC, and BLF.

The GDC is designed to approximate a linear phase response after convolving with the BLF. As a result, the combined response $c_n * g_n$ approximates a (conjugate) symmetric impulse response. In contrast, the FMS pre-filter is designed to force a combined response with its energy condensed within the first symbol interval so that it can be equalized with the smallest number of symbol memory locations in the MLSE process. The combined responses with the GDC and the FMS pre-filters are illustrated in Figure 37.

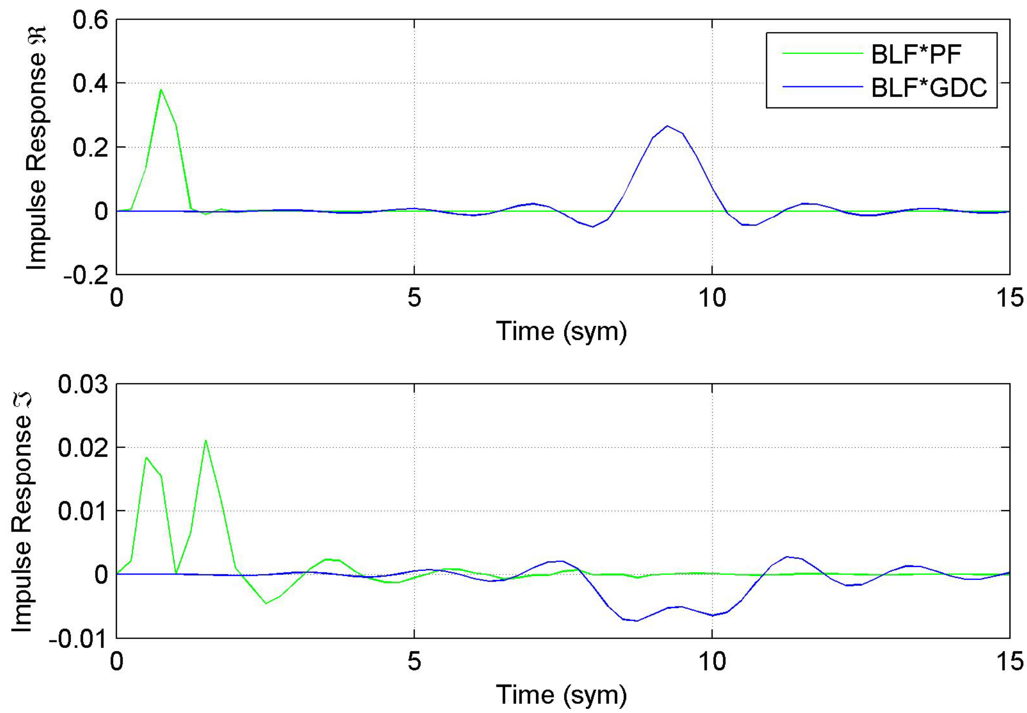

Figure 38. Combined impulse response between the GDC and FMS PF with the BLF.

Both combined responses have in common that they concentrate the energy in a relatively short time interval. The FMS PF concentrates the energy at the beginning of the impulse response, whereas the GDC concentrates the energy after a delay introduced by the GDC itself. However, as we can see in Figure 39, the combined impulse response of the BLF with the FMS PF is narrower than the impulse responses of the BLF and the combination of the BLF and the GDC.

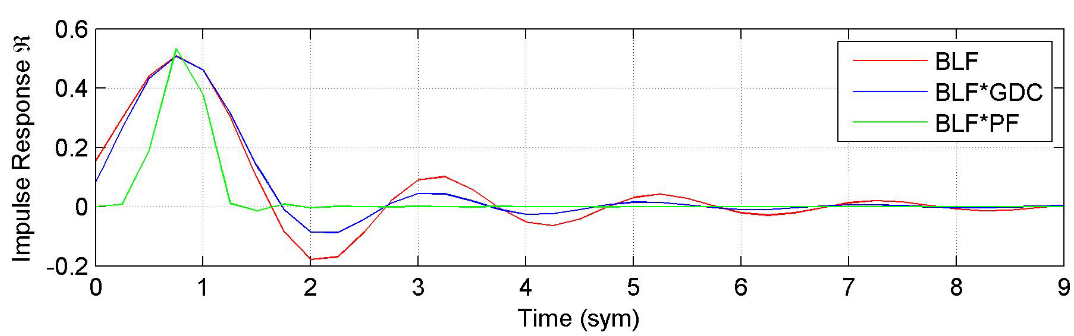

Figure 39. Aligned impulse responses for the BLF, and the overall responses BLF*PF and BLF*GDC.

The GDC does not shorten the overall impulse response by compacting the energy, instead the GDC shortens the length of the impulse response by reducing the magnitude of the side oscillations of the combined response. The latter suggests that the combined response between the GDC and the BLF could perhaps be equalized with the addition of one symbol of memory, thereby perhaps improving the BER performance.

Despite the different approaches, the phase and group-delay responses for both pre-filters are alike, in the sense that when combined with the BLF, both attempt to produce linear phase and constant group-delay. The group-delay responses for the GDC $G(f)$ and the FMS PF $P(f)$, as well as the group-delay responses when combined with the BLF, are shown in Figure 40.

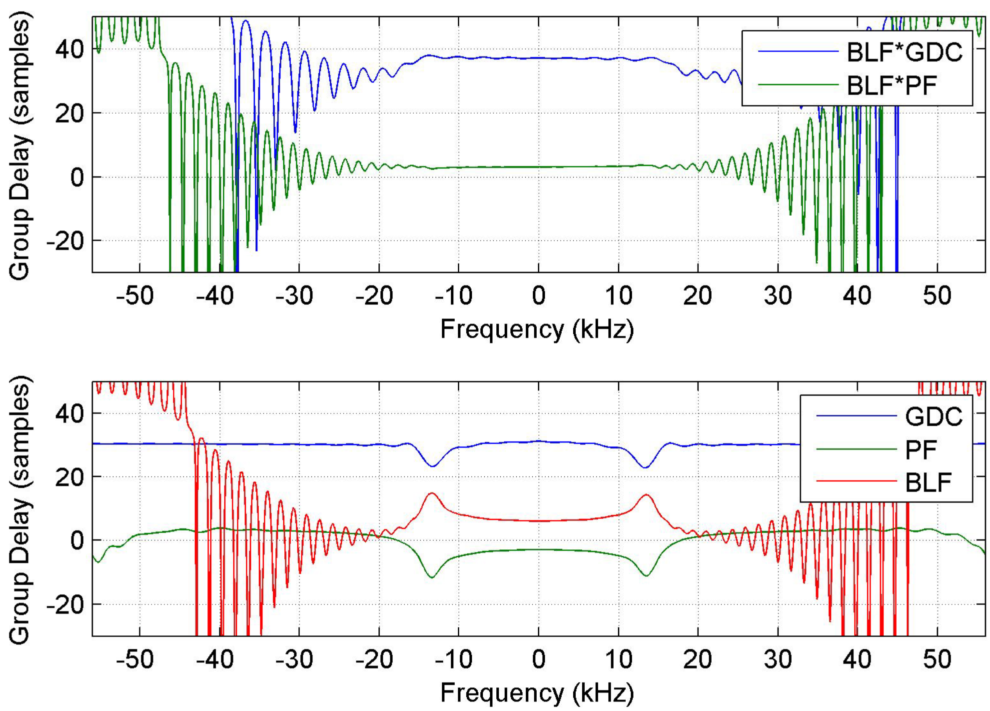

Figure 40. Phase and group-delay responses for pre-filter of sub-optimum equalizer structures.

Figure 40 shows that both combined group-delay responses (top plot) are relatively flat over the BLF passband. Surprisingly, the combined response between the FMS PF and the BLF (top, green) is visibly flatter over a longer frequency range. In addition, the average group-delay for the combined response between the GDC and the BLF (top, blue), is ≈ 37 samples, resulting from half the GDC order (60 in this example) plus the group-delay of the BLF (bottom, red).

Thus, the larger the GDC order, the more latency gets added to the system. On the other hand, the FMS PF (bottom, green) reduces the average group-delay of the system, since it adds negative group-delay.

Another difference between the GDC and the FMS PF is the magnitude response characteristic. The magnitude response in the GDC was designed to be constant for all frequencies so that the combined response with the BLF would preserve the magnitude response of the BLF. In contrast, the FMS PF has the effect of whitening the channel. Although the whitening of the channel can potentially enhance the out of band noise, this configuration is optimum in the MSE criterion and, as we will see, the BER performance is just 1 dB from that when receiving a clean CPM signal. The magnitude responses for the FMS PF, the GDC, the BLF, and the combined responses are shown in Figure 41 for a channel without AWGN.

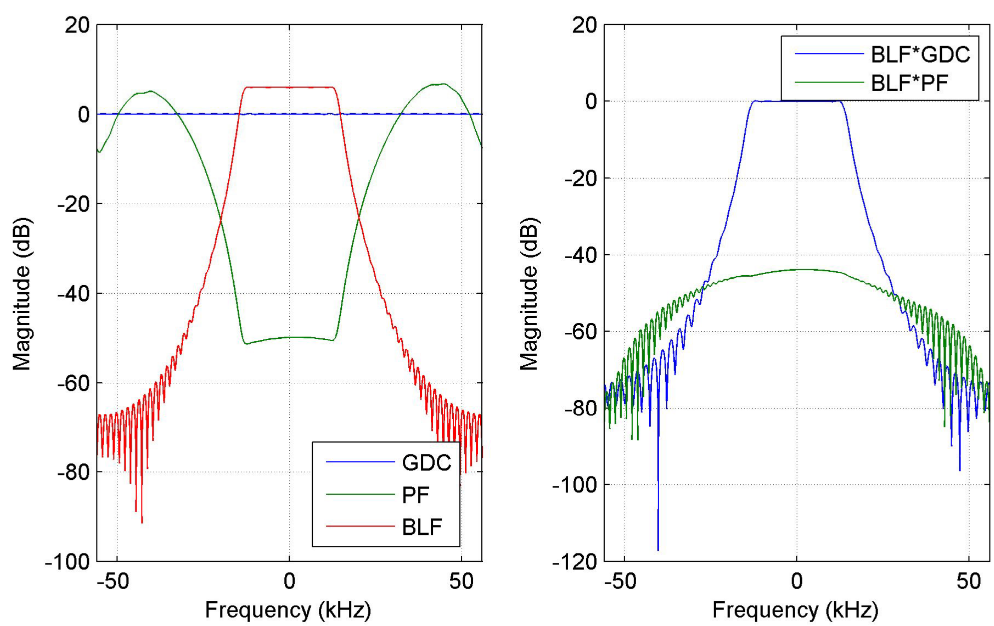

Figure 41. Magnitude responses for FMS PF, GDC, BLF, and combinations.

In Figure 41 (left) we observe that the magnitude response of the FMS PF $P(f)$ mirrors the magnitude response of the BLF so that the combined response with the BLF results in a flatter, whiter spectrum - as shown in Figure 41 (right). The flatter spectrum is a direct effect of the energy of the combined impulse response being concentrated in the first symbol interval (i.e. the magnitude response of a delta function is constant for all frequencies). In addition, we observe that even though the overall combined response is flatter, the selectivity of the channel inside the BLF passband is increased. Fortunately, this selectivity is the result of an impulse response that extends mostly over one symbol period and it can therefore be equalized by increasing the symbol memory in the MLSE equalizer by one.

4.2 Performance

Throughout Chapter 3 we have shown separately the BER performance for the different structures in a band limited channel. In this section we make a head-to-head comparison of all the BER performances obtained using different equalizer structures. The signal is chosen to be a dual-h, quaternary (M = 4) CPM signal with modulation indexes {4/16 5/16} as is commonly used in satellite communications. The simulations are performed at baseband using a sampling frequency of 112 kHz and the data rate is chosen to be 28 ksym/s, which results in having 4 samples per symbol. Figure 42 shows a comparison of the BER performance for the different equalization structures.

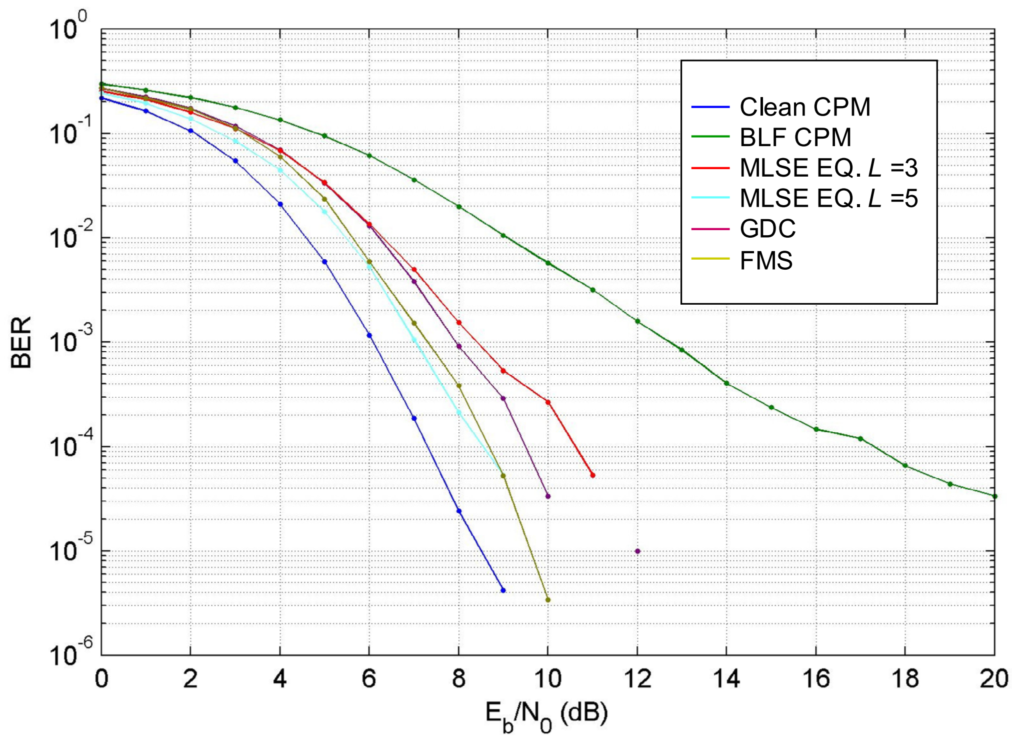

Figure 42. BER for different equalization approaches.

Figure 42 shows that the performance of the FMS is very close to that of the pure MLSE

CPM equalizer with $L = 5$. Nonetheless, the MLSE CPM equalizer, with $L = 5$ requires M^3 ($= 64$) times the number of trellis states required by the FMS, where M is the size of the CPM alphabet. The small difference in performance combined with the significant difference in complexity makes the FMS a far better option than the MLSE CPM equalizer with $L = 5$. Furthermore, the performance of the FMS is only roughly 1 dB off from that of the clean CPM signal. Also, the BER performance with the GDC (magenta) system is superior to that of the MLSE Equalizer with L = 3 (red) for Eb/N_0 greater than 6 dB. Moreover, the GDC system is off by less than a dB from the performance of the FMS. The performance improvement by the GDC is dependent on the severity of the group-delay variations in the BLF passband since it compensates only for the phase distortion.

4.3 Complexity

When it comes to complexity, clearly the FIR GDC filter is the most desirable option. If the BER requirements are not too severe, we have shown that a GDC can be implemented successfully with an FIR of 60^{th} order or less, depending on the BLF. Furthermore, a GDC implementation does not require additional symbol memory in the MLSE process. The complexity in the GDC approach comes from estimating the channel. Nonetheless, if the channel is considered time-invariant – as has been the case in this thesis – the channel estimation can be performed offline.

As for the complexity of the FMS, we have shown that with only one symbol of memory, we can obtain BER performance that is only about 1 dB from the performance for a clean CPM signal, equivalent to the performance of the MLSE CPM equalizer with four symbols of memory (L = 5). Thus, when compared to a pure MLSE CPM equalizer, the FMS saves significant complexity with a small trade-off in BER. Also, if the processing power allows it, the length of the DIR in the FMS can be incremented to add additional symbol memory and improve the BER performance.

4.4 Versatility

The FMS has the advantage that it can be implemented adaptively with just minor modifications. This advantage makes it very attractive for more complex systems were additional processing blocks may be added before the signal gets to the decoder. Another advantage to the FMS is that it does not require an external channel estimator block.

On the other hand, the current GDC structure is done strictly for a time-invariant channel and thus may not be so easily implemented adaptively. In order to implement an adaptive group-delay compensator method, several processes must be added. An efficient channel estimation block would be needed, followed by a separate group-delay compensator. Another limitation for the GDC system, is that the GDC improves the BER only when the effects of the passband group-delay variations are significant. In other words, the GDC may only be used for system in which the BLF has non-constant group-delay response within the BLF passband.

5 Conclusion and Suggestions for Future Work

5.1 Conclusion

In this thesis, we have studied the problem of equalization of continuous phase modulation (CPM) signals in band-limited channels. First, we provided details on the band limiting channel and the characteristics of CPM signals. During the introduction of CPM signals, we presented the trellis representation and the benefits of using multiple modulation indexes. Later, we introduced the optimum maximum likelihood sequence estimation (MLSE) CPM receiver for an additive white Gaussian noise (AWGN) channel. This receiver does not require symbol memory and thus it facilitates the description of the MLSE process which is implemented through the Viterbi Algorithm. The complexity of the MLSE CPM receiver is given by the size of the trellis of the CPM signal, since each trellis state corresponds to a discrete phase value. The MLSE CPM receiver was later expanded into the MLSE CPM equalizer, which is optimum for channels with intersymbol interference (ISI). The MLSE CPM equalizer implements a more complex trellis, in which each trellis state carries a phase value and also memory symbols. The size of the trellis for the MLSE CPM equalizer grows exponentially with each additional symbol in memory. The amount of symbol memory required depends on the spread of the channel impulse response.

We proposed two approaches to the problem of equalization of band-limited CPM signals. First, our efforts were focused on shortening the channel impulse response by means of a pre-filter. We implemented the channel truncation structure by Falconer and Magee and adapted it to work with CPM signals. In our second approach, we focused on eliminating the group-delay variations inside the channel passband using an FIR pre-filter.

The channel truncation approach implements a pre-filter to force the overall response of the channel to a desired impulse response of short duration. In this work, we obtained the optimum pre-filter and desired impulse response that minimizes the mean squared error. Using the optimum pre-filter and desired impulse responses we showed that using only one symbol worth of memory, we can obtain BER performance comparable to the MLSE CPM equalizer with 4 memory symbols. Moreover, the BER performance was only 1 dB off from the performance of a clean CPM signal.

In the group-delay compensator approach, we assumed the channel to be time-invariant and provided a method to design an FIR group-delay compensator filter so that – when convolved with the band limiting filter – it would result in constant group-delay over the filter passband. The magnitude response of the group-delay filter was relatively constant for all frequencies, so that the overall selectivity of the band limiting filter is kept. Furthermore, we showed that eliminating the group-delay variations in the band limiting filter passband reduced the amount of ISI, with the residual ISI being equivalent to that resulting from band limiting the signal with a linear phase filter. Moreover, we showed that for the band limited filter model used in this thesis, a group-delay compensator FIR filter of 60^{th} order effectively improved performance by more than 6 dB for a BER of 10^{-4} with respect to the band limited CPM. This approach is attractive only when the band limiting filter exhibits group-delay variation in the passband.

Finally, we discussed similarities between both approaches in terms of the effect of the pre-filters on the resulting channel and provided a comparison based on the criteria of BER performance, complexity, and versatility of the structure.

5.2 Future Work Recommendation

As discussed in Chapter 4, the current implementation of the group-delay compensator only considers time-invariant channels. For actual implementation this might not be the best assumption. To make the group-delay compensator structure adaptive, efficient methods for estimating the channel and finding the group-delay compensator are needed. In addition, a structure that combines the group-delay compensator approach with a MLSE CPM equalizer with low complexity may be used to improve performance. Furthermore, the observations on the combined response, for the band limiting filter, and the GDC, show that there is still room for improvement and that its performance can perhaps be improved using a better GDC estimate.

References

[1] G. D. Forney-Jr, "Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference," IEEE Transactions on Information Theory, vol. 18, pp. 363-378, 1972.

[2] D. D. Falconer and F. R. Magee Jr., "Adaptive Channel Memory Truncation for Maximum Likelihood Sequence Estimation," *Bell System Technical Journal*, vol. 52, pp. 1541-1562, 1973.

[3] W. U. Lee and F. S. Hill, "A maximum-likelihood sequence estimator with decision feedback equalizer," *IEEE Transactions on Communications*, vol. 25, pp. 971-979, 1977.

[4] A. Duel-Hallen and C. Heegard, "Delayed Decision Feedback Sequence Estimation," *IEEE Trans. on Comm.*, vol. 37, pp. 428-436, 1989.

[5] J. C. S. Cheung and R. Steele, "Soft-Decision Feedback Equalizer for Continuous Phase Modulated Signals in Wideband Mobile Radio Channels," *IEEE Trans. on Comm.*, vol. 42, pp. 1628--1638, 1994.

[6] H. C. Guren and N. Nolte, "Decision Feedback Sequence Estimation for Continuous Phase Modulation on a Linear Multipath Channel," *IEEE Trans. on Comm.*, vol. 41, pp. 280-284, 1993.

[7] J. G. Proakis, *Digital Communications*, 4th Edition ed: McGraw-Hill, 2001.

[8] I. Sasase and S. Mori, "Multi-h Phase Coded Modulation," *IEEE Communications Magazine*, vol. 29, pp. 46-56, 1991.

[9] B. W. Peterson, D. R. Stephens, and W. H. Tranter, "DFSE equalization of dual-h CPM over UHF MILSATCOM channels " presented at MILCOM 2002, Anaheim, CA, 2002.

[10] J. B. Anderson, T. Aulin, and C.-E. Sundberg, *Digital Phase Modulation*: Pleanum Press, New York and London, 1986.

[11] J. G. Proakis, *Digital Communications*, 4th ed. Boston: McGraw-Hill, 2001.

Vita

Andres Moctezuma was born on December 5th, 1981 in Mexico City, Mexico. He grew up in the city of Mixquiahuala, in the Mexican state of Hidalgo. In August 1999, after finishing high school, he moved to Montgomery, WV, to study at the West Virginia University Institute of Technology. In December 2003 he completed his B.S. in Computer Engineering from the same school. In January 2004, he enrolled in the graduate program in Electrical Engineering at Virginia Tech. At Virginia Tech he was a Research Assistant in the Digital Signal Processing Research Laboratory, where he worked on projects related to signal processing and digital communications. After completing his M.S.E.E., Andres will be working for TMEIC in Salem, VA, working as a Field Engineering in the area of systems and controls.