# CPM RECEIVER ISSUES: THE MATCHED FILTER BANK AND SEQUENTIAL SEQUENCE ESTIMATION

by

Gerhard Kramer

A Thesis
Presented to the Faculty of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree

MASTER OF SCIENCE

Department of Electrical & Computer Engineering
University of Manitoba
Winnipeg, Manitoba

ISBN 0-315-77766-4

# CPM RECEIVER ISSUES:

# THE MATCHED FILTER BANK AND SEQUENTIAL SEQUENCE ESTIMATION

BY

GERHARD KRAMER

A Thesis submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

# CPM RECEIVER ISSUES:
# THE MATCHED FILTER BANK AND SEQUENTIAL SEQUENCE ESTIMATION

BY

GERHARD KRAMER

A practicum submitted to the Faculty of Graduate Studies of the University of Manitoba in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

# ACKNOWLEDGEMENTS

I would like to thank Professor Ed Shwedyk for his support and friendship over the past two years. The time spent doing research with him, both during my summer in the Biomedical Engineering Laboratory and the past year in Digital Communications, has been both enjoyable and productive.

I am especially indebted to my family for providing their love and support throughout my undergraduate and Master's studies. To my parents, Günter and Martha Kramer, I wish to express my heartfelt gratitude for their enduring patience and constant encouragement.

Last, but certainly not least, I want to thank the Canadian People for their funding support and the Natural Sciences and Engineering Research Council of Canada for doing such a fine job in selecting the NSERC Postgraduate Scholarship recipients and administering the award. Monetary support from an NSERC operating grant (grant #OGP122127) is also gratefully acknowledged.

# ABSTRACT

This thesis deals with complexity reduction of Continuous Phase Modulation (CPM) receivers. Complexity reduction is vital for the implementation of the most power- and bandwidth-efficient CPM schemes as optimal detection is too expensive. Both the filter bank and the sequence estimation algorithm are considered.

First, CPM signal sets are decomposed to allow complexity reduction of the matched filter bank, the first stage in the receiver structure. A simple method for calculating the dimensionality of CPM signal sets is presented and it is shown that virtually all CPM schemes given in the literature need fewer matched filters for optimum detection than previously thought. Several schemes which require a small number of filters are presented and are shown to perform poorly compared to schemes with more complex signal sets. A suboptimal filter bank for these more complex schemes is developed.

The second stage of the receiver structure is the sequence estimation algorithm. Here a sequential algorithm, namely the Stack Algorithm, is applied to search the CPM state transition graph. This algorithm performs as well as the maximum likelihood Viterbi Algorithm in terms of error performance while requiring much less time to search the graph. However, simulations show that the Stack Algorithm is sensitive to adjacent channel interference and that care must be taken to model the channel properly.

# TABLE OF CONTENTS

# LIST OF FIGURES AND TABLES

# CHAPTER 1. INTRODUCTION

Continuous phase modulation (CPM) has been an area of intense research over the past decade. CPM schemes are more power- and bandwidth-efficient than their predecessors, and because of the many different possible configurations, there has been no lack of research topics. At the same time, the field has matured to a stage where the advantages and disadvantages of CPM have been recognized. The advantages of choosing a CPM modulation over one of the fundamental modulations, for example binary phase shift keying (BPSK) or quadrature amplitude modulation (QAM), are due to the restriction of phase continuity.

The most obvious advantage of phase continuity is bandwidth efficiency. Because CPM signals have smooth phase transitions from one symbol interval to the next, their bandwidth requirements are dramatically reduced from those digital modulations which allow discontinuities. A further advantage is power efficiency. Phase continuity can be considered a type of coding performed on the information to be transmitted, and the memory introduced by this "coding" can be exploited to achieve more noise immune schemes. Moreover, CPM signals have a constant envelope which enables the receiver to use a cheaper, more power efficient nonlinear amplifier. Thus, phase continuity enhances both the power and bandwidth characteristics of a modulation. The price paid for these advantages is implementation complexity at the transmitter and, more importantly, at the receiver.

The primary disadvantage of CPM is that the complexity of the optimal receiver becomes impractical for the most efficient schemes. The reason for this is that, because phase continuity introduces memory into the modulating process, the receiver must take this memory into account. This is done by using a graph search technique such as the Viterbi Algorithm to search the state transition graph defined by the CPM process (assuming maximum likelihood *sequence* estimation is desired). Moreover, because CPM schemes are nonlinear in general, the optimal receiver may require a large filter bank to obtain the necessary statistics for detection. Such factors will usually require the system designer to use a suboptimal detection scheme, both in terms of the filter bank and the graph search technique, but before becoming immersed in such detail, a brief historical introduction is given.

## 1.1. BACKGROUND

Continuous phase modulated schemes were introduced before 1963 at which time it was already well-known that CPM was more bandwidth efficient than non-continuous modulations [8]. However, it was not realized that knowledge of phase continuity should be used in designing the receiver. Ie. the signal was only observed over one symbol interval, thereby not using the full potential of phase continuity. This was discovered by de Buda [9] who also presented a simple receiver structure for a particular CPM scheme. Further work in designing more bandwidth efficient schemes and a practical transmitter and receiver was done by de Jager and Dekker [25]. Soon thereafter, many different approaches appeared (multi-h [3], partial response [14] and Aulin et al. [6] who analyzed CPM schemes in great detail). This led to the publishing of the text *Digital Phase Modulation* [1] in 1986, a book which is the standard reference in the area of CPM. The large bibliographies at the end of each chapter of this text bear witness to the huge amount of work done up to 1986. Since then, research has begun to focus more on implementation issues [5,23,24,35] and coding [17,22,29,33,34,39,46].

Complexity reduction for CPM modulations, both in terms of hardware and computations, is a crucial issue because optimal detection requires complex receivers. In [1, Chapter 8] simpler schemes for both the matched filter bank and the detection algorithm are presented. Some of these approaches yield excellent results and are useful for implementation. At the same time, there are still many open questions regarding CPM. For example, the signal set of the most power- and bandwidth-efficient schemes is very large and not much is known about its properties. Also, the suboptimum detection schemes are known to work well in the additive white Gaussian noise channel, but less is known about their performance under more severe and realistic channel impairments such as adjacent channel interference (ACI). These questions are addressed in the sequel, but first the communications model is introduced.

### 1.1.1. Models

The communications model considered here is given in its most general form in Figure 1.1. The sender transmits digital information, $\alpha$, via a modulator through a channel which corrupts the transmitted signal, s(t). The corrupted signal, r(t), is received by the demodulator and the information is estimated based on r(t) and knowledge of the modulation process. The estimated information, $\hat{\alpha}$, will in general be different than the input data

stream and it is the receiver's job to ensure that $\hat{\alpha}$ is as close as possible to the transmitted information, based on some measure of closeness.

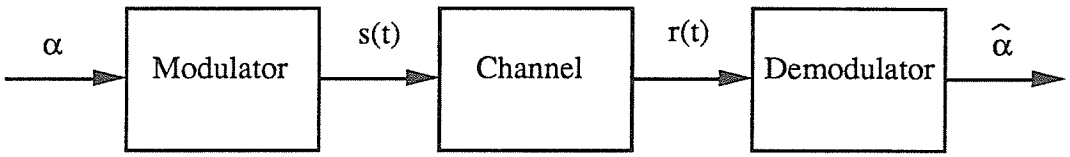


*Figure 1.1. The communications model.*

In designing and analyzing modulation schemes for this type of model, it is imperative that one choose a channel model which is both practical and easy to analyze. The model most widely used for this purpose is, of course, the Additive White Gaussian Noise (AWGN) channel. The received signal is given by: $r(t) = s(t)+n(t)$ where $n(t)$ is a white Gaussian stochastic process with spectral height $N_o/2$. Under this type of noise, signals can be analyzed using Euclidean space geometric concepts. This is further developed below, once the CPM model is introduced.

**The CPM Model**

The equations defining a CPM signal are:

$$s(t) = \sqrt{\frac{2E}{T}}\cos\left(2\pi f_c t + \phi(\alpha,t)\right)$$
$$\phi(\alpha,t) = 2\pi h\sum_{i=0}^{n}\alpha_i\, q(t-iT)\ ,\quad nT \le t \le (n+1)T \qquad (1.1)$$

where: E is the energy of s(t) per symbol interval, T,
$f_c$ is the carrier frequency,
h is the modulation index,
$\alpha_i \in \{-(M-1), -(M-3),\ldots,M-1\}$,
M is the size of the input symbol alphabet and
q(t) is the phase pulse.

The above equation, by itself, does not yet define a continuous phase modulated signal; to do so one must restrict q(t) to be a continuous function of time. The phase response is usually further restricted:

$$q(t) = \begin{cases} 0 & , t < 0 \\ 1/2 & , t \geq LT \end{cases} \qquad (1.2)$$

where L is the length of the frequency pulse, f(t), which is simply the derivative of q(t). If L=1, the CPM scheme is termed *full-response* and if L>1 it is called *partial response*. The restrictions of equation (1.2) are applied to maintain causality and to normalize q(t). Note that the phase pulse is not finite. Modulation schemes with this property (and the property that f(t) is finite) are sometimes referred to as frequency modulations, whereas schemes with finite q(t) are referred to as phase modulations.

Examples of often used phase pulses are shown in Figure 1.2. The motivation for choosing pulses of this kind is as follows:

1) The rectangular phase and frequency pulses, termed REC or LREC, are the simplest CPM schemes. The REC phase pulse is also the scheme with the smallest maximum slope, a property which translates into a small main lobe in its power spectral density (PSD). At the same time, the frequency pulse is discontinuous which adversely affects the asymptotic spectral performance.

2) The raised cosine phase and frequency pulses, termed RC or LRC, are smoother than REC. Because of this, an RC pulse yields good asymptotic spectral performance. However, the maximum slope of the RC phase pulse is twice as large as that of REC, a property which degrades main lobe performance.

These two points have described some aspects of the bandwidth properties of REC and RC schemes. Other important properties include the error performance and receiver complexity; these factors are considered in detail in Chapter 3.

### State Representation

The receiver must estimate the transmitted data based on the corrupted signal, r(t). To do this it is useful to have a state representation of the modulating process at the transmitter. Consider again equation (1.1). Based on the restrictions of equation (1.2) one can write:

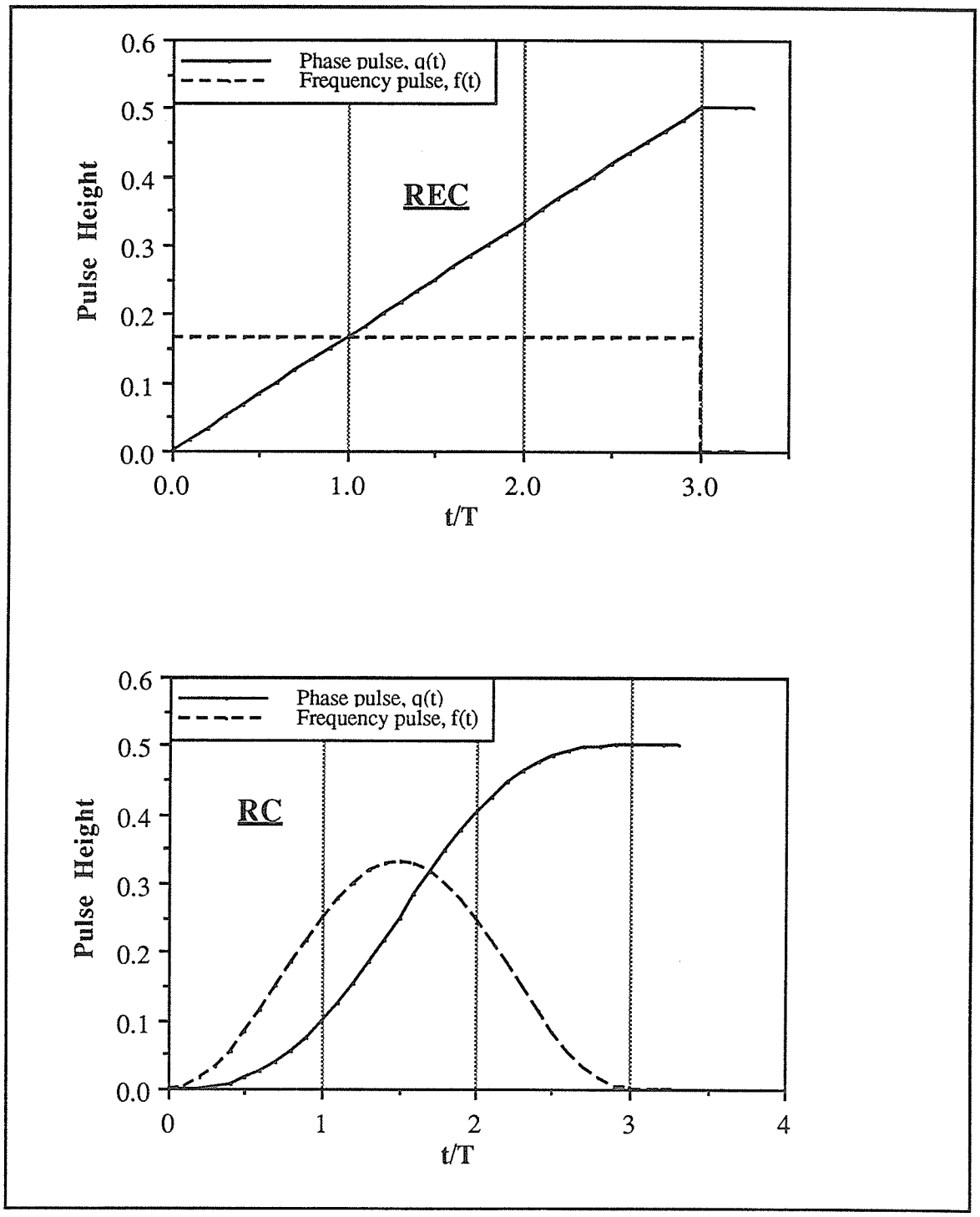


*Figure 1.2. RECtangular and Raised Cosine pulses.*

$$\begin{aligned}\phi(\alpha,t) &= 2\pi h\sum_{i=0}^{n} \alpha_i\, q(t-iT) \\ &= 2\pi h\alpha_n q(t-nT) + 2\pi h \sum_{i=n-L+1}^{n-1} \alpha_i\, q(t-iT) + \pi h\sum_{i=0}^{n-L} \alpha_i \\ &= \theta_c(\alpha_n,t) + \theta_{cp}(\alpha,t) + \theta_p(\alpha)\end{aligned} \tag{1.3}$$

where:

$\theta_c(\alpha_n,t)$ is the phase due to the current symbol.

$\theta_{cp}(\alpha,t)$ is due to q(t) extending over more than one symbol interval. The vector of L-1 symbols determining $\theta_{cp}$ is called the *correlative phase state*.

$\theta_p(\alpha)$ is termed the *phase state* and is due to q(t) being semi-infinite.

The above decomposition allows one to represent a CPM signal by a state transition sequence where the actual states are defined by the correlative phase state and the phase state, as determined by the vectors:

$$\begin{aligned}\chi_{cp} &= [\alpha_{n-1}, \alpha_{n-2},\ldots,\alpha_{n-L+1}] \quad \text{and} \\ \chi_p &= [\alpha_{n-L}, \alpha_{n-L-1},\ldots,\alpha_0]\end{aligned} \tag{1.4}$$

respectively. Thus, the transmitter follows a path through a graph where the next state is determined by the present state and the next symbol. Considering a CPM signal in this fashion is extremely useful in designing the receiver; to sequentially estimate the transmitted information the receiver should try to find the path followed by the transmitter. The manner in which this is done depends to a certain degree on the modulation index, h.

When h is irrational, the number of possible $\theta_p(\alpha)$ grows indefinitely as n increases. In this case a tree graph search technique must be employed. When h is rational, one can write h=k/p, where k and p are relatively prime and the number of possible $\theta_p(\alpha)$ is limited to p and 2p if k is even and odd, respectively. In this case the graph collapses into a trellis and a trellis search technique such as the Viterbi Algorithm can be used. Rational modulation indices are used throughout the rest of this chapter.

**An Alternate Representation**

More insight into the nature of a CPM signal is gained by allowing the use of an M-ary alphabet different than the one given in equation (1.1). Suppose the information symbols to be transmitted are $U_i \in \{0,1,\ldots,M-1\}$. These symbols can easily be accommodated in equation (1.1) by a simple substitution: $\alpha_i = 2U_i - (M-1)$. The phase in equation (1.1) then becomes:

$$\begin{aligned}\phi(\mathbf{U},t) &= 2\pi h\sum_{i=0}^{n} [2U_i - (M-1)]\, q(t-iT) \\ &= 4\pi h\sum_{i=0}^{n} U_i\, q(t-iT) - 2\pi h(M-1)\sum_{i=0}^{n} q(t-iT) \\ &= \psi_{info}(\mathbf{U},t) + \psi(t)\end{aligned} \tag{1.5}$$

The second term above, $\psi(t)$, is independent of the information sequence and can be ignored in the ensuing discussion. Further, one can write $\psi_{info}(\mathbf{U},t)$ as in equation (1.3):

$$\begin{aligned}\psi_{info}(\mathbf{U},t) &= 4\pi h U_n q(t-nT) + 4\pi h \sum_{i=n-L+1}^{n-1} U_i q(t-iT) + 2\pi h\sum_{i=0}^{n-L} U_i \\ &= \theta_c'(U_n,t) + \theta_{cp}'(\mathbf{U},t) + \theta_p'(\mathbf{U})\end{aligned} \tag{1.6}$$

Now if $h=k/p$ there are only p possible $\theta_p'(\mathbf{U})$ regardless of k. This is actually consistent with the above where, even though 2p $\theta_p(\alpha)$ are possible for odd k, only p $\theta_p(\alpha)$ are possible in any given symbol interval. This was developed by Rimoldi [38]. Using this fact, the number of states in a CPM scheme with rational modulation index can be enumerated. There are p possible phase states and, since the correlative phase state is defined by a vector of L-1 M-ary elements, there are $M^{L-1}$ possible correlative phase states. Combining these two numbers, the total number of states is $pM^{L-1}$ (it is implicitly assumed that the information symbols are statistically independent and that every symbol is equally likely in each symbol interval; if the symbols are correlated, as in the case of coded information symbols, then the number of states may be smaller [46]).

This alternate representation is adopted at times rather than the natural one of equation (1.1), as it lends itself readily to the determination of the dimensionality of CPM signal sets. More on this in Chapter 2, but consider now the CPM receiver.

### 1.1.2. The Optimal CPM Receiver

The optimal CPM receiver (optimal in terms of error performance) performs a maximum likelihood sequence estimation (MLSE) on the received signal. In the AWGN channel case, the MLSE detector can be described using well-known Euclidean geometric concepts [43]. Every signal can be thought of as a point in a higher dimensional space and the receiver simply locates this point, r(t), in this space. Since one is working in Euclidean space, the maximum likelihood sequence is the one which corresponds to that transmitter signal, $s_i(t)$, which is closest to r(t) in the squared distance or Euclidean norm sense; in other words minimize:

$$\int_0^{NT} [r(t) - s_i(t)]^2 \, dt = \int_0^{NT} [r^2(t) + s_i^2(t)] \, dt - 2 \int_0^{NT} r(t) \, s_i(t) \, dt \tag{1.7}$$

with respect to all sequences, where N is the total number of symbols transmitted. Since CPM signals have a constant envelope, the first integral in equation (1.7) is the same for all sequences (up to double frequency terms) so that minimizing the squared distance is equivalent to maximizing the correlation between r(t) and $s_i(t)$. Furthermore, this correlation can be written:

$$\int_0^{NT} r(t) \, s_i(t) \, dt = \sum_{n=0}^{N-1} \left\{ \int_{nT}^{(n+1)T} r(t) \, s_i(t) \, dt \right\} \tag{1.8}$$

so that sequence estimation can be performed on parts of the signal before the transmitter finishes transmitting. To accommodate this, the estimation process is split up into two parts: the matched filter bank which performs the symbol interval correlations and the sequence estimation algorithm which combines these correlations to form the total correlation of equation (1.8) (Figure 1.3).

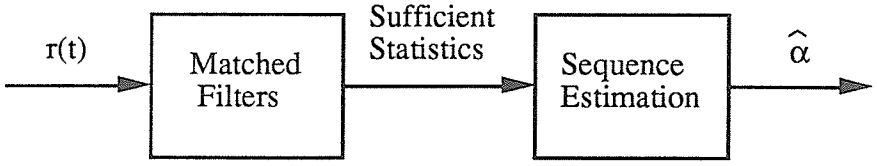


*Figure 1.3. The CPM receiver.*

## The Matched Filter Bank

Since CPM is in general a nonlinear modulation process, the most general receiver will require as many filters as the size of the signal set. The transmitter can send out $pM^L$ signals in any symbol interval ($pM^{L-1}$ states and M symbols per state), so that not more than $pM^L$ filters are needed with each filter matched to, or performing the correlation for, one of the signals in the signal set. However, some reduction is possible since the factor p comes from the term $\theta_p(\alpha)$ in equation (1.1) which is a constant phase offset over a given symbol interval. The set of p signals equal up to a constant phase offset lies in at most 2 dimensions so that not more than $2M^L$ matched filters are needed for optimum detection of any CPM scheme.

The filter bank will normally be implemented using baseband components so that a baseband representation of equation (1.1) is useful:

$$\begin{aligned} s(t) &= \cos(2\pi f_c t)\left[\sqrt{\frac{2E}{T}}\cos\left(\phi(\alpha,t)\right)\right] - \sin(2\pi f_c t)\left[\sqrt{\frac{2E}{T}}\sin\left(\phi(\alpha,t)\right)\right] \\ &= \cos(2\pi f_c t)\, I(\alpha,t) - \sin(2\pi f_c t)\, Q(\alpha,t) \end{aligned} \tag{1.9}$$

At the receiver, r(t) is converted to baseband using phase synchronized quadrature product detectors and a pair of low pass filters (Figure 1.4) [1].

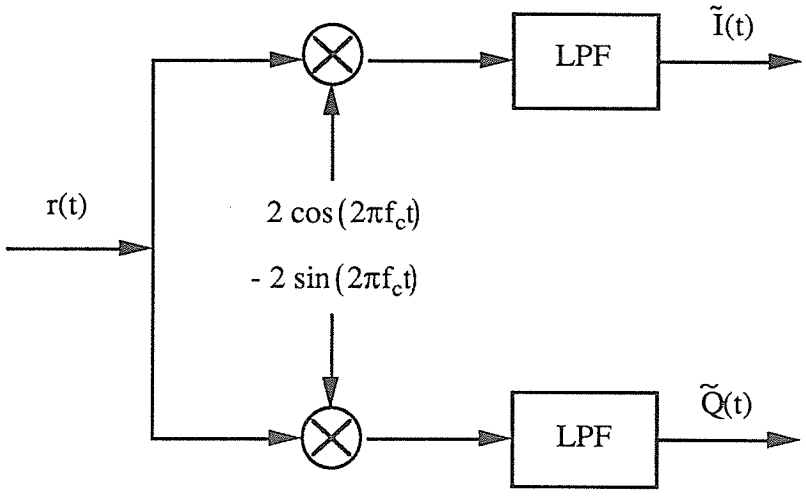


*Figure 1.4. The quadrature baseband receiver.*

The baseband matched filters perform their correlation operations on the noisy baseband components:

$$c_{ij} = \int_{jT}^{(j+1)T} 2\, r(t)\, s_i(t)\, dt \approx \int_{jT}^{(j+1)T} \left[\tilde{I}(t)\, I_i(t) + \tilde{Q}(t)\, Q_i(t)\right] dt$$

$$= \cos(\theta_p) \int_{jT}^{(j+1)T} \tilde{I}(t) \cos(\theta_c+\theta_{cp})\, dt - \sin(\theta_p) \int_{jT}^{(j+1)T} \tilde{I}(t) \sin(\theta_c+\theta_{cp})\, dt \qquad (1.10)$$

$$+ \sin(\theta_p) \int_{jT}^{(j+1)T} \tilde{Q}(t) \cos(\theta_c+\theta_{cp})\, dt + \cos(\theta_p) \int_{jT}^{(j+1)T} \tilde{Q}(t) \sin(\theta_c+\theta_{cp})\, dt$$

where the approximation is due to ignoring the double frequency terms and the notation of equation (1.3) has been adopted. The interval correlations in the j'th symbol interval, $c_{ij}$, are called branch metrics and are fed to the sequence estimation algorithm. Based on equation (1.10), four baseband matched filters are required for each set of p signals. Two of these filters are redundant due to the symmetry of the input symbol alphabet, so that one is left with $2M^L$ baseband matched filters, the same number as that needed for the high-frequency filters. Because this number is the same, no distinction is made between these two filters banks in the ensuing discussion and only the signal sets themselves are considered.

### The Sequence Estimation Algorithm

Maximum likelihood sequence estimation is performed via the Viterbi Algorithm which is really a brute force search through the trellis. The number of states is $pM^{L-1}$ which grows exponentially with the length of the frequency pulse, L. There are M branches leaving and entering each state, only one of which can represent the maximum likelihood path. Of all paths entering a state, only the one with the largest metric is kept; the other paths are discarded. This algorithm and various issues concerning its implementation have been widely studied (see [2,16,19], to name a few).

The primary disadvantage of the Viterbi Algorithm is the exponential growth in complexity with L. Furthermore, the most power- and bandwidth-efficient CPM schemes are those with large M and L. For these schemes the Viterbi receiver is too expensive. To alleviate this problem, reduced complexity search techniques need to be employed. This is further developed in Chapter 5.

## 1.2. SCOPE OF THESIS

Based on the above discussion, a vital issue in the practical implementation of CPM receivers is how to attain near optimal performance with simple receivers. However, before such receivers can be designed, the modulation process must be well understood. Therefore, this thesis deals first with optimal receivers and the underlying signal set. Chapter 2 analyzes the CPM signal set in some detail and leads to a broad discussion of special phase pulses in Chapter 3. The second section of the thesis, chapters 4 and 5, deals with suboptimal receivers. Chapter 4 addresses complexity reduction of the filter bank, whereas Chapter 5 is concerned with sequential sequence estimation.

# CHAPTER 2. CPM SIGNAL SETS

This chapter focuses on CPM signal sets. The goal is to determine the dimensionality of these signal sets since it is the dimensionality which determines the number of matched filters required for optimum detection. For example, amplitude shift keying (ASK) has a one dimensional signal set which can be demodulated with one matched filter, while quadrature amplitude modulation (QAM) spans two dimensions and hence requires two matched filters. In general, one matched filter will be required for each dimension.

A CPM signal set consists of $pM^L$ signals. The factor p is due to the constant phase offsets which define the phase states, so that the maximum number of dimensions required to represent any CPM signal set is $2M^L$, where the extra factor of 2 comes from the in-phase and quadrature components needed to obtain all phase offsets. However, for virtually all CPM phase pulses given in the literature, the dimensionality of the signal set is actually smaller than $2M^L$. To show this, a subset of the signal set called the dynamic signal set, as defined in section 2.1, is utilized. In section 2.2, the dimensionality of the REC class of phase pulses is obtained and is found to increase only linearly with the length of the frequency pulse, L. In section 2.3, the ideas of section 2.2 are generalized to other CPM schemes. The results lead to a general classification of CPM pulses and to schemes which require a small amount of hardware at the receiver (Chapter 3). Moreover, the results lead naturally to a suboptimal reduced-complexity detection scheme, as discussed in Chapter 4.

## 2.1. THE DYNAMIC SIGNAL SET

The $pM^L$ possible signals which the transmitter can send out during any symbol interval can be subdivided into $M^L$ classes where each class contains p signals identical up to a constant phase offset. For example, consider the duobinary (M=2, L=2) rectangular frequency pulse scheme with h=1/2 (so that p=2). There are then $pM^L=8$ signals in the signal set, with $M^L=4$ classes containing two signals each (Figure 2.1). Thus, for this particular CPM scheme $2M^L=8$ matched filters are needed, one set of matched filters for each class.

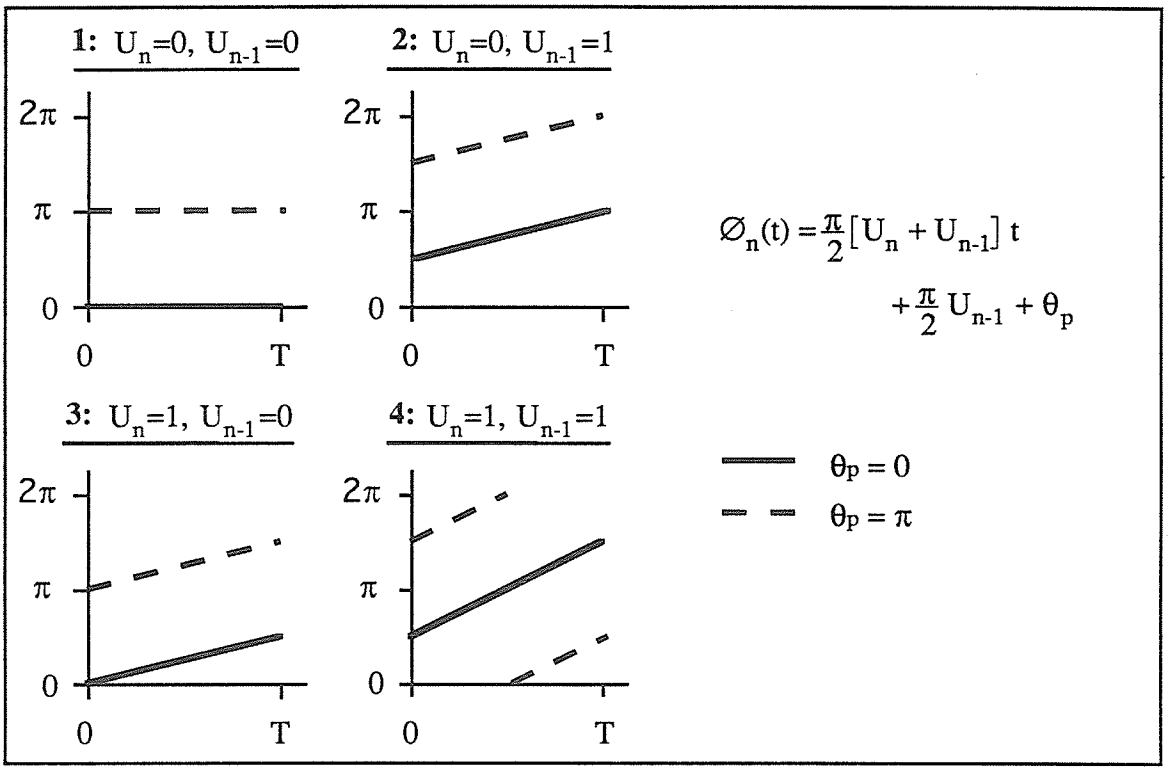


*Figure 2.1. Phase of the binary 2REC, h=1/2 signal set.*

Based on the above discussion, in a given symbol interval any CPM signal can be represented in the $2M^L$ dimensions (using equation (1.5)):

$$D_{U,n} = \cos\left(2\pi f_c t + \psi(t) + 4\pi h \sum_{i=0}^{L-1} U_{n-i} q(t-iT) + \frac{n\pi}{2}\right) \tag{2.1}$$

where $\mathbf{U}$ is one of the $M^L$ M-ary vectors of length L and n=0,1. However, the simplification need not stop here. It turns out that the sum in this equation contains many redundant terms; the problem is in determining how this redundancy comes about. To do this, partition the CPM signal set into a set of classes for which each class contains *all* of the signals identical up to a constant phase offset. Define this set of classes as *the dynamic signal set*, S; each class is represented by that signal which starts at zero phase. The size of the matched filter bank for optimum detection is therefore determined by the size of the dynamic signal set. In the next section, the number of such classes is counted for rectangular frequency pulse CPM schemes.

## 2.2. REC SCHEMES

A REC phase pulse is given by:

$$q(t) = \begin{cases} 0 & , t < 0 \\ \frac{t}{2LT} & , 0 \le t < LT \\ 1/2 & , t \ge LT \end{cases} \tag{2.2}$$

This class of pulses includes CPFSK (L=1) and MSK (M=2, L=1, h=1/2). The phase state independent portion of the phase of a REC signal is given by:

$$\theta(\mathbf{U},t) = 4\pi h \sum_{i=0}^{L-1} U_{n-i} q(t+iT-nT) \qquad , nT \le t < (n+1)T \tag{2.3}$$

Inserting (2.2) into (2.3) yields an equation with two distinct parts:

$$\theta(\mathbf{U},t) = \theta_t(\mathbf{U},t) + \theta_c(\mathbf{U}) = 4\pi h \frac{t}{2LT}\left[\sum_{i=0}^{L-1} U_{n-i}\right] + 4\pi h \left[\sum_{i=0}^{L-1} U_{n-i} \frac{(i-n)}{2L}\right] \tag{2.4}$$

The first term in the above equation defines the dynamic signal set, whereas the second term is a "static" constant phase offset. This implies that the number of dimensions needed to represent the signal set will not be more than twice the number of possible $\theta_t(\mathbf{U},t)$. But this is just twice the number of different values which the sum $\sum_{i=0}^{L-1} U_{n-i}$ in equation (2.4) can take on, and since $\sum_{i=0}^{L-1} U_{n-i} \in \{0, 1,..., L(M-1)\}$, there are only $|S| = L(M-1)+1$ different $\theta_t(\alpha,t)$. For example, consider again the duobinary CPFSK signal set. From Figure 2.1 it is immediately apparent that there are only L(M-1)+1=3 distinct slopes in the signal set, as opposed to $M^L$=4. Thus, only six matched filters, rather than eight, are needed for optimum detection. Such decreases in dimensionality were noticed by several authors [1,14,22] and are consistent with results from partial response signaling [26].

The above indicates that the number of matched filters needed for optimum detection of REC CPM schemes is $2|S| = 2[L(M-1)+1]$, a number which increases linearly with both M and L and not exponentially as for general CPM schemes. Thus, optimal detection does

not require a prohibitively large bank of matched filters, even for relatively large M and L. For example, for M=4 and L=3 only 20 matched filters are needed instead of the 128 for general schemes (it should be noted that REC schemes with M=2, L>3 and $M \geq 2^n$, n>1, L>1 exhibit undesirable distance properties; this is discussed in more detail in Chapter 3).

Since q(t) is a linear function, the REC dynamic signal set is particularly nice and consists of L(M-1)+1 equally spaced frequencies:

$$f_n = f_c + \frac{h}{2LT}[-L(M-1) + 2n] \quad , n = 0, 1,..., L(M-1) \tag{2.5}$$

Other CPM pulses do not have as nice a representation. However, for most schemes, some simplification is still possible.

## 2.3. OTHER CPM SCHEMES

The approach for REC can be generalized to other phase pulses. In the ensuing discussion, a simple bound for the dimensionality of any CPM scheme is derived, as well as results for particular schemes.

We reiterate that to obtain the number of dimensions spanned by a CPM signal set it is necessary to count the number of different dynamic phase transitions possible over one symbol interval. To do this, one must consider the individual "chips" of the CPM phase pulse. This is illustrated with the following example.

### 2.3.1. An Example: The 4RC Pulse

The 4RC pulse has four unique chips, ignoring constant phase offsets (Figure 2.2). Time translating each chip into the interval $0 \leq t / T < 1$ yields:

$$\begin{aligned} q_0(t) &= \frac{t}{8T} - \frac{1}{4\pi}\sin\left(\frac{\pi t}{2T}\right), \quad q_1(t) = \frac{t}{8T} - \frac{1}{4\pi}\cos\left(\frac{\pi t}{2T}\right) \\ q_2(t) &= \frac{t}{8T} + \frac{1}{4\pi}\sin\left(\frac{\pi t}{2T}\right), \quad q_3(t) = \frac{t}{8T} + \frac{1}{4\pi}\cos\left(\frac{\pi t}{2T}\right) \end{aligned} \tag{2.6}$$

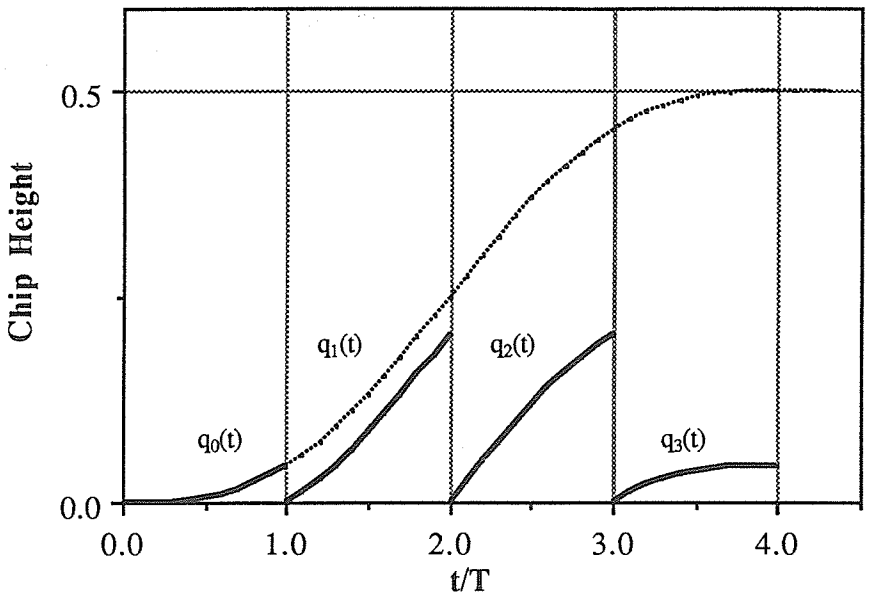


*Figure 2.2. Chips of the 4RC pulse.*

These four chips, however, are linearly dependent and can be expressed as:

$$\mathbf{Q}(t) = \begin{bmatrix} q_0(t) \\ q_1(t) \\ q_2(t) \\ q_3(t) \end{bmatrix} = \mathbf{Q}_B\mathbf{B}(t) = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} -\frac{1}{4\pi}\sin\left(\frac{\pi t}{2T}\right) \\ -\frac{1}{4\pi}\cos\left(\frac{\pi t}{2T}\right) \\ \frac{t}{8T} \end{bmatrix} \tag{2.7}$$

This expression is simply the representation of the 4RC chips by a linearly independent set of chips which span the dynamic signal set, S, and thus $\mathbf{B}(t)$ forms a basis for the linear space spanned by the elements of S. Of course, this basis is not unique. For example, one could write:

$$\mathbf{Q}(t) = \mathbf{Q}_B\mathbf{B}(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{t}{8T} - \frac{1}{4\pi}\sin\left(\frac{\pi t}{2T}\right) \\ \frac{t}{8T} - \frac{1}{4\pi}\cos\left(\frac{\pi t}{2T}\right) \\ \frac{t}{4T} \end{bmatrix} \tag{2.8}$$

It turns out that this second representation is more suited for our purposes. The signal sent out in the n'th interval is:

$$\begin{aligned}\theta_t(t) &= [\ U_n \ \ U_{n-1} \ U_{n-2} \ U_{n-3}\ ] \cdot \mathbf{Q}(t) \\ &= \widetilde{\mathbf{U}} \cdot \mathbf{Q}_B\, \mathbf{B}(t)\end{aligned} \tag{2.9}$$

Using the representation of equation (2.8), one obtains:

$$\begin{aligned}\theta_t(t) &= \widetilde{\mathbf{U}}\, \mathbf{Q}_B \cdot\ \mathbf{B}(t) \\ &= [\ U_n - U_{n-2} \quad U_{n-1} - U_{n-3} \quad U_{n-2} + U_{n-3}\ ] \cdot \mathbf{B}(t)\end{aligned} \tag{2.10}$$

The actual number of signals possible is the number of unique vectors which $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$ can become. This is just the size of the space generated by the rows of the 4x3 matrix $\mathbf{Q}_B$ when scalar multiplications by only the M-ary data symbols are allowed. For example, if M=2 there are $2^4$=16 possible $\widetilde{\mathbf{U}}$ but only 15 possible $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$. Thus, the representation of equations (2.7) and (2.8) gives us a straightforward approach to counting the number of signals in the dynamic signal set.

Counting the number of unique $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$ is a brute force approach to counting the number of possible signals. However, a simple upper bound can be obtained by noticing that each element of $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$ can take on only 2(M-1)+1 values. Since each element represents the contribution of one of the linearly independent portions of the dynamic signal set, the size of this signal set, |S|, cannot be more than:

$$|S| \le [2(M-1) + 1]^3 = (2M - 1)^3 \tag{2.11}$$

This equation is only an upper bound because it is based on the assumptions that each position in $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$ is independent and that all values between 0 and 2(M-1) can occur. The former assumption is obviously not true here.

Equation (2.11) tells us that |S| increases asymptotically as $8M^3$ and not as $M^L$=$M^4$, as for general CPM signals. It is now also clear why the second basis in equation (2.8) gives a better upper bound on the size of the 4RC dynamic signal set. If the basis and representation of equation (2.7) had been used and the same assumption of independence had been made, the upper bound would have been: $|S| \le [2(M-1) + 1]^2[4(M-1) + 1]$. In general, the fewer and the smaller in magnitude the number of nonzero positions in $\mathbf{Q}_B$, the

tighter the upper bound. For 4RC, the actual signal set size and the upper bound of equation (2.11) for various M are given in Table 1 (for information regarding the software written for this thesis, see Appendix C). Notice that the actual size of |S| is not much less than $M^4$ for small M, and that the 4RC dynamic signal set is quite large.

Table 1. Size of the 4RC dynamic signal set.

| M | $M^L = M^4$ | Eqn. (2.11) $(2M-1)^3$ | Actual Size |
|---|---|---|---|
| 2 | 16 | 27 | 15 |
| 4 | 256 | 343 | 175 |
| 8 | 4 096 | 3 375 | 1 695 |
| 16 | 65 536 | 29 791 | 14 911 |

### 2.3.2. General Schemes

In the same manner as in the above example, one can obtain the size of the dynamic signal set of arbitrary CPM schemes. The procedure is:

1) Obtain a basis for the space spanned by the chips of the CPM phase pulse. From this form the basis vector $\mathbf{B}(t)$.
2) Find the matrix $\mathbf{Q}_B$, as in equation (2.7), which defines the phase pulse chip vector, $\mathbf{Q}(t)$, via the basis vector $\mathbf{B}(t)$.
3) Find the number of unique $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$ by counting.

Instead of brute force counting of the number of unique $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$, one can assume that every position in $\widetilde{\mathbf{U}}\,\mathbf{Q}_B$ is independent to obtain an upper bound on the number of possible signals. This upper bound can be obtained as follows:

1) Find the sum of the magnitudes of the numbers in each column of $\mathbf{Q}_B$.
2) Multiply each sum by (M-1) and then add 1.
3) Multiply all the resulting numbers together.

The above procedure allows one to easily obtain upper bounds to the number of signals in the signal set of any CPM scheme. For example, consider the class of frequency pulses for which the second half of the pulse is a linearly dependent version of the first half

raised up by some constant frequency. This class includes RC and can be represented by a basis which has a maximum of $\lceil L/2 \rceil + 1$ elements, expressed as in equation (2.8) with two numbers of magnitude one in the first $\lceil L/2 \rceil$ columns and $\lceil L/2 \rceil$ ones in the last column. Thus, an upper bound to $|S|$ is:

$$\begin{aligned} |S| &\le [2(M-1)+1]^{\lceil L/2 \rceil} \{\lceil L/2 \rceil (M-1)+1\} \\ &\le \{2^{\lceil L/2 \rceil} \lceil L/2 \rceil\} M^{\lceil L/2 \rceil + 1} \end{aligned} \tag{2.12}$$

so that the size of the dynamic signal set increases exponentially as $\lceil L/2 \rceil + 1$, and not as L (in general, the exponential increase is simply the number of basis elements in $\mathbf{B}(t)$). As is evident from Table 1, however, these reductions only become significant when $|S|$ is large. Thus, for schemes like RC where the basis actually has $\lceil L/2 \rceil + 1$ elements, the number of matched filters needed for optimum detection, $2|S|$, is prohibitively large even for small values of M and L.

The construction of the signal set vector $\mathbf{Q}(t)$ from the basis vector $\mathbf{B}(t)$ suggests that if one wants to optimally detect a CPM scheme with a reasonable number of matched filters, the number of signals in the basis vector, $\mathbf{B}(t)$, should be kept small. Indeed, for REC CPM the basis vector has one element: $\mathbf{B}(t)=t/2LT$. In other words:

$$\mathbf{Q}(t) = \begin{bmatrix} q_0(t) \\ q_1(t) \\ \vdots \\ q_{L-1}(t) \end{bmatrix} = \mathbf{Q}_B \mathbf{B}(t) = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} [t/2LT] \tag{2.13}$$

In this case the upper bound obtained by the method described above becomes exact since the assumption of independence becomes valid: $|S| = L(M-1)+1$.

A simple generalization of REC is now possible: as long as $|\mathbf{B}|=1$ (where $|\mathbf{B}|$ is the number of basis elements), no matter what $\mathbf{B}(t)$ actually is, one will obtain $|S|=L(M-1)+1$ and therefore require only $2[L(M-1)+1]$ matched filters for optimum detection. This class of phase pulses is called *single basis phase pulses*. It is reasonable to predict that a judicious choice of basis $\mathbf{B}(t)$ other than that of equation (2.13) can yield better spectral and distance properties than those of REC CPM. The design and performance of these schemes is considered in the next chapter.

# CHAPTER 3. SCHEMES WITH A SMALL FILTER BANK

The single basis phase pulses described in Chapter 2 have the advantage of requiring a small number of matched filters for optimum detection. In this chapter, various CPM schemes are considered based on the complexity of the matched filter bank when optimum detection is desired. The schemes are compared in terms of their error performance and spectral properties. In section 3.1, these performance measures are explained. The rest of the chapter is organized in terms of the complexity of the matched filter bank: in section 3.2 single basis phase pulses and in section 3.3 double basis phase pulses are considered. The results show that schemes with a small filter bank for optimum detection do not perform as well as those schemes which have a more complex dynamic signal set. Detection of the more power- and bandwidth-efficient schemes is considered in Chapters 4 and 5.

## 3.1. PERFORMANCE MEASURES

To compare the many CPM schemes considered in this chapter one needs certain performance criteria. The performance measures basic to any modulation scheme are error performance and spectral occupancy, both of which are considered in some detail here. Of course, an equally important criterion is cost of implementation of the CPM system. The many facets of cost, however, make it somewhat difficult to analyze, with engineering issues ranging from circuitry and hardware to choice of carrier frequency and synchronization, all of which are interrelated. One must, therefore, focus on a specific implementation issue which in this chapter is the complexity of the analog portion of the receiver, viz. the matched filter bank. The complexity of the digital portion of the receiver, the detection algorithm, is considered in Chapter 5. (The issue of synchronization is not addressed here, see [1,24,35].)

### Error Performance

The error performance of any modulation scheme over the AWGN channel can be analyzed using the concepts of Euclidean signal space [1,43]. The standard approach for measuring error performance is briefly developed below, following the discussion in [1, Chapter 2].

Assuming that all possible data signals, $s_i(t)$, are equally likely, the maximum

likelihood receiver will choose that signal which is closest to the received noisy signal, r(t), in terms of Euclidean distance. Since the noise, n(t), is a Gaussian stochastic process, the probability of choosing signal $s_j(t)$ when $s_i(t)$ was sent is:

$$P_e\left(s_j(t) \text{ chosen} \mid s_i(t) \text{ sent}\right) = \int_{d_{ij}}^{\infty} \frac{1}{\sqrt{\pi N_o}} \exp\left(-\frac{x^2}{N_o}\right) dx \tag{3.1}$$

where $d_{ij} = \left\{ \int_{\Re} \left[s_j(t) - s_i(t)\right]^2 dt \right\}^{1/2}$ is the Euclidean distance between the two signals, $\Re$ is the region over which $s_i$ and $s_j$ are nonzero and $N_o/2$ is the spectral height of the noise.

The total probability of error is then:

$$\begin{aligned} P_e &= \sum_i P\{s_i(t) \text{ sent}\}\, P\{\text{error} \mid s_i(t) \text{ sent}\} \\ &\le \sum_i P\{s_i(t) \text{ sent}\} \left\{ \sum_{j \ne i} \int_{d_{ij}}^{\infty} \frac{1}{\sqrt{\pi N_o}} \exp\left(-\frac{x^2}{N_o}\right) dx \right\} \\ &= \sum_i \frac{1}{[\text{Number of } s_i(t)]} \sum_{j \ne i} Q\left(\frac{d_{ij}}{\sqrt{2N_o}}\right) \end{aligned} \tag{3.2}$$

where the union bound was used, all signals $s_i(t)$ were assumed equally likely and Q is the familiar Q-function: $Q(x) \equiv \frac{1}{\sqrt{2\pi}} \int_x^{\infty} \exp\left(-y^2/2\right) dy$. At high signal to noise ratios, $P_e$ will be dominated by the term $d_{min} = \min_{i \ne j} d_{ij}$ so that:

$$P_e \approx K\, Q\left(\sqrt{\frac{d_{min}^2}{2N_o}}\right) \tag{3.3}$$

where it is assumed that every signal has K other signals a distance $d_{min}$ away. Experience has shown that Equation (3.3) is a reasonable approximation even for moderate values of signal to noise ratio (SNR) and because of the simplicity of this expression, $d_{min}$ (or $d_{min}^2$) is commonly used as a performance measure. Further, this measure can be applied to sequences to determine the probability of an error event occurring; an error event occurs when the detection algorithm releases a path section not belonging to the transmitted path

and ends when the transmitted path is reentered. The probability of error event is the measure of performance for the simulation results presented in Chapter 5.

Normalizing the energy/bit to 1/2, $d^2_{min}$ for CPM signals is the minima of all (see Equation 1.1):

$$d^2_{\substack{ij \\ i \neq j}} = \int_{\Re} [s_i(t) - s_j(t)]^2 \, dt \approx \frac{\log_2 M}{T} \int_{\Re} \left[1 - \cos\left(\Delta\phi_{ij}(t)\right)\right] dt \qquad (3.4)$$

where the high frequency terms were discarded and $\Delta\phi_{ij}(t) = \phi_{s_i}(t) - \phi_{s_j}(t)$. The calculation of $d^2_{min}$ can be accomplished using a search procedure described in Appendix A of [1]. The results of section 3.2 were obtained in this manner.

### Spectral Properties

The spectral performance of a modulation scheme is measured by the amount of bandwidth the scheme occupies. Since bandwidth is limited, it is important to use up as little of it as possible. There are many different definitions for bandwidth, some of which are listed in [12, p. 104]. A definition used by many (cf. [1,12,39]) is the frequency band in which 99% of the total power resides.

Another important factor to consider is the adjacent channel interference (ACI) caused by a particular modulation scheme. ACI is measured by the average power of the modulation scheme in adjacent channels, so that some indication of ACI is given by the asymptotic decrease in the level of the power spectral density (PSD).

In section 3.3, both the PSD and out-of-band power are considered as measures of spectral performance. Both of these can be evaluated numerically using methods described in [1,21,27]. The autocorrelation method given in [1, Chapter 4] was used here.

### Energy-Bandwidth Comparison

Up to this point, the error and spectral performance have been considered separately. It is easier to compare different modulation schemes by considering these two criteria simultaneously in the energy-bandwidth plane [1]. The advantages and disadvantages of the various schemes discussed in this chapter become more readily apparent using such an approach.

### Detection Complexity

The measure of cost used in this chapter will be the size of the matched filter bank

for optimum detection. Other important measures of detection complexity are the amount and type of hardware required for the detection algorithm (Viterbi Algorithm if optimal detection is desired) and synchronization circuitry, as well as speed (computational complexity). Some of these issues are dealt with in Chapter 4 and Chapter 5..

The next section deals with single basis CPM schemes since these have the simplest signal set and the smallest matched filter bank. It is shown that this simplicity comes at the cost of both error- and spectral-performance.

## 3.2. SINGLE BASIS PHASE PULSES

In the previous chapter it was shown that, for optimum detection, single basis phase pulses require only a linearly increasing number of matched filters with L, the length of the frequency pulse. It was also mentioned that a proper choice of basis could possibly yield better power and spectral properties than the REC scheme. The design of single basis phase pulses is based on the following observations:

1) The REC basis, $\mathbf{B}(t)=t/2LT$, yields a phase response, q(t), which is not smooth at $t=0$ and $t=LT$. This causes the power spectral density (PSD) of this scheme to fall off at a relatively slow rate asymptotically.
2) The distance properties of REC pulses are generally poorer than those of RC at low modulation indices ($h<1/2$). However, one usually needs to use low modulation indices to obtain good power-bandwidth properties.

The first observation suggests that choosing a basis, $\mathbf{B}(t)$, that produces a phase response which is smooth everywhere should yield good spectral properties. For example, one can choose a basis of the RC type (Figure 3.1):

$$\mathbf{B}_{RC}(t) = \frac{t}{2LT} - \frac{1}{4\pi L}\sin\left(\frac{2\pi t}{T}\right) \tag{3.5}$$

This phase pulse has the same degree of smoothness as does the usual RC pulse (although the higher order derivatives are somewhat larger). Thus, one would expect that the PSD of this pulse should decrease asymptotically at the same rate as RC pulses [7]. At the same time, $\mathbf{B}_{RC}(t)$ is very similar to the REC pulse and one would therefore expect the distance properties of these two schemes to be very similar. The distance and bandwidth properties

of the $\mathbf{B}_{RC}(t)$ pulse are discussed in the next section.[1]

The second observation is concerned with error performance. It suggests that if one could shape the REC pulse so that it more closely resembles the RC pulse, perhaps better distance properties could be achieved. The shaping can be easily accomplished by weighting each chip of the phase pulse (this is called partial response FM in [14]). For example, the phase pulse vector for L=4 could become:

$$\mathbf{Q}(t) = \begin{bmatrix} 1 \\ 3 \\ 3 \\ 1 \end{bmatrix} \mathbf{B}(t) \tag{3.6}$$

This type of pulse is a reasonably good fit to the 4RC pulse and should yield similar distance properties (or error performance). Using the results of section 2.3.2, this scheme will require 2[8(M-1)+1] matched filters for optimum detection (note that, from equation (2.12), 4RC requires on the order of $16M^3$ matched filters). Weighted pulses of this type are considered in section 3.2.2.

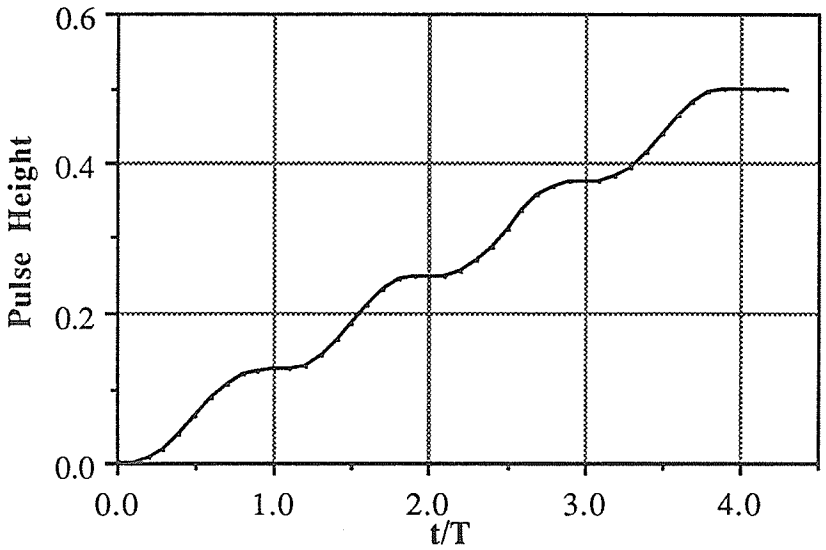


*Figure 3.1. Single basis RC phase pulse.*

---

[1] The properties of this particular phase pulse have been examined by Deshpande and Wittke in [14], who found that the out-of-band power of such a scheme is rather large. Nevertheless, we feel that the discussion of this phase pulse is necessary, especially in light of the findings of Chapter 2.

### 3.2.1. Single Basis RC

The error and spectral performances of the single basis RC pulse (abbreviated SB-RC below) are now considered. As representative cases, results for M=2 and L=3,4 for different observation intervals are plotted in Figures 3.2 to 3.5 (the observation interval, $N_{OBS}$, is defined as the number of symbol intervals the receiver observes the received signal; for example, for symbol-by-symbol detection the observation interval is 1). Recall that the number of matched filters needed for optimum detection of these schemes is 2[L(M-1)+1] as opposed to $2M^L$ for general CPM schemes.

#### Error Performance

From the minimum Euclidean distance plots (Figures 3.2 and 3.3) the following observations can be made:

1) Both Figures: the distance profile of SB-RC follows very closely that of REC, both for L=3 and L=4. This was expected, since these pulses are very similar, especially for larger L.

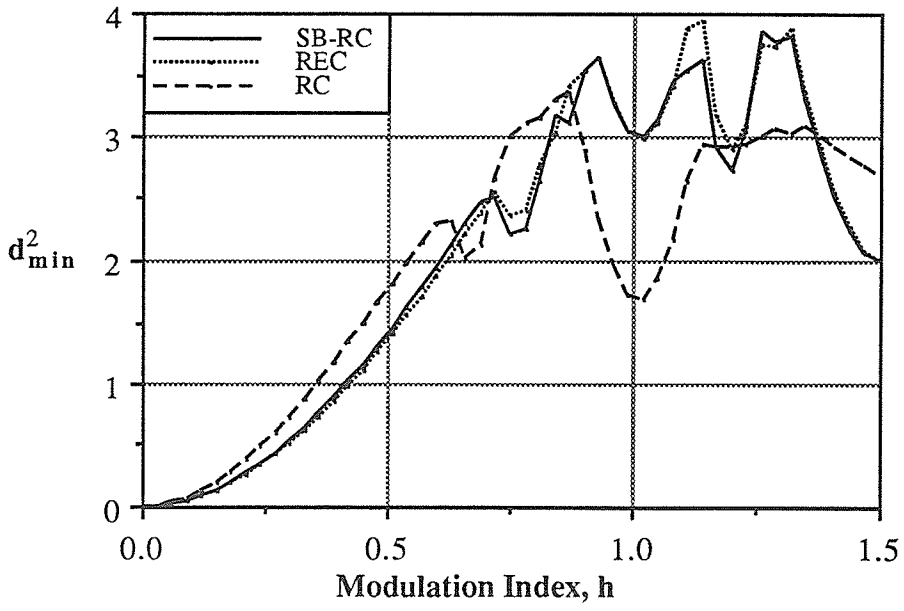


*Figure 3.2. $d^2_{min}$ for M=2, L=3 ($N_{OBS}$ = 8).*

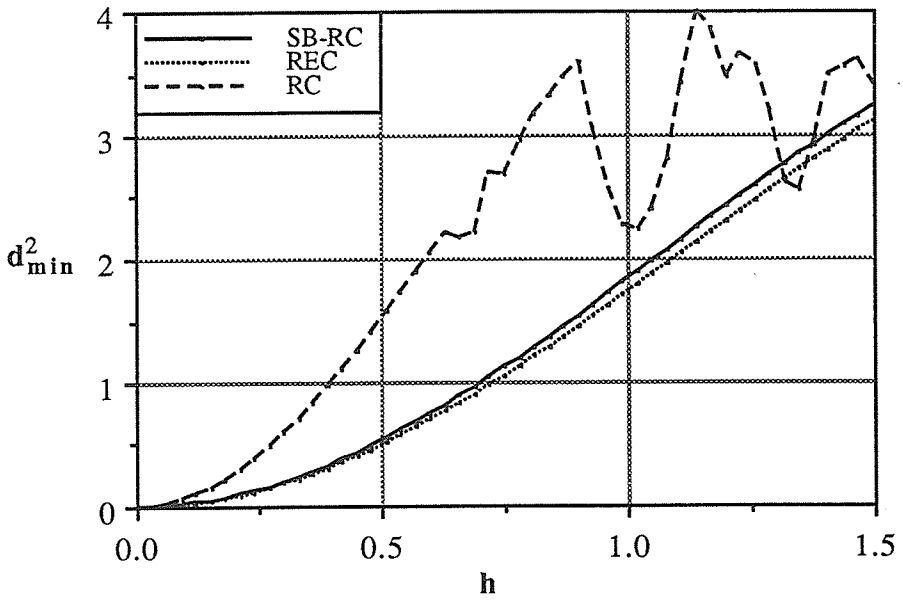


*Figure 3.3.* $d^2_{min}$ *for M=2, L=4 ($N_{OBS}$ = 10).*

2) Figure 3.2: the RC pulse yields better distances at low modulation indices (h = 0 to 0.5). For example, at h = 0.5 the RC pulse is about 0.95 dB more power efficient than either the REC or SB-RC pulses. This suggests that weighting the pulse chips so that the resulting pulse is similar to RC will improve error performance (see section 3.2.2). The price paid for such pulse shaping is an increase in the number of matched filters required for optimum detection, as mentioned above.
3) Figure 3.3: the REC and SB-RC pulses seem to perform much worse than the RC pulse. The reason for this degradation is that there exist specific sequences whose Euclidean distance from the correct sequence does not increase with time. This effect occurs for M=2 when L>3 and for M=$2^n$ when n>1 and L>1.

The effect described in the third observation is not really as bad as it seems. In fact, the distances given in Figure 3.3 come from the difference sequence +2,-2,-2,+2,+2,-2,-2,+2,... or -2,+2,+2,-2,-2,+2,+2,-2,... (a difference sequence is the symbol-by-symbol difference between two data sequences and is used in equation (3.4) to find $d^2_{min}$). Since M=2, these two difference sequences can occur only if the data stream is -1,+1,+1,-1,-1,...

or +1,-1, -1,+1,+1,..., respectively. The probability of either of these two sequences occurring is, of course, vanishingly small. In practice the effect of such sequences depends on the observation interval length, $N_{OBS}$. The longer $N_{OBS}$ is, the less probable an error event of this nature is. Thus, a better indication of the performance of such schemes, at least for small modulation indices, is the distance of those paths which, after diverging from the correct state, meet in the same state after the smallest number of symbol intervals, namely N=L+1 (for larger modulation indices some minimum distance paths can meet some multiple of $2\pi$ away in phase) [1]. Nevertheless, the effect of these sequences is not desirable.

### Spectral Properties

The spectral properties of the SB-RC pulse are compared with the REC and RC schemes in Figures 3.4 and 3.5. From these figures:

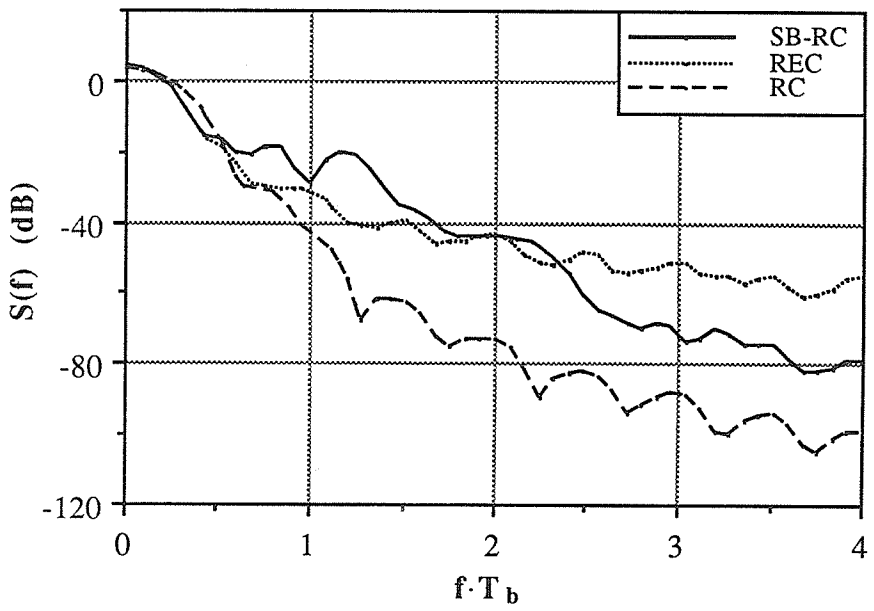


*Figure 3.4. PSD's for M=2, L=3, h=1/2.*

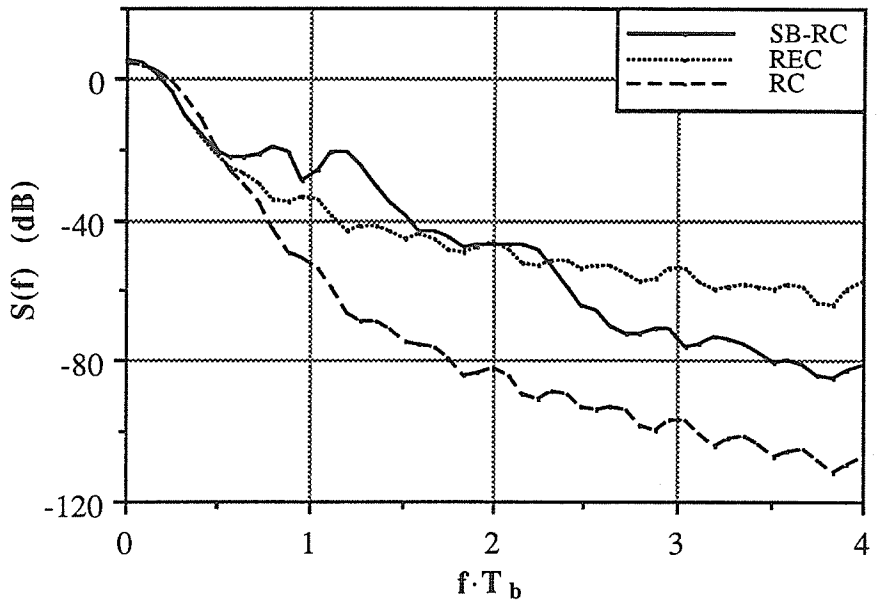


*Figure 3.5. PSD's for M=2, L=4, h=1/2.*

1) As expected, the PSD of the SB-RC pulse decreases at the same asymptotic rate as the RC pulse. This statement, namely that the asymptotic decrease in PSD of the single basis pulse is the same as that pulse from which it was derived, is true for other single basis schemes. At the same time, the SB-RC PSD is larger than the RC PSD at high frequencies, a consequence of the larger higher order derivatives of the SB-RC's phase pulse.
2) The main lobe of SB-RC is much wider than that of either REC or RC. This is caused by both the increase in maximum slope of the SB-RC pulse over REC and the periodicity of the SB-RC pulse (the sum of the pulse chips is not a linear function; see [38]). Again, this carries over to other single basis schemes so that REC has the best main lobe performance of all single basis schemes.

From the PSD's of the SB-RC schemes, one would expect that their out-of-band power performance is quite poor also. This is certainly the case (Figure 3.6). In Figure 3.6, B refers to the one-sided bandwidth containing 99% (and later 99.9%) of the in-band power and $T_b$ is the time per bit. For this particular scheme (M=2, L=3) the bandwidth of SB-RC is much larger than that of REC at 99% in-band power (-20 dB out-of-band power)

and becomes better only at 99.997% in-band power (-45 dB out-of-band power). Therefore, single basis pulses other than REC may only be useful for such channels where a large main lobe is acceptable, but it is important to avoid adjacent channel interference (note that the RC scheme performs better than REC and SB-RC but requires more matched filters for optimum detection). The poor performance of the SB-RC scheme becomes even more apparent in the energy-bandwidth plane.

### Energy-Bandwidth Comparison

The energy-bandwidth characteristics of the L=3 schemes discussed in the above section are compared below (Figure 3.7); the results for L=4 are similar. The minimum distance is normalized by that of MSK ($d^2_{min}$=2) and the bandwidth for 99% in-band power is considered.

The graph shows that the SB-RC pulse performs much worse than either REC or RC in the energy-bandwidth plane at 99% in-band power. This is, of course, due to the main lobe characteristics of the PSD's. The energy-bandwidth properties of REC and RC are similar at 99% in-band power but if more in-band power is considered, RC performs better than REC since the asymptotic spectral performance is better.

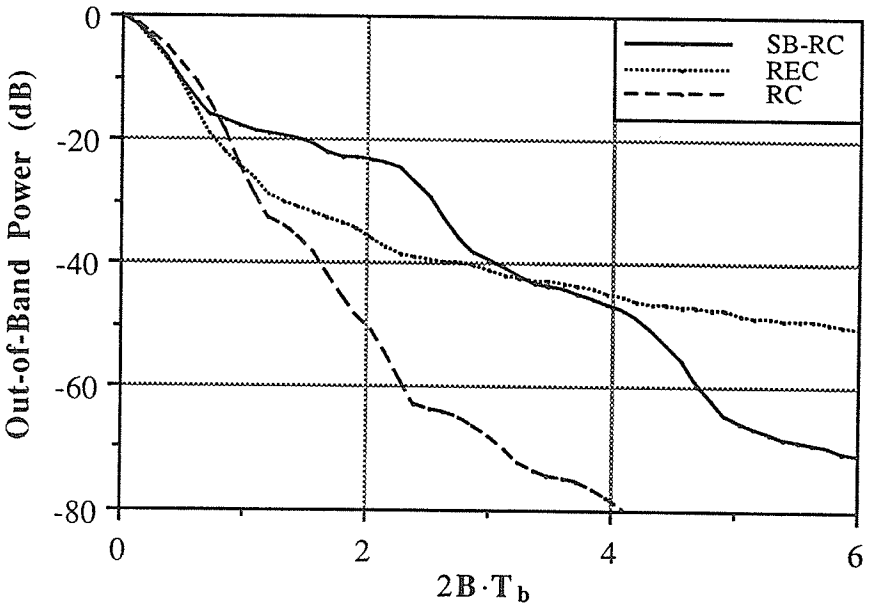


*Figure 3.6. Out-of-band power for M=2, L=3, h=1/2.*

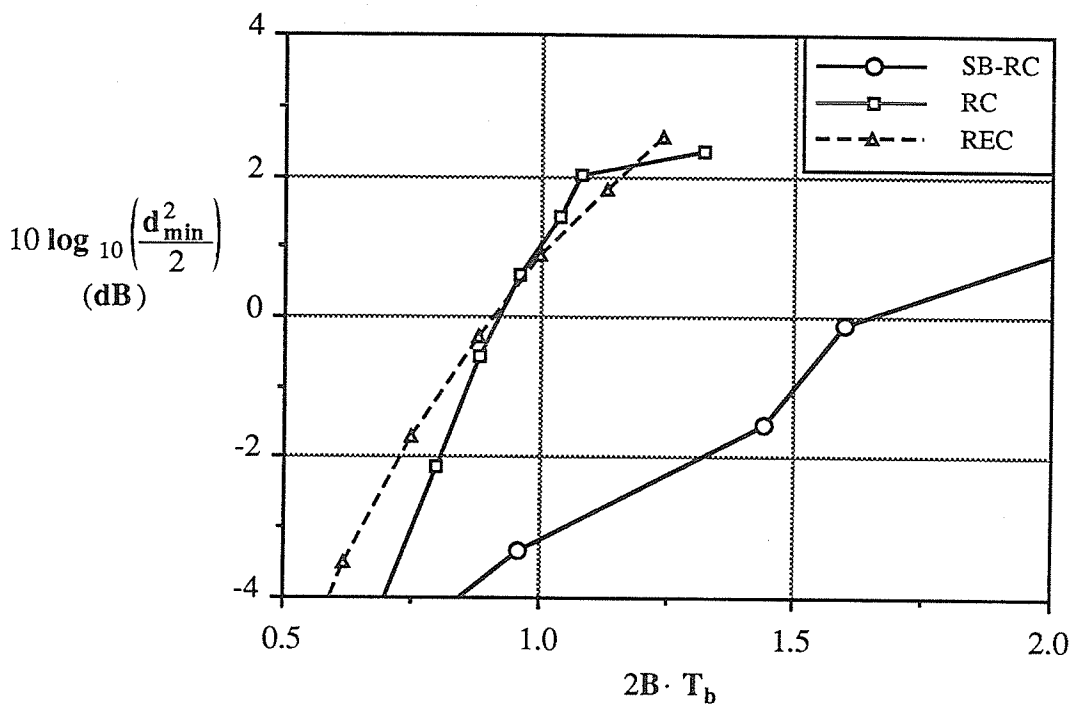


*Figure 3.7. Energy-bandwidth characteristics of SB-RC, L=3 (99% in-band power).*

The poor performance of the single basis pulses somewhat tempers the results of Chapter 2. The reduction in dimensionality of the single basis schemes is offset either by poor asymptotic spectral performance as for REC, or by a much widened main lobe as for SB-RC. This being the case, a natural question to ask is whether allowing a slight increase in the size of the dynamic signal set over single basis pulses (without going to large exponential increases as for RC) allows significant performance improvement. The answer to this question is the focus of section 3.3, but before leaving single basis phase pulses, consider weighted single basis pulses.

### 3.2.2. Weighted Single Basis Pulses

From the above, there are two reasons for considering weighted single basis phase pulses:

1) One would like to shape the phase pulse in the form of an RC pulse to improve error performance at low modulation indices.
2) One would like to prevent distance effects like those of Figure 3.3 from occurring.

Consider weighted pulses of the REC type (called correlative encoded FM with rectangular pulse shaping in [14]). One can shape the REC pulse to closely match the RC pulse shape (Figure 3.8). The chip weights used are (1,3,3,1) as in equation 3.2 and the basis vector must be modified to normalize the final phase pulse height to 1/2:

$$\mathbf{B}(t) = \frac{t}{2\cdot(1+3+3+1)\cdot T} = \frac{t}{16\cdot T} \tag{3.7}$$

The distance properties for this pulse with M=2 are shown in Figure 3.9. As expected, the weighted REC pulse's asymptotic error performance is similar to that of RC. However, there is a further advantage: the effects of Figure 3.3 which occur for unweighted 4REC have been eliminated. As mentioned above, the price paid for such improvements is a slight increase in number of matched filters for optimum detection, namely 2[8(M-1)+1] matched filters rather than 2[4(M-1)+1].

Weighted and unweighted REC schemes have similar spectral properties ([14, Table II, rows 8 and 20] and Figure 3.10) so that one need concentrate solely on error performance and receiver complexity. (This seems at odds with statements made above, but in this case increasing the maximum slope does not greatly affect the PSD; smoothness of the phase pulse is the critical parameter here.)

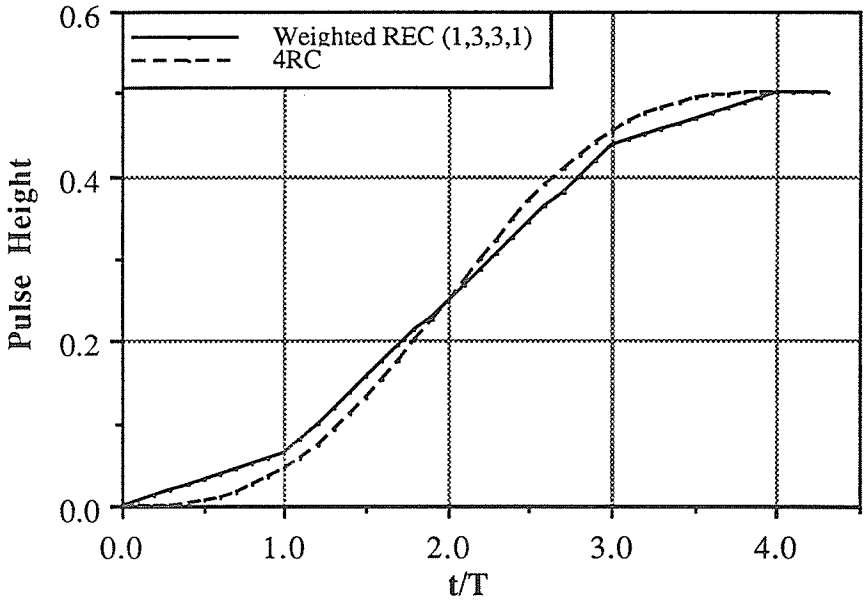


*Figure 3.8. Weighted REC (1,3,3,1) fit to 4RC.*

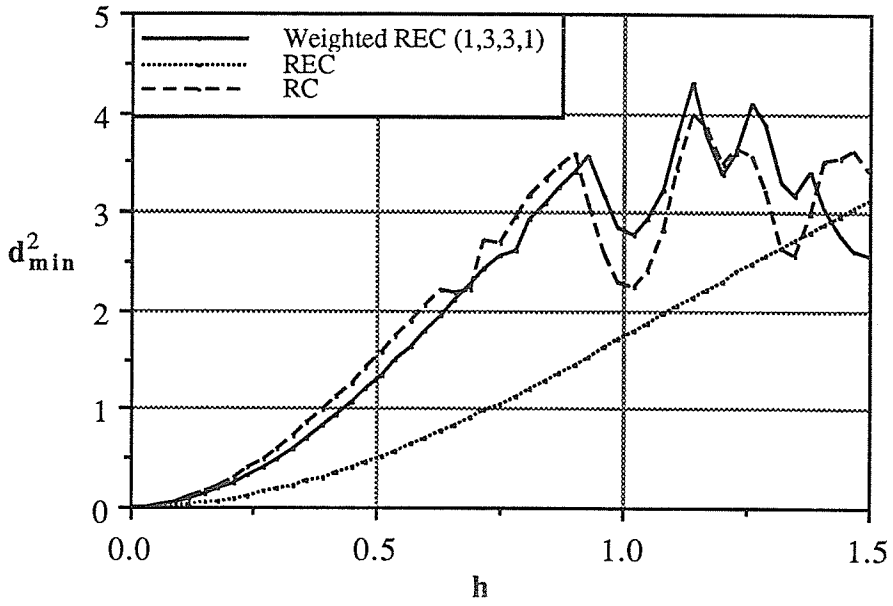


*Figure 3.9.* $d^2_{min}$ *for weighted REC (1,3,3,1) (M = 2,* $N_{OBS}$ *= 10).*

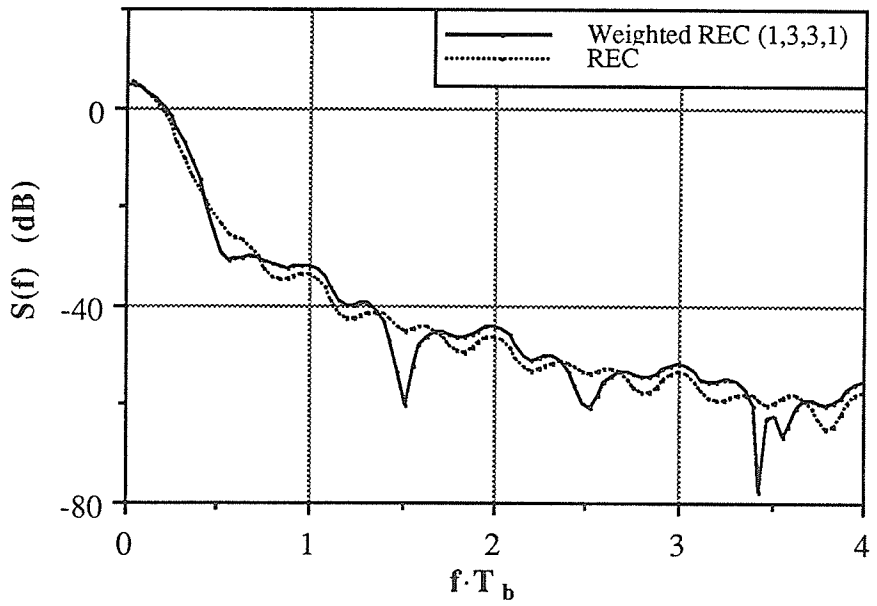


*Figure 3.10. PSD's of REC-type schemes (M = 2, L=4).*

The same type of shaping just described using REC can also be done using other single basis pulses. The results are exactly as one might expect: the error performance of such weighted single basis schemes is similar to that of RC, while the main lobe of the PSD is much widened over that of REC or RC. Therefore, it seems that larger signal sets than single basis must be used for good energy-bandwidth performance.

## 3.3. DOUBLE BASIS PHASE PULSES

The same analysis done for single basis phase pulses is repeated here for schemes which have two element bases (and are thus termed *double basis phase pulses*). The motivation for analyzing these types of pulses is apparent from the results of section 3.2.1: it may be possible to improve spectral performance over single basis pulses significantly without dramatically increasing the number of matched filters for optimum detection. As shown in the sequel, certain double basis schemes do just that. The first double basis pulse examined is a natural extension of the SB-RC pulse (section 3.3.1) whereas the second one is a smoothed REC pulse (section 3.3.2).

### 3.3.1. Double Basis RC

The double basis RC pulse (DB-RC) is limited to even L and has a phase pulse vector of the form (Figure 3.11):

$$Q(t) = Q_B B(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ \vdots & \vdots \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{t}{2LT} - \frac{1}{2\pi L}\sin\left(\frac{\pi t}{T}\right) \\ \frac{t}{2LT} + \frac{1}{2\pi L}\sin\left(\frac{\pi t}{T}\right) \end{bmatrix} \tag{3.8}$$

The rationale for choosing such a pulse is that the same asymptotic spectral performance as RC or SB-RC can be expected without the much widened main lobe, since this scheme does not show the periodicity of the SB-RC pulse (linearity of the sums of pulse chips). Further, the number of matched filters for optimum detection will be $2\left[\frac{L}{2}(M\text{-}1)+1\right]^2$, which increases asymptotically as $(LM)^2$ and not as $M^L$.

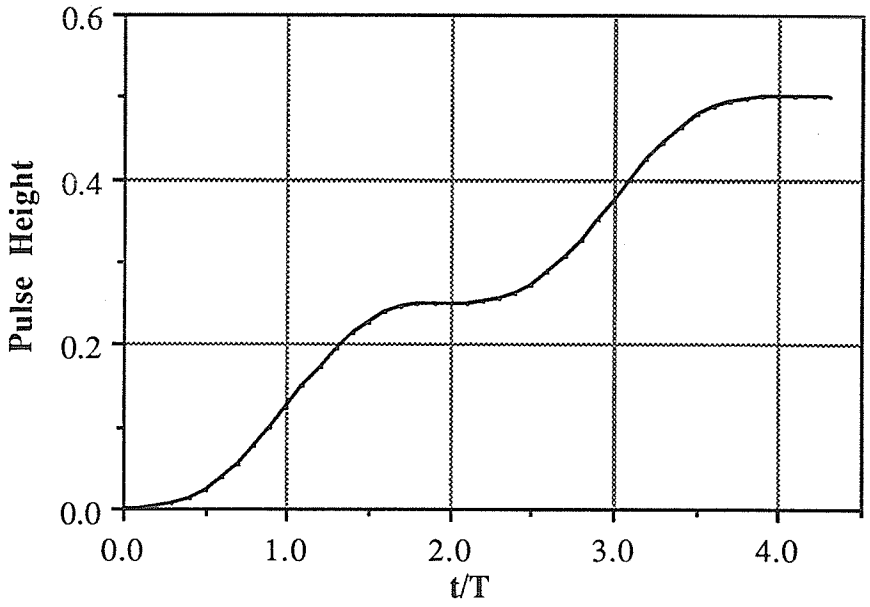


*Figure 3.11. Double basis RC phase pulse.*

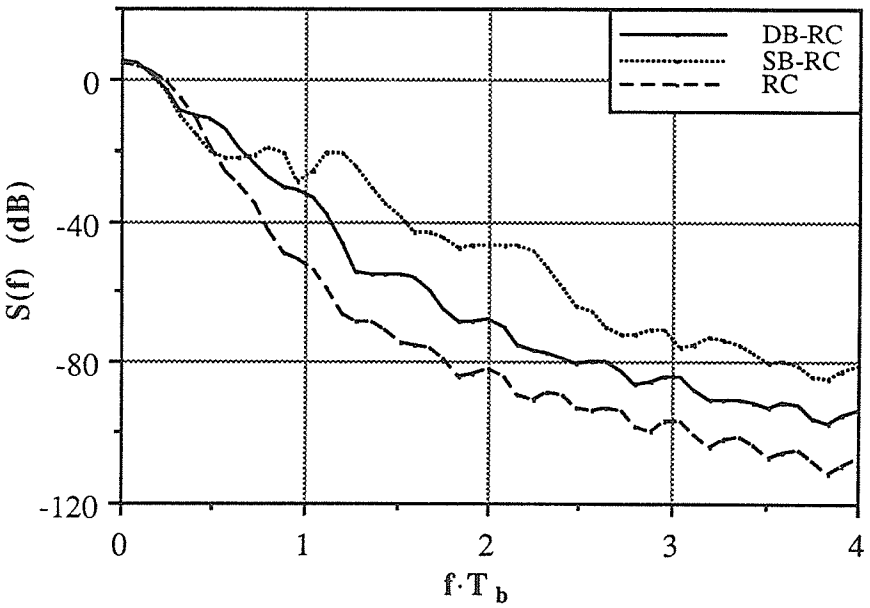


*Figure 3.12. PSD of DB-RC (M=2, L=4, h=1/2).*

As one might expect, the distance properties of DB-RC fall between that of REC and RC and are not shown here. It is the spectral properties of such pulses that are more interesting. The PSD for a DB-RC pulse with M=2 and L=4 is plotted in Figure 3.12, along with the corresponding SB-RC and RC PSD's. It is apparent from the figure that the DB-RC scheme does not improve much over the SB-RC scheme in terms of main lobe performance, and for this reason the energy-bandwidth performance of DB-RC is not much better than SB-RC either. A better choice of basis is possible.

### 3.3.2. Smoothed REC

A good phase pulse, as far as bandwidth performance is concerned, would exhibit a small PSD main lobe and good asymptotic spectral performance. The PSD main lobe is affected by the phase pulse's maximum slope and periodicity (in the manner described in [38]). The REC pulse has the smallest maximum slope of any CPM scheme and therefore also the smallest main lobe. However, due to the discontinuities at the edges of the pulse (Ie. at t = 0 and t = LT), the asymptotic spectral performance is poor. A logical remedy to this is to choose a pulse which is REC except smoothed at the edges. The Smoothed REC pulse considered here is (Figure 3.13):

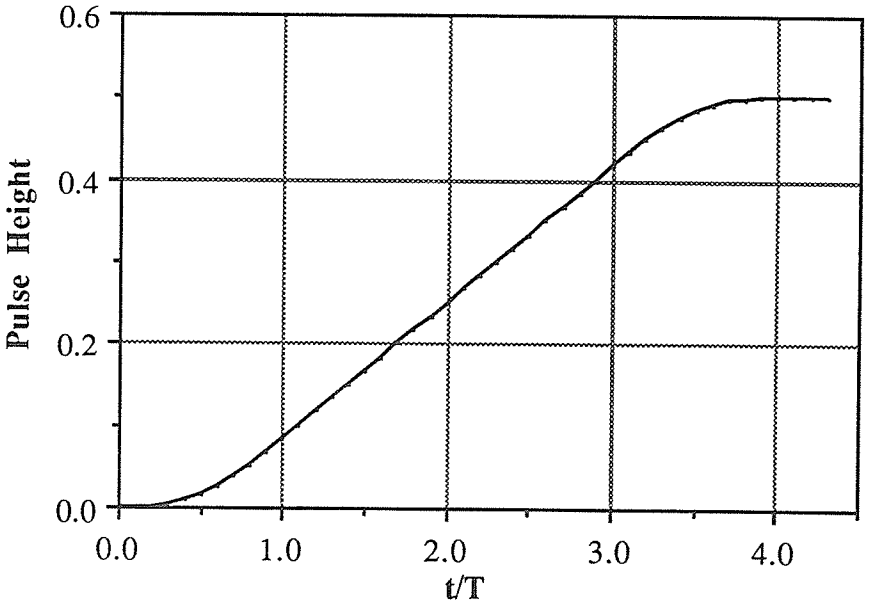


*Figure 3.13. Smoothed REC pulse (L=4).*

$$Q(t) = Q_B B(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ \vdots & \vdots \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \frac{t}{4(L-1)T} - \frac{1}{4\pi(L-1)} \sin\left(\frac{\pi t}{T}\right) \\ \frac{t}{2(L-1)T} \end{bmatrix} \tag{3.9}$$

This scheme will require less than $2[2(M-1)+1][(L-1)(M-1)+1]$ filters for optimum detection. The error performance of this pulse is, just as SB-RC, very similar to that of REC. One notable difference is that, due to the nonlinear terms, the effect of weak sequences is diffused from that of REC.

The PSD of this scheme is shown in Figure 3.14 where the tight main lobe of Smoothed REC is apparent. The asymptotic spectral performance of this pulse is very similar to that of DB-RC. The advantages gained by this pulse over DB-RC become clear in the energy-bandwidth plane for 99% in-band power (Figure 3.15). However, the 99.9% in-band power performance of Smoothed REC is not nearly as good as RC and therefore not much advantage in error or spectral performance is gained by using either double basis pulse presented here.

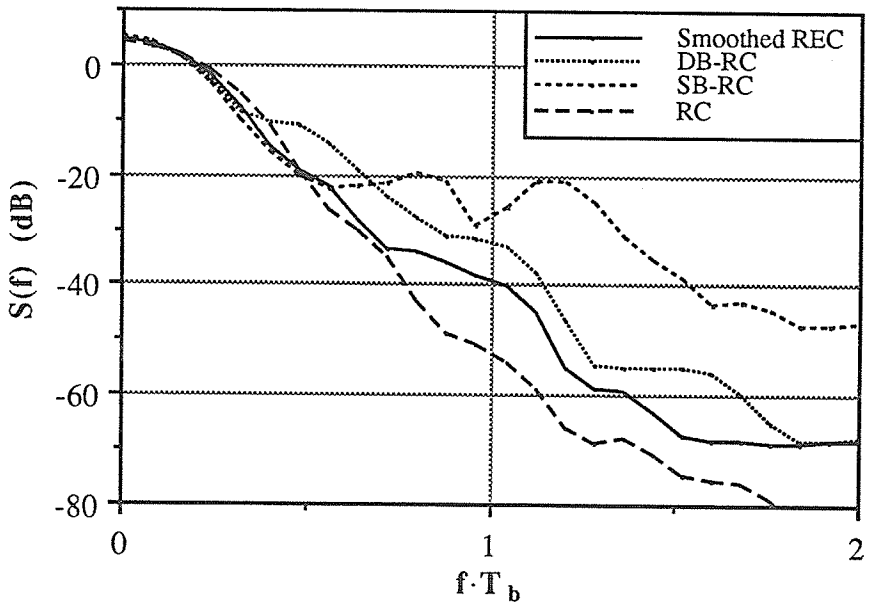


*Figure 3.14. PSD of Smoothed REC (M=2, L=4, h=1/2).*

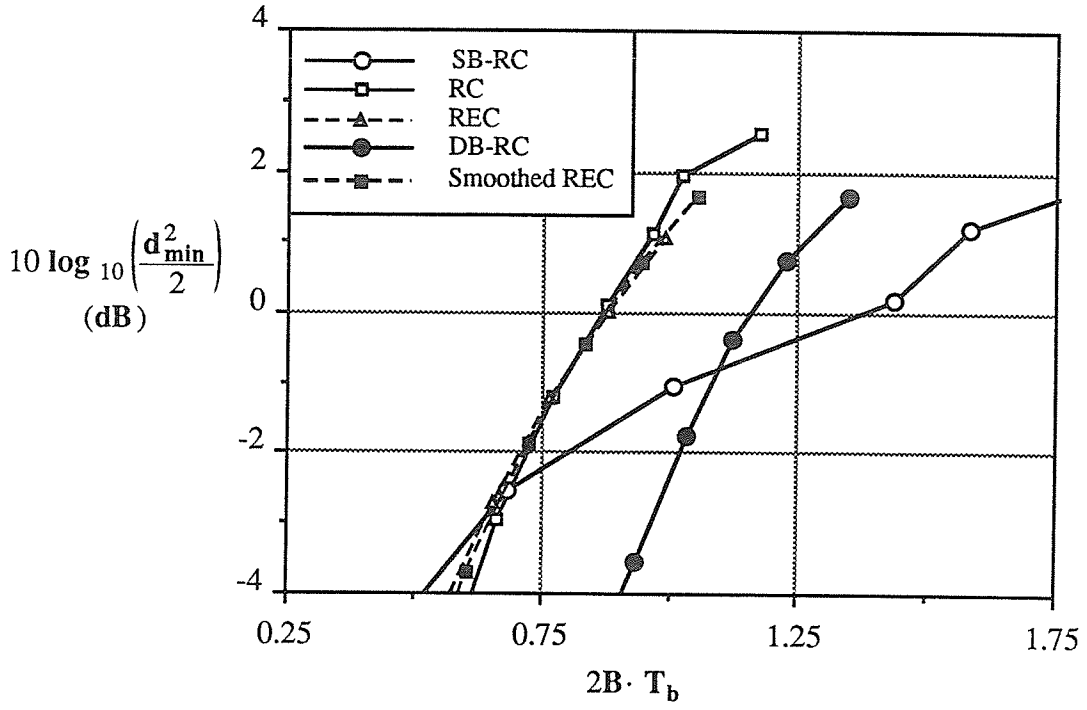


*Figure 3.15. Energy-bandwidth characteristics of DB-RC and Smoothed REC, L=4 (99% in-band power).*

## 3.4. SUMMARY

The discussion above has focussed on schemes which require a comparatively small number of matched filters for optimum detection. The single basis schemes require only a linearly increasing number of matched filters with L and M but yield poor bandwidth performance. The double basis phase pulses perform somewhat better but require more matched filters. None of these schemes can match the error and spectral performance of the RC phase pulse and it seems that a direct trade-off between the dimensionality of the signal set and performance exists.

The tacit assumption in this chapter was that the receiver must perform an optimal detection of the CPM signal. An important question which arises at this point is whether optimum detection is necessary. Perhaps suboptimum detection suffices for most situations. This question is addressed in the next chapter.

# CHAPTER 4. SUBOPTIMUM DETECTION: THE FILTER BANK

The prohibitive size of the matched filter bank for optimum detection of CPM has led to the development of several suboptimum schemes which sacrifice error performance for reduced receiver complexity [1,23]. The need for reducing the complexity of the filter bank was disputed by the results of Chapter 2, where it was shown that single basis CPM pulses require many fewer matched filters than general schemes. However, the bandwidth performance of single basis schemes is disappointing; either the asymptotic spectral performance is poor (REC) or the main lobe is excessively wide (SB-RC). One would therefore want to send information via multiple basis pulses like RC, but now one must return to suboptimum receivers. The goal is to keep the filter bank as small as possible without sacrificing error performance.

This chapter is organized as follows. In section 4.1, a suboptimal technique based on single basis pulses is proposed. It is shown that little degradation in error performance is possible even when few filters are used. This scheme, however, does not perform as well as a recent approach proposed by Huber and Liu in [23] and reviewed in section 4.2. In fact, based on simple arguments, the approach of [23] is likely the best way of reducing the size of the filter bank. Some of the ideas of the first section are therefore applied to Huber and Liu's approach.

## 4.1. PIECEWISE LINEAR DETECTION

The principles of the Piecewise Linear Detector (PLD) are illustrated via Figure 4.1. The multiple basis chips (in this case RC) are approximated by linear chips so that the approximating pulse is piecewise linear and therefore single basis. Such a pulse requires only a linearly increasing number of matched filters with the length of the frequency pulse. The principles of this receiver are similar to that of the Reduced Complexity Viterbi Detector described in [1, section 8.1]. The difference is that no state reduction is attempted here, only reduction in the size of the filter bank (note that, if the Reduced Complexity Viterbi Detector in [1] is based on REC pulses, the size of its filter bank is further reduced so that the complexity reduction factor for the receiver filters on p. 281 of [1] may be as much as $\frac{M^{L_T}}{[L(M-1)+1]}$ and not just $M^{(L_T - L_R)}$; the actual reduction depends on the choice of slopes of the REC scheme, as explained below).

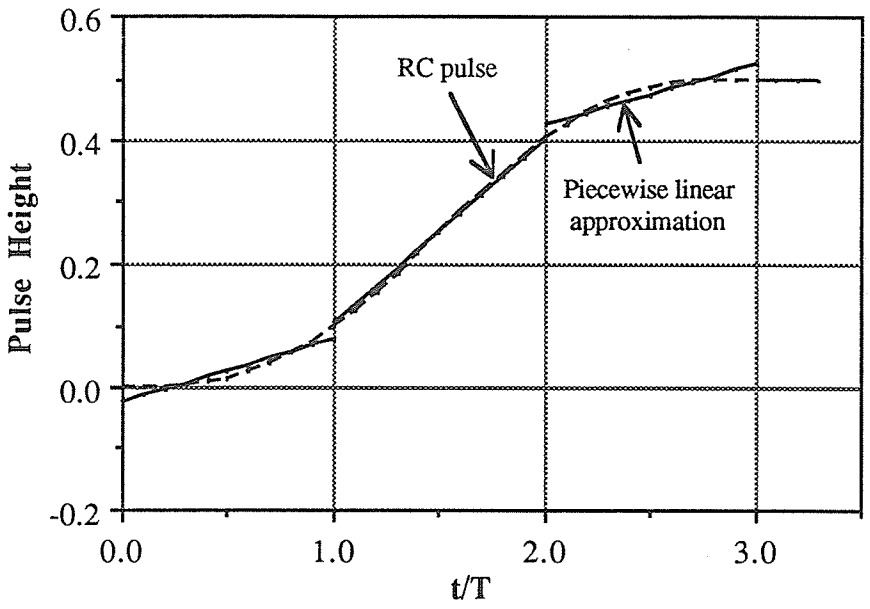


*Figure 4.1. The Piecewise Linear Detector.*

It remains to choose the slopes and initial phases of the piecewise linear sections. A good criterion for doing this is minimization of the loss in $d^2_{min}$, but this is rather difficult [1, Chapter 8]. A simpler approach is to choose the chips in such a way that, given the same symbols transmitted for both the smooth and piecewise linear schemes, the distance between the smooth and PLD signals is minimized. Ie. Minimize (equation (3.4)):

$$\begin{aligned} d^2 &= \frac{\log_2 M}{T} \int_0^{(N-1)T} \left[1 - \cos\left(\phi_T(t,\alpha) - \phi_R(t,\alpha)\right)\right] dt \\ &= \frac{\log_2 M}{T} \int_0^{(N-1)T} \left\{1 - \cos\left(2\pi h \sum_{i=0}^{\lceil t/T \rceil} \alpha_i \left[q_T(t-iT) - q_R(t-iT)\right]\right)\right\} dt \end{aligned} \tag{4.1}$$

where $q_T(t)$ is the transmitter phase pulse (usually RC) and $q_R(t)$ is the piecewise linear phase pulse, assumed to be at the receiver. Equivalently, one could minimize the distance between the two signals in any given symbol interval. If the term in the cosine of equation 4.1 is small, which will occur if $q_R(t)$ is properly chosen, then the goal becomes minimization

of $\left(\text{using } \cos x \approx 1 - \frac{x^2}{2}\right)$:

$$d^2 \approx \frac{\log_2 M}{T} \int_{nT}^{(n+1)T} 1 - \left\{1 - \frac{\left[\phi_T(t,\alpha) - \phi_R(t,\alpha)\right]^2}{2}\right\} dt$$

$$= \frac{\log_2 M}{2T} \int_{nT}^{(n+1)T} \left[2\pi h \sum_{i=0}^{n} \alpha_i \Delta q_{TR}(t-iT)\right]^2 dt \qquad (4.2)$$

The minimization should take place over all particular transmitted sequences. However, this is not readily possible, and a simpler approach is to take an average over all possible sequences. In other words, minimize:

$$\frac{1}{M^n} \sum_{\alpha} \int_{nT}^{(n+1)T} (2\pi h)^2 \left[\sum_{i=0}^{n} \alpha_i^2 \left(\Delta q_{TR}(t-iT)\right)^2 + 2 \sum_{\substack{i=0 \\ i \neq j}}^{n} \sum_{j=0}^{n} \alpha_i \alpha_j \Delta q_{TR}(t-iT)\, \Delta q_{TR}(t-jT)\right] dt \qquad (4.3)$$

The second term in equation (4.3) will disappear since the $\alpha_i$ are symmetric about 0, leaving only a minimization of the sum of $\left(\Delta q_{TR}(t-iT)\right)^2$ over multiple symbol intervals. This is the same as minimizing the mean square error between $q_T(t)$ and $q_R(t)$ for any given symbol interval. Thus, the final conclusion is that a good way, although not the best way, of choosing the piecewise linear sections is to minimize the mean square error between the transmitter and receiver phase pulses.

Another factor to keep in mind when choosing slopes for the PLD is that the number of filters required depends directly on the number of slopes (or frequencies) the receiver scheme can take on. The slopes should be chosen as integer multiples of each other using integers as small as possible, so that the size of the dynamic signal set is kept small. In fact, it may be prudent to choose some slopes to be zero, so that fewer filters are needed. For example, consider again Figure 4.1. The pulse vector is given by:

$$\mathbf{Q}(t) = \begin{bmatrix} 1 \\ 3 \\ 1 \end{bmatrix} \mathbf{B}(t) \qquad (4.4)$$

and requires 2[5(M-1)+1] filters for detection (optimal detection if this was the scheme used at the transmitter). Instead, one could choose the PLD of Figure 4.2, which has a phase pulse vector:

$$\mathbf{Q}(t) = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \mathbf{B}(t) \tag{4.5}$$

and therefore requires only 2M filters for detection. Of course, this second scheme will not perform as well as the first.

The minimum squared Euclidean distance for the PLD can be calculated in the manner described in [1, sections 8.1.2. and 8.1.3.]. The results are presented below (Figures 4.3 and 4.4). As shown in Figure 4.4, using the PLD of Figure 4.1 limits the loss in $d^2_{min}$ to 0.1 dB for small modulation indices; the loss tends to increase for larger h. Based on the figures, the PLD detector seems to perform reasonably well and when compared to the schemes of [1, Chapter 8], the results are encouraging. However, in comparison with more recent schemes, the results pale somewhat.

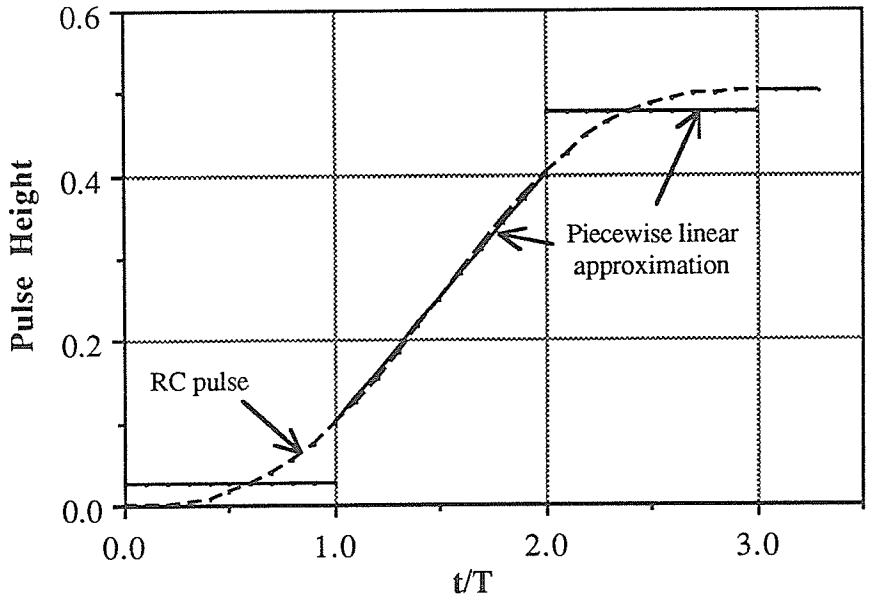


*Figure 4.2. A simpler PLD.*

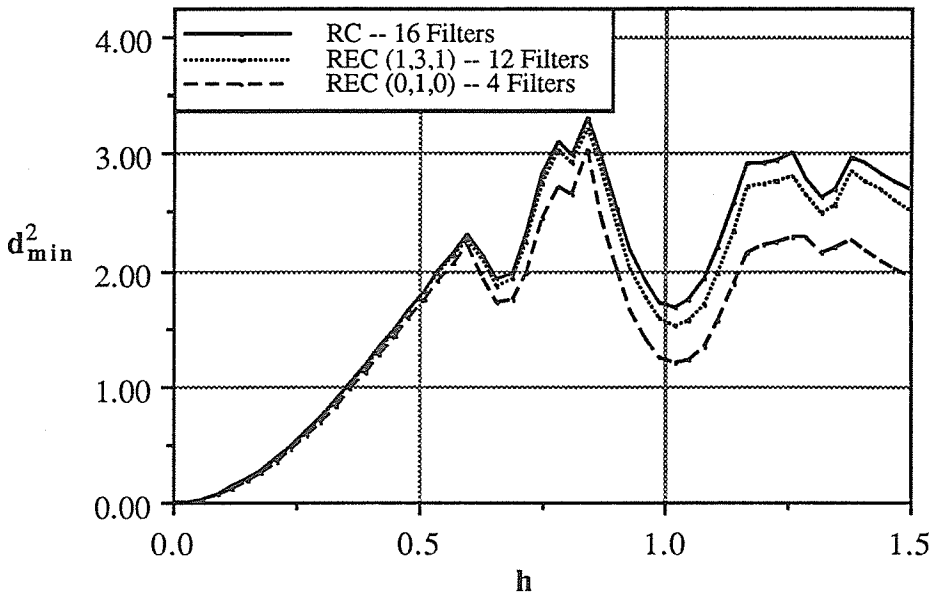


*Figure 4.3. $d^2_{min}$ of the PLD's for binary 3RC ($N_{OBS}$=6).*

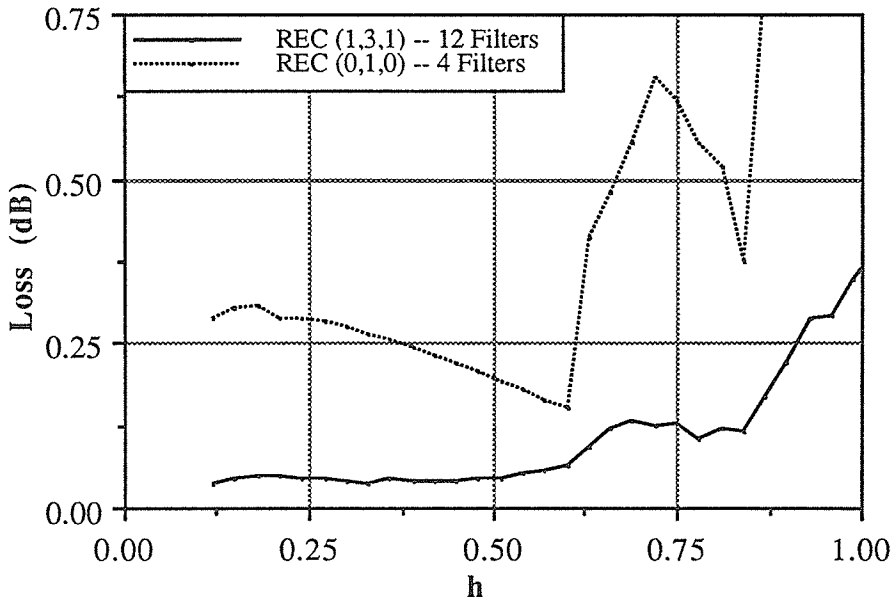


*Figure 4.4. Loss in $d^2_{min}$ due to using PLD's (M = 2, 3RC, $N_{OBS}$ = 6).*

## 4.2. H+L DETECTION

In [23], Huber and Liu proposed two detectors based on signal space approaches. The first approach, called the Gram-Schmidt Approach here, involves a Gram-Schmidt orthogonalization on all of the signals in the signal set. Huber and Liu argue that only those four to six filters which correspond to the first four to six orthogonal dimensions of the Gram-Schmidt procedure are actually needed to capture most of the energy of any signal in the signal set. Their arguments are based on results from Landau and Pollak [28] and the performance of this approach is good, but they subsequently proposed a more practical receiver.

The second approach, here termed H+L Detection, is based on practical considerations, viz. a filter bank which is easy to build. The idea is to simply project every signal in the CPM signal set onto two (or three) frequencies, namely $f_c \pm \Delta f$ (or $f_c$ and $f_c \pm \Delta f$), resulting in four (or six) baseband filters (four filters per frequency, with symmetry reducing this number by 1/2). This approach could also be generalized to more frequencies with a corresponding increase in the number of filters.

H+L Detection with six filters yields excellent results. For most schemes studied in [23], the loss in $d^2_{min}$ is limited to less than 0.1 dB, even with dynamic signal set sizes of 64. As a comparison, results for quaternary 3RC (M=4, L=3) are shown in Figure 4.5. The H+L Detector outperforms the Piecewise Linear Detectors, even with a smaller number of filters.

The results of Figure 4.5 are to be expected. Given any number of filters, the best representation of any signal in the signal set is its Euclidean projection (at least for the AWGN channel) onto the space defined by those filters. Thus, the approach of Huber and Liu is the method of choice for building a practical receiver for CPM signal sets.

### 4.2.1. Multiple Receiver Chips / Transmitted Chip

Based on the results, it seems that the discussion about the Piecewise Linear Detector is somewhat superfluous. However, these ideas point to further improvements of the simple H+L detector of the previous section. For example, a simple extension of the PLD is to use more than one receiver pulse chip per transmitter pulse chip, as shown in Figure 4.6.

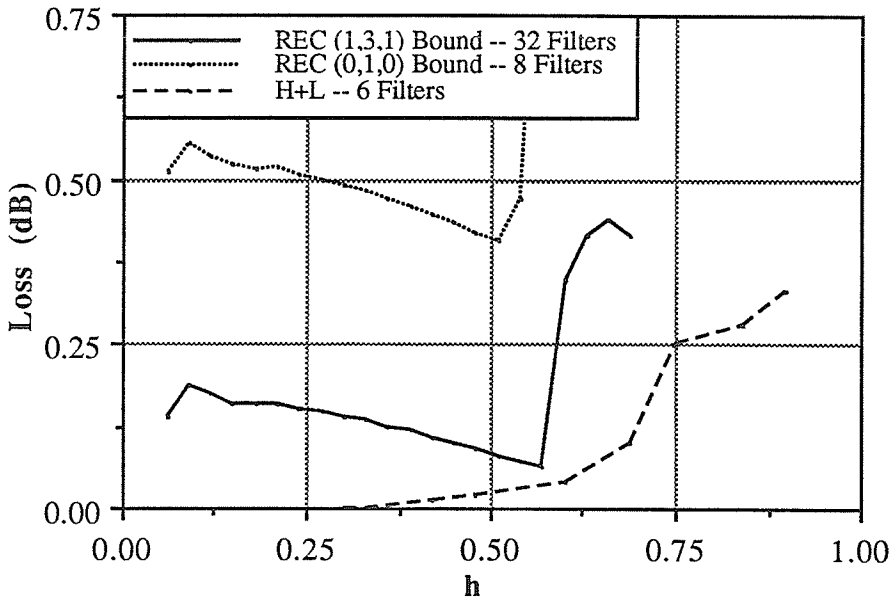


*Figure 4.5. PLD's vs. Huber and Liu's filtering scheme (M=4, 3RC).*

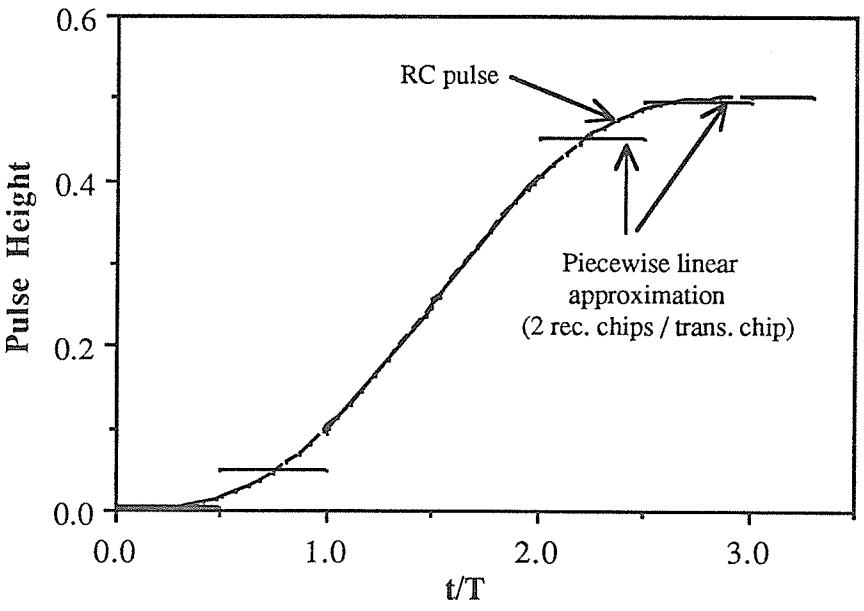


*Figure 4.6. Multiple receiver chips / transmitter chip.*

This approach will improve the receiver performance at the expense of requiring faster circuitry. For example, in Figure 4.4 the loss in error performance at h=1/2 for the (0,1,0) scheme is 0.17 dB while for the scheme of Figure 4.6 it is 0.07 dB. Thus, for this particular case, an improvement of 0.1 dB is gained by using 2 receiver chips for each transmitter chip. Both schemes require 4 filters but the second scheme must sample each filter output twice every symbol interval rather than just once.

Instead of applying this idea to the PLD, we know from the results of Huber and Liu that projecting the partial-chips (in Figure 4.6 they are half-chips) onto the space generated by the filters will better represent the partial-chip signal set than the PLD receiver can. This approach will improve the performance of the H+L scheme and it may in fact be more practical to use multiple receiver chips / transmitter chip than increasing the number of filters.

Distance calculations for the multiple receiver chips / transmitter chip method combined with Huber and Liu's approach have not been done here. This is left for future work.

# CHAPTER 5. SEQUENTIAL SEQUENCE ESTIMATION

The previous three chapters concentrated on the filter bank which supplies the detection algorithm with statistics. This chapter deals with the detection algorithm itself which processes these statistics so as to minimize the probability of an error occurring. A Sequential Algorithm is considered for the detection of CPM modulated data sequences. The motivation for this is that the most power- and bandwidth-efficient CPM schemes have a complex state representation and optimum detection with the Viterbi Algorithm is expensive. The size of the state representation becomes even larger if factors such as bandlimiting, fading and coding are added to the channel model. A reduction in the computational complexity of the detection algorithm is therefore of great interest.

A large assortment of techniques for reducing the computational complexity of the detection algorithm have been proposed. Some of the earlier schemes, from Sequential Algorithms to the M-Algorithm, are reviewed in [2,4] while the more recently developed schemes such as Reduced State Sequence Estimation (RSSE) are listed in [2, Chapter 6]. Several of these techniques have already been applied to CPM detection/decoding [1,23,40]. The results have been both encouraging and disappointing. For example, in Simmons and Wittke [40], a limited search algorithm is used to detect CPFSK signals by pursuing only two paths through the trellis and this simple approach performs essentially as well as the Viterbi Algorithm at moderate and high SNR's. However, when k in the modulation index h=k/p is allowed to become larger than unity, catastrophic effects occur, viz. error events tend to become extremely long because the detection algorithm plunges into a false lock [40]. This false-lock phenomenon necessitates extra processing which slows down detection. Another example, also from a study by Simmons [41], is that although the M-Algorithm and RSSE perform well for the AWGN channel, performance is seriously degraded over a more realistic channel with adjacent channel interference (ACI). Given these problems, it seems prudent to try other computationally efficient algorithms for CPM and test their response to different channel impairments.

The algorithm applied to CPM in this chapter is the Sequential Stack Algorithm. A sequential algorithm is chosen because of its wide range of applicability -- it has been extensively studied and successfully applied for the decoding of convolutional codes [18,31,44] and more recently it has been shown to yield excellent results when applied to channels with severe impairments [13,36,45]. Sequential algorithms can also be applied to the decoding of block codes [42]. Of the many sequential algorithms that exist, the one

used here is the simplest, the Stack Algorithm. Apart from its simplicity, another reason for choosing the Stack Algorithm is that there exists a simple architecture for its implementation, called the Systolic Array Architecture [10]. This architecture further reduces the computational complexity of the Stack Algorithm by only partially sorting the stack of paths. This is discussed in more detail in the sequel.

This chapter is organized as follows. In section 5.1 the Stack Algorithm and Systolic Array Architecture are described. The advantages and pitfalls inherent to sequential sequence estimation/detection/decoding are also discussed here. Finally, the performance of the algorithm, as tested via software implementations, is presented in section 5.2 (see also Appendix C). Measures of performance include error probability, computational complexity and sensitivity of performance to inaccurate channel parameter estimation and to channel impairments.

## 5.1 THE STACK ALGORITHM

The Stack Algorithm (hereafter called SA) falls into the class of backtracking and metric-first algorithms [2]. In other words, when searching the CPM tree/trellis, the algorithm is allowed to move both backwards and forwards and the determining factors for movement are the metrics of the paths under consideration. The motivation for choosing the SA over the Viterbi Algorithm is that the more power- and bandwidth-efficient CPM schemes have a large number of states, making Viterbi detection infeasible. The SA can handle many states, so that the effects of coding, fading or even irrational modulation indices (requiring a tree graph) can be included in the channel model. It is desirable to include such effects so that good error performance can be achieved.

The motivation for choosing the Stack Algorithm over other sequential algorithms, such as the more popular Fano algorithm, is threefold. First, the SA is faster than the Fano algorithm at communication rates approaching the computational cutoff rate, $R_o$ [11]. Secondly, the principal disadvantage of the SA relative to the Fano algorithm is the memory required by the stack, but since memory has become a much less expensive commodity and because speed is our primary goal, this factor is not as important as it once was. Finally, the problem of reordering of the stack, also considered a debilitating factor of the SA, is solved by sorting with parallel architectures.

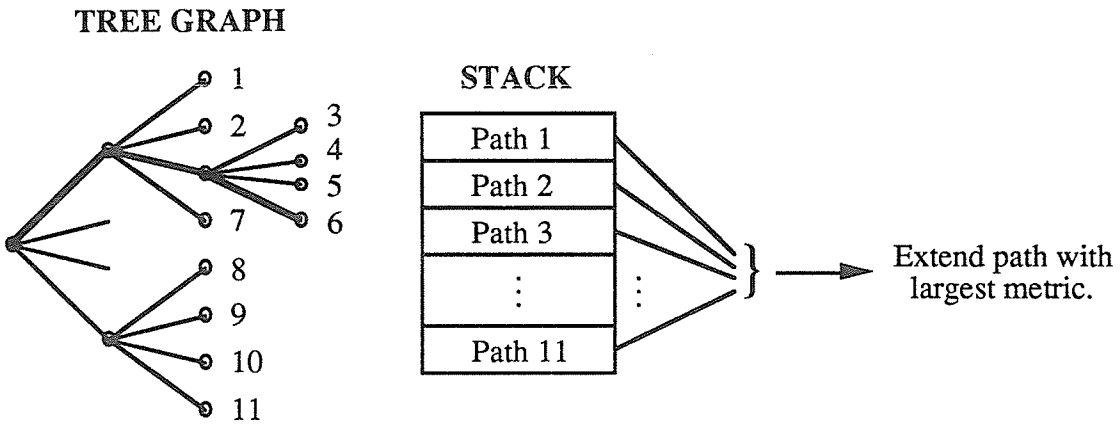


*Figure 5.1. The Stack Algorithm.*

The basic SA can be described by just one step: extend the best path in the stack one step further into the CPM tree/trellis, where "best" refers to the path with the largest metric (or smallest, depending on the definition of the metric used) (Figure 5.1). The metric generally used is the Fano Metric, suggested by Fano in [15] and shown by Massey [32] to be the metric which identifies the path most likely followed by the transmitter given the information available. Of course, to find the path with the largest metric, some type of sorting must be performed and it is the sorting which slows down the SA. Thus, care must be taken in choosing a sorting procedure; indeed, many different procedures have been suggested. Eg. Merge and bucket sorting. However, the assumption here is that the sorting must be done sequentially rather than in parallel. Recently, a parallel sorting approach for the SA has been suggested which speeds up the sorting procedure at the expense of hardware. This is discussed next.

### Systolic Array Architecture

A hardware implementation of the SA which performs sorting in parallel has been given in [10] (see also [2] where other parallel architectures are described). In this implementation, the paths under consideration are stored in a systolic priority queue (Figure 5.2), and are reordered in a fixed and short period of time without completely reordering the stack. The operation of the systolic priority queue is as follows:

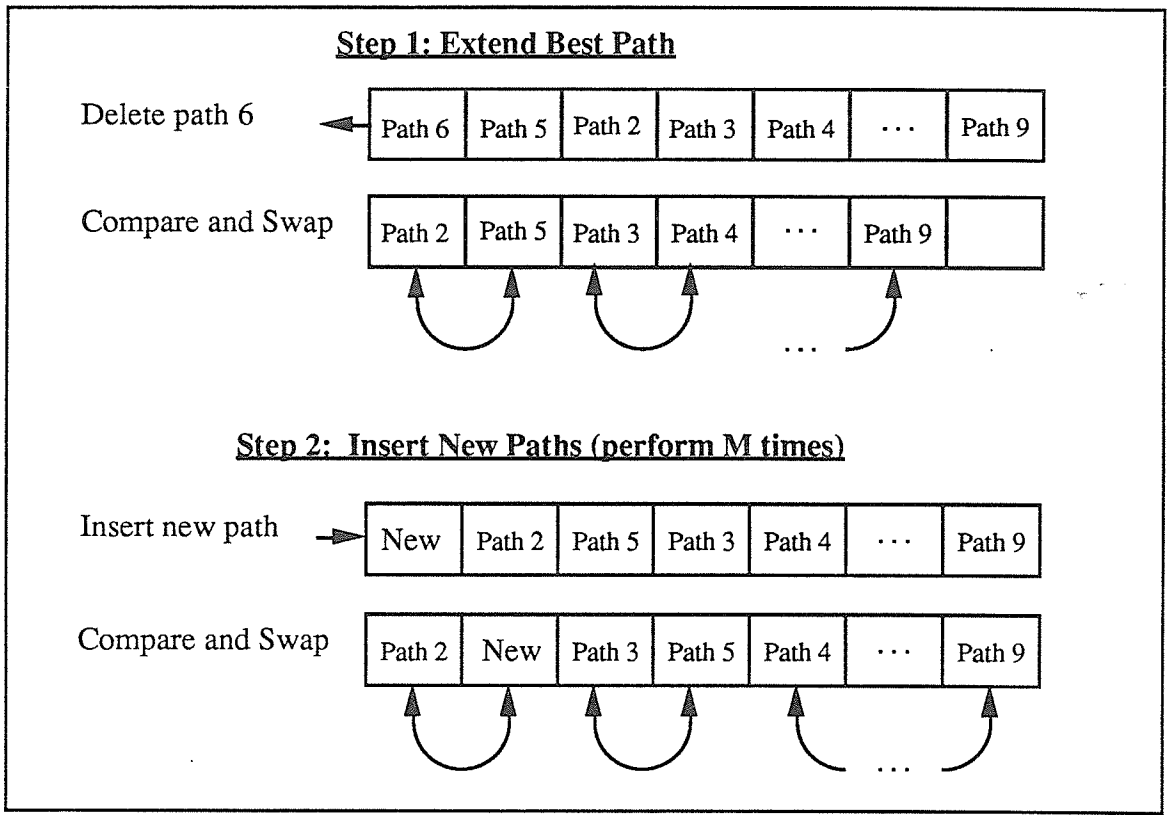


*Figure 5.2. The Systolic Priority Queue*

1) The top path (the one with the best metric) is deleted and the adjacent pairs of paths are ordered by swapping. This swapping is performed in parallel and only a comparison and path exchange (if necessary) takes place.
2) The deleted path is extended to its M new paths. These paths are fed into the queue sequentially. After every insertion into the queue, adjacent pairs of paths are again compared and reordered.

Chang proves in [10] that by using the above procedure, the path remaining in the top stack entry will always be the path with the best metric. Since the SA only requires that the best path be known, this approach is much more efficient than a complete sort (note that many comparisons are performed for the extension of a single path, but the parallel processing of this architecture alleviates this problem). Moreover, this architecture is easily implemented in hardware.

The problem of sorting for the SA is therefore not as bad as it may otherwise seem. However, the other pitfalls inherent to sequential algorithms remain.

## Pitfalls

In addition to the sorting issue specific to the stack algorithm, a problem inherent to all sequential algorithms is that the number of path extensions required to move one level further in the tree/trellis is a random variable and not a constant as for breadth-first searches such as the Viterbi Algorithm. If the noise local to a few symbol intervals is severe then the sequential algorithm is forced to take up much time searching for the best path. This can occur, for example, in a bursty channel; or it could happen in the AWGN channel if the sample noise function (from the ensemble of noise functions) is large over several symbol intervals.

The result of such variability in computations is that one will need to buffer the incoming data while the detection algorithm is searching, and if the delay becomes too long, stack overflow occurs. In fact, beyond a certain SNR limit given in terms of the computational cutoff rate, $R_o$, the number of computations increases rapidly with the severity of the noise [11]. To overcome these effects, one possible approach is to force a decision before the algorithm has finished its search; of course, this will increase the error probability. Moreover, such early decision techniques are plagued by the type of problems described earlier for limited search algorithms (Eg. the false lock problem).

An alternative solution is possible if the communication system has automatic repeat request (ARQ) capability. In this case, the problem of stack overflow can be dealt with by sending packets of information and allowing erasures. Such issues and further practical problems related to sequential algorithms have been analyzed in great detail in many texts and papers (see [11, Chapter 7] and [2, Chapter 6]). Here, only two of these issues are investigated.

The first problem considered is that, for the AWGN channel, the receiver must know the noise strength, $N_o$, to calculate the Fano Metric. This problem arises because paths of different lengths are being considered. Thus, measurement of the channel noise is required for the SA, and if this measurement is inaccurate, performance is degraded. The sensitivity of the SA's performance to incorrect estimation of $N_o$ is examined in the simulations.

The second issue investigated is the sensitivity of performance to adjacent channel interference. The impetus for this is that a recent study showed that the M-Algorithm and RSSE perform much worse than the Viterbi Algorithm in high levels of ACI [41]. The explanation given for this was that the early decision mechanism of these algorithms was to blame. It seems possible that the SA will overcome this problem, as it does not make the same kind of rash decisions as RSSE or the M-Algorithm.

## 5.2. PERFORMANCE

Performance of the SA for the detection of CPM modulated sequences is measured via error performance and computational complexity. In addition, the robustness of the SA to poor estimation of channel parameters, namely the sensitivity to poor estimation of the channel noise $N_o$ and to the effects of adjacent channel interference, is analyzed.

The parameters of the SA used in the simulations were as follows:

1) The Fano Metric, as derived in Appendix A, was used as the path metric.
2) The length of the stack was set at 1000 paths.
3) The observation interval length (the depth of the tree examined before a symbol was put out) was set at 100.
4) The simulation was continued until 100 error events had occurred or $2x10^6$ bits had been sent.
5) Buffer overflow was handled by simply overwriting (erasing) the paths at the bottom of the stack.
6) No regeneration of the root-node of the graph (also called an *ambiguity check* in [4]) was undertaken. When root-node regeneration was performed for some data points the results were similar. In general, using a non-regenerative algorithm, as done here, will improve error performance at the expense of memory since paths are not discarded early [20].

Omissions of certain practical aspects deserve to be mentioned:

1) The symbols used in the simulations were not split up into frames or packets, as would usually occur in practice. However, this should not affect the results since the frames are usually long enough that "long-term" averages apply.
2) No absolute time limit was set for forcing the algorithm to put out a symbol. For example, it could happen that the first symbol took thousands of time intervals to be put out while the second symbol was put out immediately after the first. Even though the probability of such an event occurring is very small for large signal to noise ratios, this is the most serious omission of the simulation model. Practically, some kind of check and recovery procedure must be implemented for such situations. The actual error performance will therefore be slightly worse than that presented.
3) The systolic priority queue could not be implemented in its parallel form with the computers and software available. Thus, no indication of the performance

speedup as compared to the Viterbi Algorithm could be gleaned from the results, other than the average number of branch extensions per symbol transmitted.

The SA's performance is compared to that of the Viterbi Algorithm and the M-Algorithm. For both of these breadth-first algorithms the observation interval length was set at 30 symbol intervals and the number of paths followed by the M-Algorithm was set at $M_{M\text{-}Algo}$= 8. The M-Algorithm was implemented without state checking and without root-node regeneration (results of simulations performed with regeneration proved similar to the results shown here).

It now remains to choose specific CPM schemes for simulation. It is generally accepted that a quaternary symbol alphabet (M=4) with a raised cosine phase pulse is best for uncoded CPM schemes [1]. Further, to allow comparison with the Viterbi Algorithm, the modulations chosen should not have too many states or else the simulations cannot be performed in a reasonable amount of time. The size of the matched filter bank is not of great concern due to the results of Huber and Liu (Chapter 4). The two CPM schemes investigated here have the following parameters:

1) M=4, L=3, RC with h=1/3. This scheme has $pM^{L-1} = 3 \cdot 4^2 = 48$ states. The bandwidths containing 99% and 99.9% of the power are $2B \cdot T_b = 0.63$ and 0.82, respectively; in comparison, the corresponding MSK bandwidths are 1.18 and 2.72. The asymptotic power gain over MSK is -0.78 dB.
2) M=4, L=3, RC with h=3/4. This scheme has $pM^{L-1} = 4 \cdot 4^2 = 64$ states. The 99% and 99.9% bandwidths are 1.26 and 1.48 respectively and the asymptotic power gain over MSK is 4.28 dB.

Note that the SA can handle a much larger number of states than these two schemes have. The first scheme was chosen to allow comparison with previous results [41] whereas the second scheme was chosen to illustrate the large amount of power gain possible in the same 99% bandwidth as MSK.

### 5.2.1. Error Performance and Computational Complexity

**Scheme 1: h=1/3, 48 States**

The error performance (in terms of the probability of an error *event* occurring) of the 3RC scheme with 48 states is shown in Figure 5.3. The SNR is normalized to $E_b/N_o$ and the bound in the figure is the curve $Q\left(\sqrt{d_{min}^2 \frac{E_b}{N_o}}\right)$.

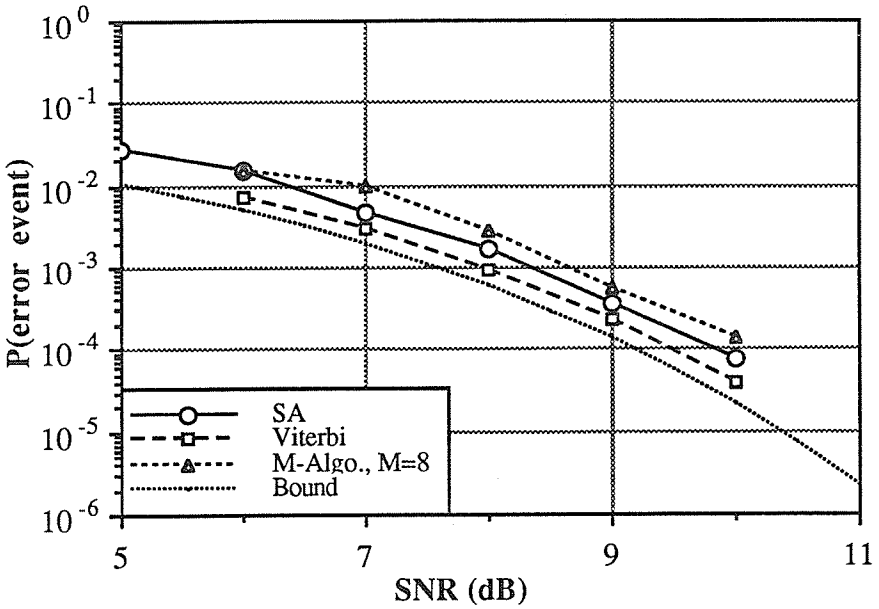


*Figure 5.3. Error performance for M=4, 3RC, h=1/3.*

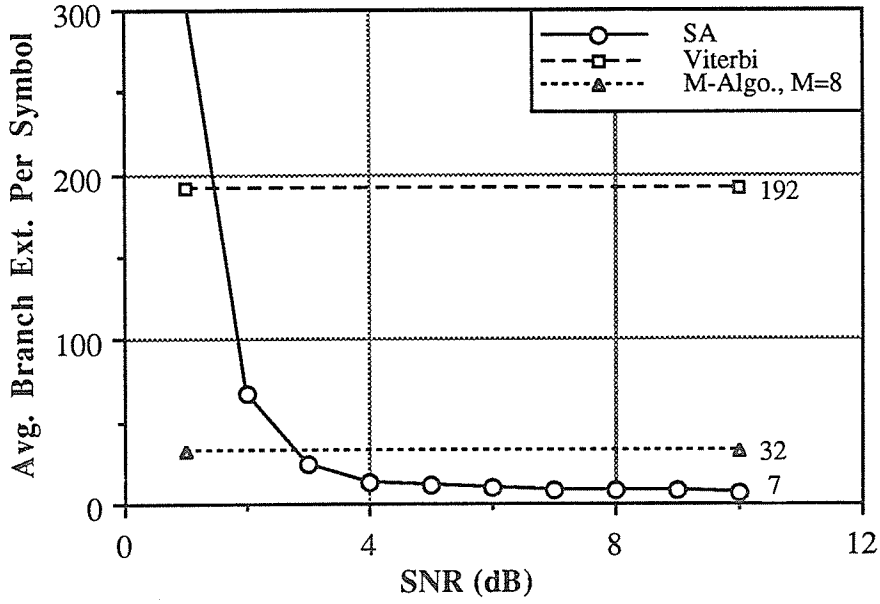


*Figure 5.4. Computational complexity for M=4, 3RC, h=1/3.*

The results are as expected: in terms of error performance all three algorithms perform similarly. The difference between these schemes lies in their computational complexities as measured in terms of the average number of branch extensions per output symbol (Figure 5.4). The SA extends a much smaller number of paths than either the Viterbi or M-Algorithms; the performance improvement factor is 27 and 5, respectively. The SA actually needs to perform more complex calculations than the other two algorithms but the savings are still noteworthy (one estimate gives the SA's performance as 10 times slower than the Viterbi Algorithm [30]; however, this estimate was based on a serial implementation and not a parallel one like the systolic array.).

The SA's performance also has a limit. Near a SNR of 1 dB the computational complexity of the SA blows up. This is because the computational cut-off rate, $R_o$, for this particular scheme has been reached [11]. Care must be taken not to operate below this SNR or else stack overflow will occur often and error performance becomes poor. Fortunately, this limit is not reached until the error performance for the MLSE Viterbi Algorithm is also poor.

### Scheme 2: h=3/4, 64 States

The error performance and computational complexity of the 64 state scheme are shown in Figures 5.5 and 5.6, respectively. The results are similar to those of the 48 state scheme. However, a subtle difference between the Stack- and M-Algorithms appears if the bit error probabilities are considered (Figure 5.7). The bit error probability for the M-Algorithm is unusually larger than its error event probability and upon closer examination of the error events, it was found that long error events tended to occur (at a SNR of 4 dB the average length of the 100 observed error events was 18 symbols). This did not occur for the M-Algorithm with the h=1/3, 48 state scheme.

The effect described above is the same as that found by Simmons to occur generally for schemes with modulation index h=k/p when $k > 1$ [40]. It turns out that due to the early decision mechanism of the M-Algorithm, the algorithm tends to lock onto paths which are located $2\pi \pm \Delta\varphi$ away in phase, where $\Delta\varphi$ is a small phase offset. The M-Algorithm has no way of getting out of such a false lock other than more noise pushing it back into the correct state. Inevitably, long error events with multiple symbol errors occur. One can combat this by introducing extra processing rules, as Simmons did in [40]. However, the extra computations required by such an approach slow down the algorithm by a factor of two to three; clearly this is undesirable.

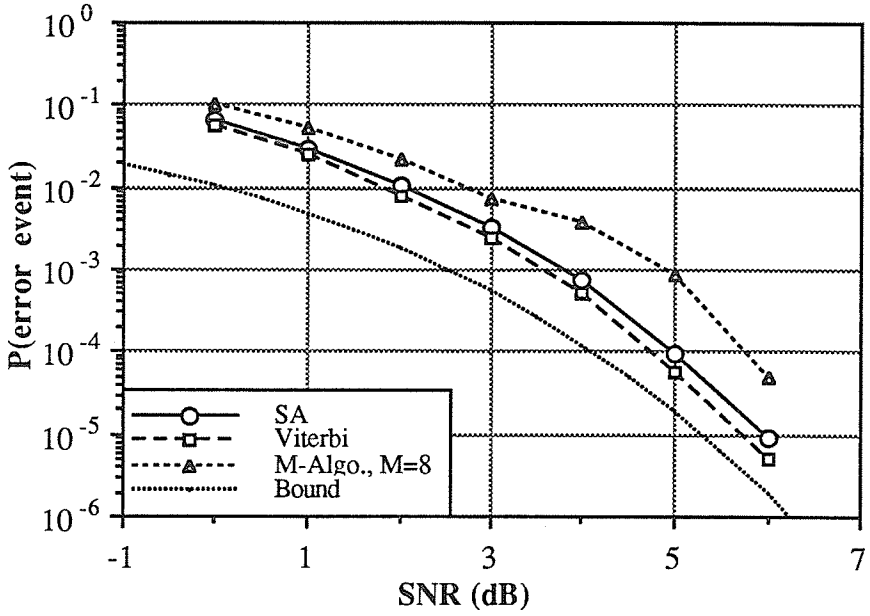


*Figure 5.5. Error performance for M=4, 3RC, h=3/4.*

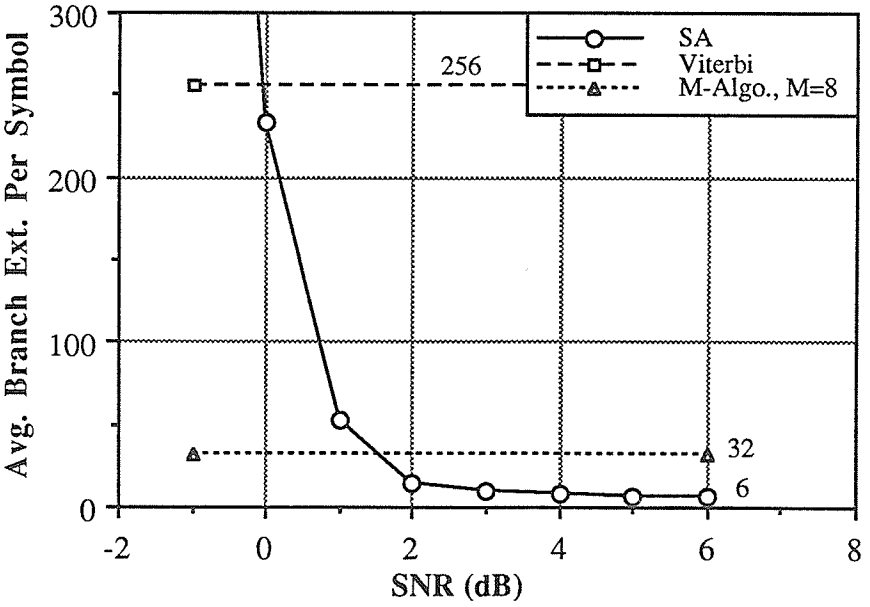


*Figure 5.6. Computational Complexity for M=4, 3RC, h=3/4.*

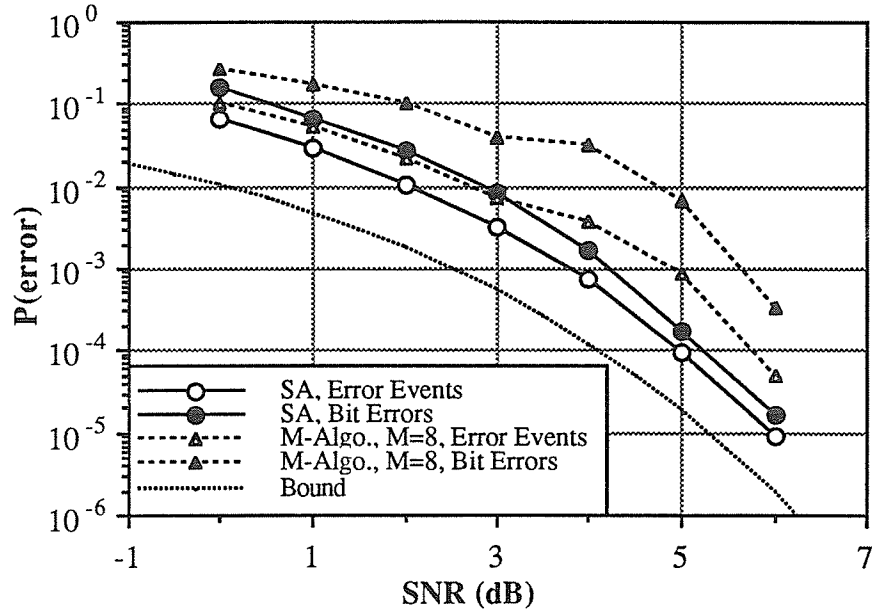


*Figure 5.7. Bit error probability for M=4, 3RC, h=3/4.*

The SA does not suffer from this problem (at 4 dB the average length of the error events was less than 6 symbols). The delayed decision mechanism under which the SA operates neutralizes the false lock phenomenon. Thus, for most modulation indices (those where k > 1 for h=k/p) the SA offers the additional advantage of keeping the length of the error events small, thereby also keeping the bit error probability low.

### 5.2.2. Sensitivity to $N_o$

The SA performs well under the conditions given in the above section, namely that the channel is AWGN and the noise level is known. Knowledge of the noise level is important for the SA, since $N_o$ must be known to calculate the Fano Metric. In this section the sensitivity of the SA's performance to poor estimation of $N_o$ is considered. The results are shown in Figures 5.8 and 5.9.

The error performance seems relatively insensitive to poor estimation of the signal-to-noise ratio. Even when the SNR estimation is ±3 dB off (double and half the actual SNR), the SA performs near the level of accurate estimation. Less degradation occurs for the case -3 dB, so that it seems better to underestimate the noise rather than overestimate it. This philosophy, however, is misleading.

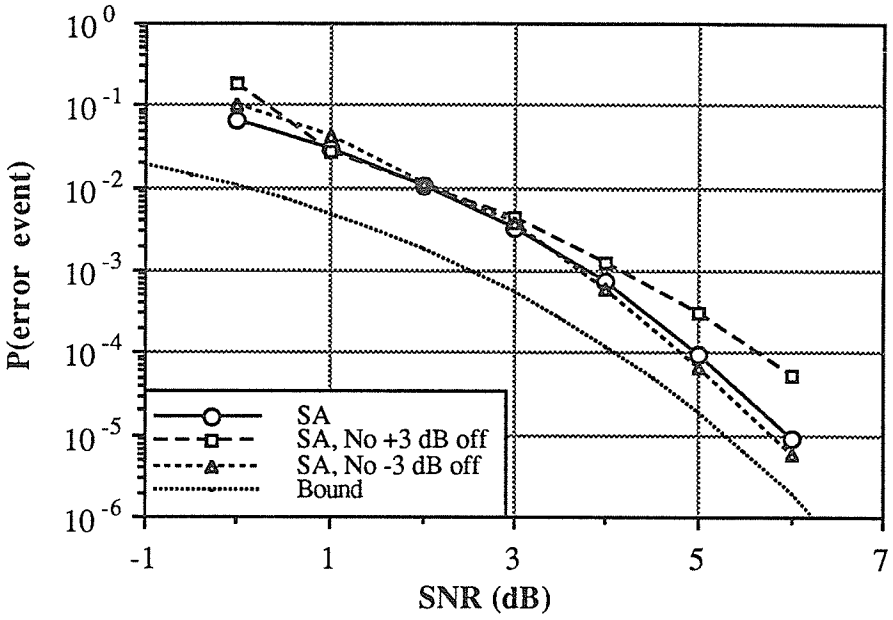


*Figure 5.8. Sensitivity of error performance to estimation of $N_o$.*

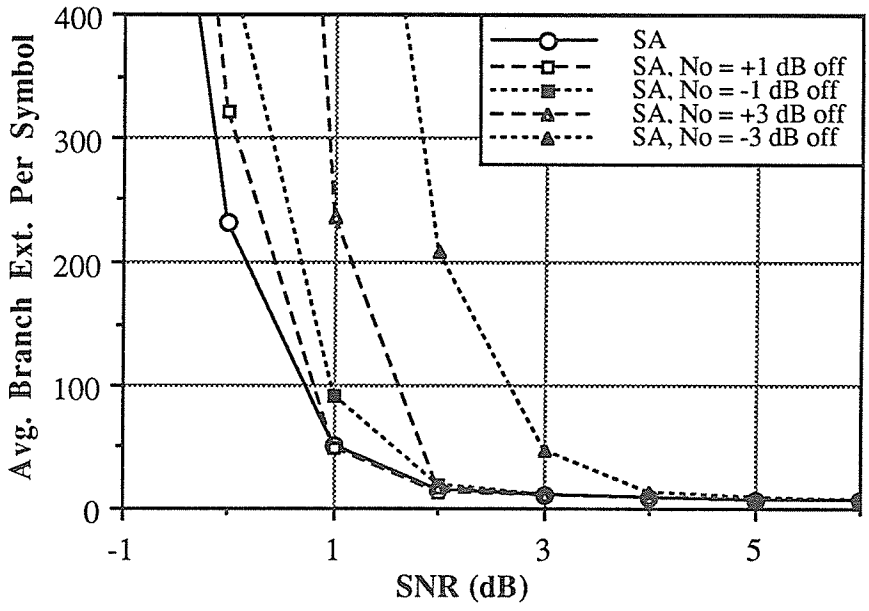


*Figure 5.9. Sensitivity of computational complexity to estimation of $N_o$.*

The real difference between the various levels of poor estimation is shown in Figure 5.9. When underestimating the noise, the receiver tends to search more; this is the actual reason for the good error performance shown in Figure 5.8. Furthermore, the limits where the number of computations becomes large (the SNR where the computational cut-off rate, $R_o$, is reached) becomes worse the poorer the estimation of $N_o$; the computational performance is also worse for underestimation of the noise than it is for overestimation. The lowest limit actually occurs when the correct Fano Metric is used. These results indicate that if one is operating near $R_o$ then care must be taken to properly estimate the noise level. On the other hand, if operation is well outside this range, poor estimation of $N_o$ has little impact on performance (as long as the estimate of SNR is not way-off, of course).

### 5.2.3. ACI Susceptibility

Real channels tend to have more impairments than white noise, and in this section the performance of the SA under the additional effects of adjacent channel interference (ACI) is considered. The effects of ACI on CPM receiver performance were studied in [37]; Viterbi detection was assumed there. More recently, the performance of the M-Algorithm and RSSE-type algorithms was examined by Simmons [41]. He found that under high levels of ACI the early decision property of these two algorithms severely degrades performance as compared to the Viterbi Algorithm. It was therefore hoped that the Stack Algorithm could overcome these problems just as for the bit error vs. error event problem. However, the results were disappointing, as described below.

The channel model used in this section assumes that a wide filter is placed around the desired signal's channel, allowing all of the two adjacent channels' signals to interfere (see Figure 5.10 where 3 dB of ACI for the M=4, 3RC, h=1/3 PSD's are shown). Further parameters of the channel model are:

1) The interfering signals are 2/T away in bandwidth.
2) The data sequences in all three channels are independent and uniformly distributed.
3) Packets of symbols of size 448 to 576 ($512 \pm 64$) are sent by both interfering channels.
4) After a packet is sent, the corresponding signal is resynchronized to random time and phase offsets (uniform distribution for both timing and phase).
5) The power levels in the adjacent channels are both equal and constant.
6) The receiver accurately estimates the noise level, $N_o$.

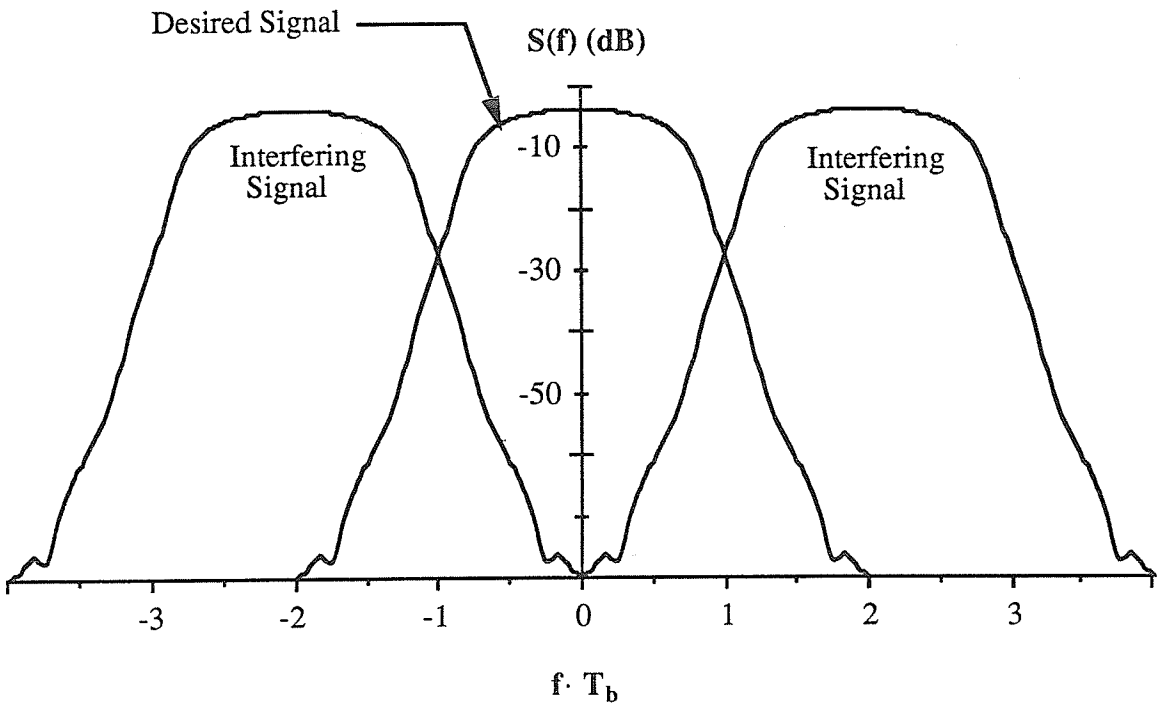


*Figure 5.10. ACI model (PSD's for M=4, 3RC, h=1/3 and 3 dB ACI).*

The first three parameters are chosen to make the interfering signals look as random as possible to the desired signal, as would realistically be the case. Nevertheless, the above model contains several unrealistic assumptions which must be considered when analyzing the results. Some problems with the model are:

1) The assumption that both interfering signals are completely passed through the bandpass filter is unrealistic for two reasons: i) the interfering signals' PSD extends to infinity in both directions so that a bandpass filter would certainly cut off some of the interfering power; ii) the bandpass filter will usually be narrow enough to filter out much of the adjacent channels' power. These inaccuracies are allowed to keep the programming simple.
2) The power levels in the adjacent channels will usually fluctuate.
3) The receiver will normally not get a good estimate of the SNR if there is additional interference.

The definition of ACI used here is the interferer- to desired-signal power ratio in the bandpass filter's bandwidth. This is not a standard definition, but we are only interested in a sensitivity comparison of the different algorithms' performances.

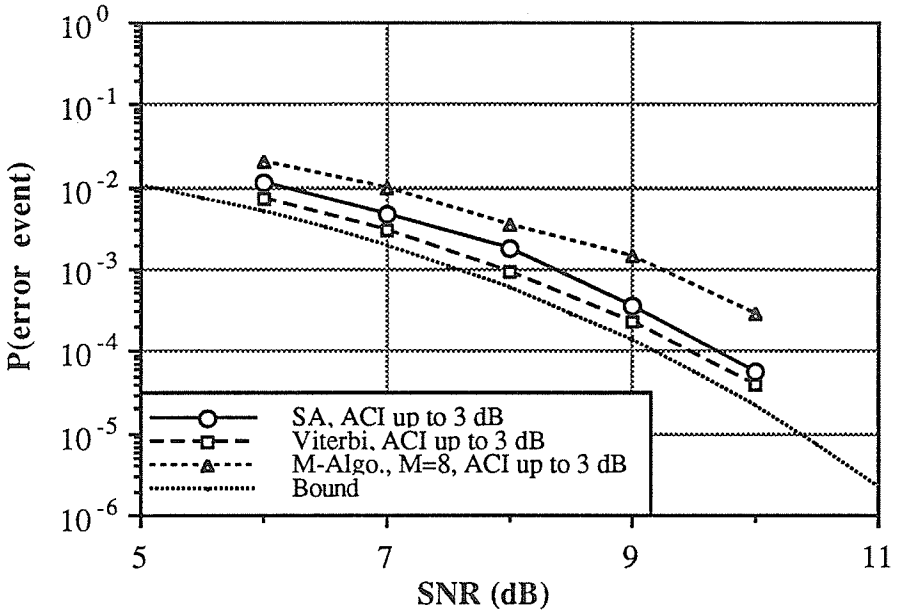


*Figure 5.11. Sensitivity of error performance to moderate levels of ACI.*

All algorithms were operated as if no ACI was present. Thus, neither the Viterbi Algorithm nor the SA are actually performing MLSE. This should be kept in mind when analyzing the results. The CPM scheme chosen for simulations is the 48 state scheme given above. This allows comparison with the results of [41].

The sensitivity of the error performance of the Stack, Viterbi and M-Algorithms to moderate levels of ACI is shown in Figure 5.11. All algorithms perform reasonably well up to ACI levels of 3 dB or double the interferer power to desired signal power. The computational complexity of the SA is still low for 3 dB ACI.

The results for high levels of ACI are shown in Figure 5.12. The Viterbi Algorithm's performance gracefully degrades as the ACI levels increase. In contrast, both the Stack and M-Algorithms' performance abruptly degrades; the poor behavior of the SA is in fact more pronounced than that of the M-Algorithm, an odd result considering the early-decision mechanisms responsible for the M-Algorithm's performance degradation are not present for the SA. A graph of the computational complexity of the SA under ACI makes this impetuous behavior even more apparent (Figure 5.13). The SA and the Fano metric seem to be very sensitive to correlated interference of the kind discussed here.

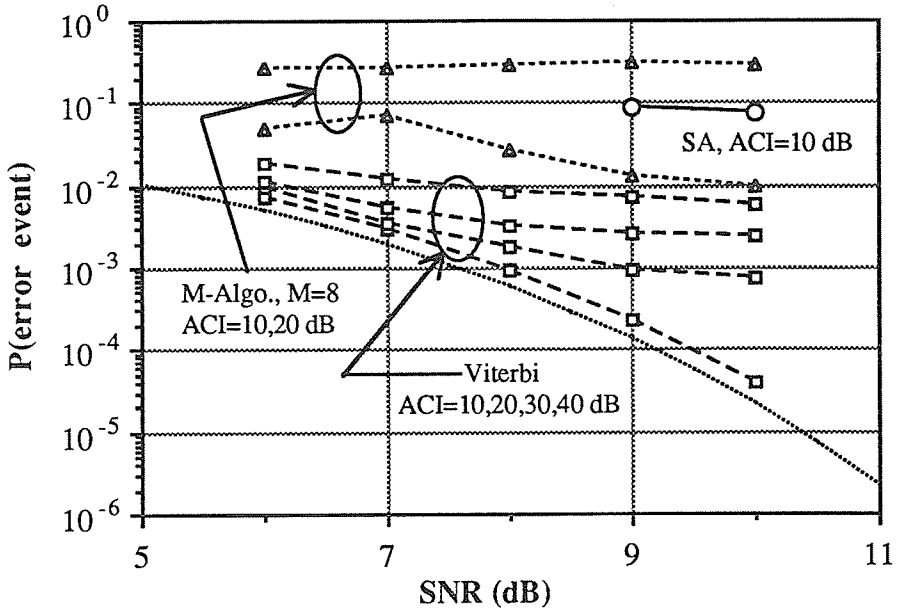


*Figure 5.12. Sensitivity of error performance to high levels of ACI.*

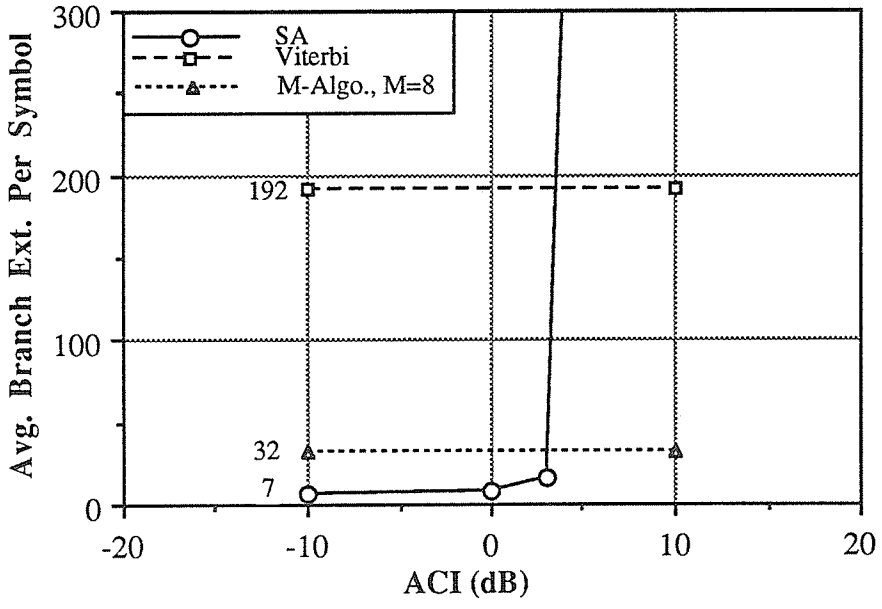


*Figure 5.13. Sensitivity of computational complexity to ACI (SNR = 10 dB).*

Based on Figures 5.12 and 5.13, it seems possible that the poor performance of the SA may be solely due to the exaggerated level of searching causing stack overflow. It was hypothesized that perhaps if the SA was forced forward by some extra mechanism both the error performance and computational complexity would return to their more usual values. Of course, forcing the algorithm forward is synonymous with making early decisions, just the type of operation one wants to avoid. Nevertheless, to force the SA forward, two parameters were modified: i) the receiver's estimate of $N_o/2$ was increased and ii) a positive bias was added to the metric to force the SA through the graph. Neither of these measures improved performance to better that of the M-Algorithm, although changing $N_o$ did improve both the error performance and computational complexity somewhat.

Another possible approach for combatting the poor performance of the SA is to incorporate the ACI model into the metric. Such an approach should also improve the Viterbi- and M-Algorithms' performance. However, the complete CPM channel model will become unwieldy theoretically so that some simplification is required. One possible approach is to assume the interfering signals are tone interferers with a random phase offset (Figure 5.14). The MLSE metric for this simplified channel model is relatively easy to obtain and could prove useful. This approach was not tested here.

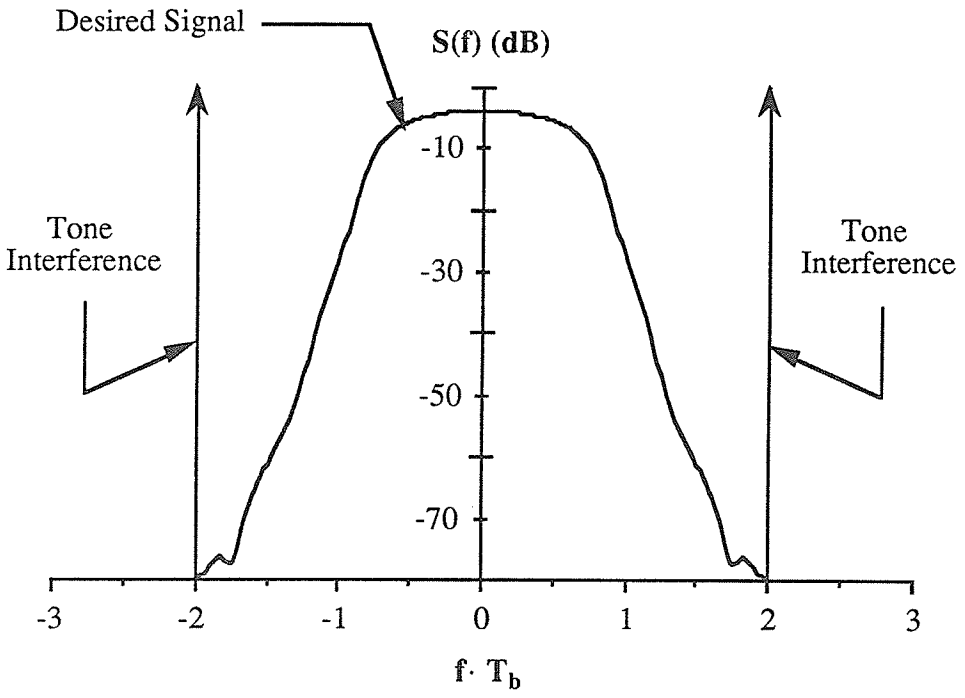


*Figure 5.14. A simplified ACI model.*

Yet another approach is to consider all three channels together as a vector channel and estimate the incoming data in both of the interfering channels along with the desired information; in effect, one is receiving the signals of all three channels simultaneously. The drawback with this approach is that the receiver becomes three times as costly since all three sequences must be estimated.

The most straight-forward solution to the problem of ACI is, of course, to use a good filter.

## 5.3. SUMMARY

The conclusions extracted from the simulation results are summarized below:

1) The Stack Algorithm outperforms both the Viterbi and M-Algorithms over the AWGN channel. The option of being able to detect schemes with a large state representation (large p, M, and L) is particularly desirable as the most power- and bandwidth-efficient CPM schemes can be utilized. Moreover, the SA does not suffer from the bit error vs. error event problem of the M-Algorithm.
2) The Stack Algorithm is not overly sensitive to poor estimation of the SNR. However, one must be cautious when operating near $R_o$.
3) The Stack Algorithm performs well for moderate levels of ACI but poorly for higher levels of ACI, even in comparison with the M-Algorithm . Thus, a good channel model or narrow bandpass filters are a necessity when using the SA to estimate the transmitted sequence under such conditions.

# CHAPTER 6. CONCLUSIONS

The two basic building blocks of a CPM receiver, the matched filter bank and the detection algorithm, were investigated. The matched filter bank, in an abstract sense, creates a space onto which the received signal can be projected and from which sufficient statistics for maximum likelihood sequence estimation can be obtained. Thus, for optimal detection, the number of matched filters depends directly on the number of dimensions which the signal set encompasses. In Chapter 2, it was shown that the dimensionality of CPM signal sets is related to the linear dependence of the phase pulse chips and can be determined by finding a basis for the space spanned by these chips. The larger the number of elements in the basis, the more matched filters are needed.

Chapter 3 explored the impact of the results of Chapter 2. Schemes which require a small filter bank were investigated; both single- and double-basis phase pulses were considered. It was found that these small basis pulses perform poorly in comparison with pulses with a large basis (Eg. RC) and this led to the investigation of suboptimal receivers.

Again using the results of Chapter 2, in Chapter 4 a suboptimal receiver with a small filter bank, the Piecewise Linear Detector (PLD), was developed. The PLD approximates a smooth CPM pulse by piecewise linear sections and since the dimensionality of the piecewise linear signal is relatively small, a reduction in the number of matched filters results. It was found that, while the PLD's performance was good, an approach proposed earlier by Huber and Liu was superior. A further improvement to Huber and Liu's receiver was suggested using the concepts of the PLD, namely using multiple receiver chips / transmitter chip.

Chapter 5 investigated sequential sequence estimation of CPM modulated data sequences via the Stack Algorithm (SA). Performance was measured in terms of error probability, computational complexity, sensitivity to estimation of channel parameters and sensitivity to adjacent channel interference. Comparisons with the Viterbi and M-Algorithms were made. It was found that, over the AWGN channel, the SA outperforms both the Viterbi and M-Algorithms; the SA performs essentially MLSE with many fewer computations. The performance of the SA is also insensitive to estimation in SNR, unless one is operating near the computational cutoff rate. In this case, care must be taken to estimate the SNR accurately.

All algorithms operate well in moderate levels of ACI, up to 3 dB for the model presented here. However, when ACI is further increased, the SA's performance degrades

sharply, even more rapidly than the M-Algorithm; the Viterbi Algorithm's performance degrades gracefully. The simulations suggest that the SA metric is sensitive to such changes in channel parameters. Therefore, a good channel model and accurate parameter estimation or a narrow bandpass filter is vital when using the SA in an interferer rich environment.

### Future Work

Several issues addressed in the thesis were left incomplete and deserve further attention:

1) Using multiple receiver chips / transmitter chip was suggested as a further improvement for the Huber and Liu receiver. The performance gains of such an approach need to be determined and the trade-offs (performance vs. complexity) should be examined in more detail.
2) The ACI simulations were performed by simply applying interference and letting the graph search algorithms run as if no ACI was present. A better method is to include the effects of ACI in the channel model to obtain the MLSE metric for this channel. Some approaches for doing this were suggested in Chapter 5 and deserve to be investigated.
3) In Appendix A, techniques for making the Fano Metric calculation simpler are suggested. These must be tested.
4) Some of the results presented in Chapter 5 are applicable to coding. For example, sequential algorithms are well suited for decoding large constraint length codes which have a large state representation. These codes could be further investigated for CPM.
5) Recent work by Yang and Taylor [46], building on results from Rimoldi [38,39] and Morales-Moreno and Pasupathy [33], shows that it is advantageous to use codes over rings to improve CPM error performance and simultaneously keep the number of states small. It turns out that by using such codes, spectral performance can be improved. Some indication of the bandwidth improvement can be obtained using the results of Chapter 2; this is further developed in Appendix B.

# REFERENCES


[1] J.B. Anderson, T. Aulin, and C-E.W. Sundberg, *Digital Phase Modulation.* New York, NY: Plenum, 1986.

[2] J.B. Anderson and S. Mohan, *Source and Channel Coding, An Algorithmic Approach.* Boston: Kluwer Academic Publishers, 1991.

[3] J.B. Anderson and D.P. Taylor, "A bandwidth-efficient class of signal-space codes," *IEEE Trans. Inform. Theory*, vol. IT-24, pp. 703-712, Nov. 1978.

[4] J.B. Anderson and S. Mohan, "Sequential coding algorithms: a survey and cost analysis," *IEEE Trans. Commun.*, vol. COM-32, pp. 169-176, Feb. 1984.

[5] J.B. Anderson and C-E.W. Sundberg, "Advances in constant envelope coded modulation," *IEEE Commun. Mag.*, pp. 36-45, Dec. 1991.

[6] T. Aulin, C-E.W. Sundberg, "Continuous phase modulation - Part I: Full response signaling," and T.Aulin, N. Rydbeck, and C-E.W. Sundberg, "Continuous phase modulation - Part II: Partial response signaling," *IEEE Trans. Commun.*, vol. COM-29, pp. 196-225, March 1981.

[7] T.J. Baker, "Asymptotic behavior of digital FM spectra," *IEEE Trans. Commun.*, vol. COM-22, pp. 1585-1594, Oct. 1974.

[8] W.R. Bennet and S.O. Rice, "Spectral density and autocorrelation functions associated with binary frequency-shift keying," *Bell Syst. Tech. J.*, vol. 42, pp. 2355-2385, Sept. 1963.

[9] R. de Buda, "Coherent demodulation of frequency -shift keying with low deviation ratio," *IEEE Trans. Commun.*, vol. COM-20, pp. 429-436, June 1972.

[10] C.Y. Chang and K. Yao, "Systolic array architecture for the sequential stack decoding algorithm," *SPIE Conf.*, San Diego, U.S.A., pp. 196-203, 1986.

[11] G.C. Clark, Jr., and J.B. Cain, *Error-Correction Coding for Digital Communications.* New York: Plenum, 1981.

[12] L.W. Couch, *Digital and Analog Communication Systems.* 3rd ed. New York: Macmillan, 1990.

[13] Q. Dai and E. Shwedyk, "Detection of bandlimited signals over frequency selective Rayleigh fading channels," accepted for publication by *IEEE Trans. Commun.*

[14] G.S. Deshpande and P.H. Wittke, "Correlative encoded digital FM," *IEEE Trans. Commun.*, vol. COM-29, pp. 156-162, Feb. 1981.

[15] R.M. Fano, "A heuristic discussion of probabilistic decoding," *IEEE Trans. Inform. Theory*, pp. 64-74, April 1963.

[16] G. Fettweis and H. Meyr, "High-speed parallel Viterbi decoding: algorithm and VLSI-architecture," *IEEE Commun. Mag.*, pp. 46-55, May 1991.

[17] J.P. Fonseka, "Nonlinear continuous phase frequency shift keying," *IEEE Trans. Commun.*, vol. COM-39, pp. 1473-1481, Oct. 1991.

[18] G.D. Forney, Jr. and E.K. Bower, "A high-speed sequential decoder: prototype design and test," *IEEE Trans. Commun. Technol.*, vol. COM-19, pp. 821-835, Oct. 1971.

[19] G.D. Forney, Jr., "The Viterbi Algorithm," *Proc. IEEE*, vol. 61, pp. 268-278, March 1973.

[20] H-M. Hang and J.W. Woods, "Near merging of paths in suboptimal tree searching," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 567-573, May 1984.

[21] P.K.M. Ho and P.J. McLane, "Power spectral density of digital continuous phase modulation with correlated data symbols," Parts I and II, *Proc. IEE*, vol. 133, pt. F, pp. 95-114, Feb. 1986.

[22] P.K.M. Ho and P.J. McLane, "Spectrum, distance, and receiver complexity of encoded continuous phase modulation", *IEEE Trans. Inform. Theory*, vol. IT-34, pp. 1021-1032, Sept. 1988.

[23] J. Huber and W. Liu, "An alternative approach to reduced-complexity CPM-receivers," *IEEE J. Sel. Areas in Commun.*, vol. 7, pp.1437-1449, Dec. 1989.

[24] J. Huber and W. Liu, "Data-aided synchronization of coherent CPM-receivers", *IEEE Trans. Commun.*, vol. COM-40, pp. 178-189, Jan. 1992.

[25] F. de Jager and C.B. Dekker, "Tamed frequency modulation, a novel method to achieve spectrum economy in digital transmission," *IEEE Trans. Commun.*, vol. COM-26, pp. 534-542, May 1978.

[26] P. Kabal and S. Pasupathy, "Partial-response signaling," *IEEE Trans. Commun.*, vol. COM-23, pp. 921-934, Sept. 1975.

[27] R.A. Korkosz and D.V. Sarwate, "An analytical expression for power spectra of M-ary LREC continuous phase modulated signals," in the *Conf. Record ICC '92*, pp. 323.5.1-4, June 1992.

[28] H.J. Landau and H.O. Pollak, "Prolate spheroidal wave functions, Fourier analysis and uncertainty-III: The dimension of the space of essentially time- and band-limited signals," *Bell Syst. Tech. J.*, vol. 41, pp. 1295-1336, July 1962.

[29] G. Lindell, C-E.W. Sundberg, and T. Aulin, "Minimum Euclidean distance for combinations of short rate 1/2 convolutional codes and CPFSK modulation," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 509-519, May 1984.

[30] H. Ma, "The multiple stack algorithm implemented on a Zilog Z-80 microcomputer," *IEEE Trans. Commun.*, vol. COM-28, pp. 1876-1882, Nov., 1980.

[31] J.L. Massey, "Nonsystematic convolutional codes for sequential decoding in space applications," *IEEE Trans. Commun. Technol.*, vol. COM-19, pp. 806-813, Oct. 1971.

[32] J.L. Massey, "Variable-length codes and the Fano Metric," *IEEE Trans. Inform. Theory*, vol. IT-18, pp. 196-198, Jan. 1972.

[33] F. Morales-Moreno and S. Pasupathy, "Structure, optimization, and realization of FFSK trellis codes," *IEEE Trans. Inform. Theory*, vol. IT-34, pp.730-751.

[34] S.V. Pizzi and S.G. Wilson, "Convolutional coding combined with continuous phase modulation," *IEEE Trans. Commun.*, vol. COM-33, pp. 20-29, Jan. 1985.

[35] Premji and Taylor, "Receiver structures for multi-h signaling formats", *IEEE Trans. Commun.*, vol. COM-35, pp. 439-451, April 1987.

[36] R.M.A.P. Rajatheva and E. Shwedyk, "Demodulation for intersymbol interference channels in the presence of colored Gaussian noise," in the *Proc., IEEE Int. Symp. Inform. Theory*, Budapest, Hungary, June 1991, p.33.

[37] R.B. Rhodes, S.G. Wilson, and A.B. Svensson, "MSK-type reception of continuous phase modulation: cochannel and adjacent channel interference," *IEEE Trans. Commun.*, vol. COM-35, pp. 185-193, Feb. 1987.

[38] B.E. Rimoldi, "A decomposition approach to CPM," *IEEE Trans. Inform. Theory*, vol. IT-34, pp. 260-270, March 1988.

[39] B.E. Rimoldi, "Design of coded CPFSK modulation systems for bandwidth and energy efficiency," *IEEE Trans. Commun.*, vol. COM-37, pp. 897-905, Sept. 1989.

[40] S.J. Simmons and P.H. Wittke, "Low complexity decoders for constant envelope digital modulations," *IEEE Trans. Commun.*, vol. COM-31, pp.1273-1280, Dec. 1983.

[41] S.J. Simmons, "ACI susceptibility of reduced-state decoding for CPM," in *Proc. 16th Biennial Symp. Communications*, Queen's University, Kingston, ON, Canada, May 1992, pp. 261-264.

[42] D.J. Tempel and E. Shwedyk, "Sequential decoding of linear block codes," submitted to *IEEE Int. Symp. Inform. Theory*, San Antonio, Texas, 1993.

[43] H.L. Van Trees, *Detection, Estimation, and Modulation Theory*, Part 1, New York: Wiley, 1968.

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

[45] F. Xiong, A. Zerik, and E. Shwedyk, "Sequential sequence estimation for channels with intersymbol interference of finite or infinite length," *IEEE Trans. Commun.*, vol. COM-38, pp. 795-803, June 1990.

[46] R.H. Yang and D.P. Taylor, "On trellis coded CPFSK," in *Proc. 16th Biennial Symp. Communications*, Queen's University, Kingston, ON, Canada, May 1992, pp. 87-90.

[47] R.H. Yang, private communication.

# APPENDIX A. THE FANO METRIC FOR CPM

For sequential detection one wants to extend the maximum a posteriori (MAP) sequence. To do this, one must find the sequence which maximizes the conditional probability density function:

$$\Pr(\alpha|\mathbf{r}) = \frac{\Pr(\mathbf{r},\alpha)}{\Pr(\mathbf{r})} \tag{A.1}$$

where:

$\alpha = [\alpha_0, \alpha_1, ..., \alpha_{N-1}]$ is a sequence of symbols with $\alpha_i \in \{-(M-1), -(M-3), ..., M-1\}$,

$\mathbf{r} = [\mathbf{r}_0, \mathbf{r}_1, ..., \mathbf{r}_{N-1}] = [(r_0^1, r_0^2, ..., r_0^D), ..., (r_{N-1}^1, r_{N-1}^2, ..., r_{N-1}^D)]$ are the sufficient statistics for sequence estimation, and

D is the number of orthogonal dimensions needed to represent the signal set. (Note that for CPM: $D \le 2M^L$, where L is the length of the frequency pulse).

Since $\Pr(\mathbf{r})$ is the same for all sequences, maximizing $\Pr(\alpha|\mathbf{r})$ with respect to $\alpha$ is equivalent to maximizing $\Pr(\mathbf{r},\alpha)$ with respect to $\alpha$. For the purpose of the discussion below, identify $r(t)$ with $\mathbf{r}$ so that $\Pr(r(t)|\alpha) = \Pr(\mathbf{r}|\alpha)$. Further, consider $r(t)$ as a sequence of chips: $r(t) = \sum_{i=0}^{N-1} r_i(t)$ where $r_i(t) = 0$, $t \notin [iT,(i+1)T)$.

Now assume that $r(t)$ has been observed over the first N symbol intervals and the path under consideration has length $n \le N$. Let $\alpha = [\mathbf{a},\mathbf{t}]$ where: $\mathbf{a} = [\alpha_0, \alpha_1,...,\alpha_{n-1}]$ and $\mathbf{t} = [\alpha_n, \alpha_{n+1},...,\alpha_{N-1}]$. One now wants to extend the MAP path of all paths under consideration. This is the path, $\mathbf{a}$, which maximizes:

$$\Pr(r(t),\mathbf{a}) = \Pr(\mathbf{a})\Pr(r(t)|\mathbf{a}) = \Pr(\mathbf{a})\left[\prod_{i=0}^{n-1}\Pr(r_i(t)|a_0,...,a_i)\right]\left[\prod_{j=0}^{N-n-1}\Pr(r_{j+n}(t)|\mathbf{a})\right] \tag{A.2}$$

The second product in (A.2) contains observations which occur after the n'th symbol interval. One would like to eliminate this term so that constant updating of every path metric after a new chip is received is unnecessary. However, due to the memory of the modulation scheme, this cannot be done without making some simplifying assumptions.

The Fano Metric for decoding convolutional codes is based on the assumption that tail symbols are statistically independent of the prior symbols. This assumption is unnecessary for the uncoded CPM case; instead, assume the *signal chip tails* are independent of the prior signal chips:

$$\Pr(r_{j+n}(t)|\mathbf{a}) = \Pr(r_{j+n}(t)) \ , \ j=0,1,...,N-n-1 \qquad (A.3)$$

In other words, all $r_i(t)$ are assumed possible for any **a** in the symbol intervals beyond the n'th. In effect, one is adding random tails to the received *signal* r(t).

The above is somewhat different than adding random tails to the received *sequence*, as is done in sequential decoding of convolutional codes [32] or block codes [42] over a discrete memoryless channel. In these cases the tail symbols are assumed statistically independent, even though the future symbols must be dependent due to the code. Thus, the simplifying assumption stated above is needed in either case, just in a different form. (Note that if sequential detection/decoding of a coded CPM scheme is desired, both the *signal* and the *symbol* tails must be assumed independent of each other, even though the future signal chips and symbols are actually dependent).

Using the simplification of (A.3), equation (A.2) yields:

$$\Pr(r(t),\mathbf{a}) = \Pr(\mathbf{a}) \left[ \prod_{i=0}^{n-1} \Pr(r_i(t)|a_0,...,a_i) \right] \left[ \prod_{j=0}^{N-n-1} \Pr(r_{j+n}(t)) \right] \qquad (A.4)$$

Maximizing (A.4) with respect to **a** is equivalent to maximizing:

$$\log \left[ \frac{\Pr(r(t),\mathbf{a})}{\prod_{i=0}^{N-1} \Pr(r_i(t))} \right] = \log \left[ \Pr(\mathbf{a}) \prod_{i=0}^{n-1} \frac{\Pr(r_i(t)|a_0,...,a_i)}{\Pr(r_i(t))} \right] \qquad (A.5)$$

with respect to **a**, where:

$$\Pr(r_i(t)) = \sum_{\sigma} \Pr(r_i(t)|\sigma) \Pr(\sigma) \qquad (A.6)$$

and the sum runs over all $pM^L$ possible phase transitions, $\sigma$, in a symbol interval (assuming a finite state representation is possible; irrational modulation indices are treated below). In the case where k (in h=k/p) is odd, the sum could also be made to run over all $2pM^L$ phase transitions in all symbol intervals ; by doing this, the same sum generating circuit can be used for each symbol interval.

Using (A.5), the metric for a specific path in the tree becomes:

$$\Lambda_{n,a} = \sum_{i=0}^{n-1} \left[ \log\left\{ \frac{\Pr(r_i(t)|a_0,\ldots,a_i)}{\Pr(r_i(t))} \right\} + \frac{1}{n} \log\{\Pr(a)\} \right] \tag{A.7}$$

and the branch metric is:

$$\lambda_{n,a} = \log\left\{ \frac{\Pr(r_{n-1}(t)|a_0,\ldots,a_{n-1})}{\Pr(r_{n-1}(t))} \right\} + \frac{1}{n} \log\{\Pr(a)\} \tag{A.8}$$

which is almost identical to the Fano metric for the decoding of convolutional codes over the discrete memoryless channel [32], except that the received chip is also dependent on all prior symbols and not just the present one.

### The AWGN Channel

Over the AWGN channel:

$$\begin{aligned} \Pr(r_i(t)|a) &= \frac{1}{(\pi N_o)^{D/2}} \exp\left\{ -\frac{1}{N_o} \left[ \sum_{k=1}^{D} (r_i^k - s_a^k)^2 \right] \right\} \\ &= \frac{1}{(\pi N_o)^{D/2}} \exp\left\{ -\frac{1}{N_o} \int_{(i-1)T}^{iT} [r(t) - s_a(t)]^2 \, dt \right\} \end{aligned} \tag{A.9}$$

where $N_o/2$ is the spectral height of the white noise. Inserting equations (A.6) and (A.9) into (A.8) yields (ignoring double frequency terms):

$$\begin{aligned} \lambda_{n,a} = &\left[ \frac{2}{N_o} \log(e) \int_{nT}^{(n+1)T} r(t)\, s_\alpha(t)\, dt \right] \\ &- \log\left[ \sum_{\sigma} \exp\left\{ \frac{2}{N_o} \int_{nT}^{(n+1)T} r(t)\, s_\sigma(t)\, dt \right\} f_\sigma(\sigma) \right] + \frac{1}{n} \log\{f_a(a)\} \end{aligned} \tag{A.10}$$

In most cases, both the symbols **a** and the transitions $\sigma$ can be assumed to be uniformly distributed. Including these assumptions in (A.10) gives the final form of the Fano metric for CPM and the AWGN channel:

$$\lambda_{n,a} = \left[\frac{2}{N_o}\log(e)\int_{nT}^{(n+1)T} r(t)\, s_\alpha(t)\, dt\right] - \log\left[\sum_{\sigma}\frac{1}{pM^L}\exp\left\{\frac{2}{N_o}\int_{nT}^{(n+1)T} r(t)\, s_\sigma(t)\, dt\right\}\right] - \log(M) \tag{A.11}$$

**Irrational Modulation Indices and Complexity Reduction**

To calculate the branch metric for a given interval, the second term of equation (A.11) requires one to obtain all $pM^L$ sufficient statistics. This is disadvantageous for two reasons: i) *all* branch metrics, and not only the one of interest, are needed and ii) when p, M and L are large, as for the more power and bandwidth efficient CPM schemes, calculation of the Fano Metric becomes computationally intensive. To avoid this, one can simplify the Fano Metric in equation (A.11) by using the Fano Metric for irrational modulation indices. For irrational h, $Pr(r_i(t))$ becomes:

$$Pr\,(r_i(t)) = \sum_{\sigma_{cp}}\frac{1}{M^L}\left\{\int_0^{2\pi} Pr(r_i(t)|\sigma_{cp},\theta_p)\, Pr(\sigma_{cp},\theta_p)\, d\theta_p\right\} \tag{A.12}$$

where $\sigma_{cp}$ represents the $M^L$ phase state independent classes of signals in the signal set and $\theta_p$ represents the phase state (note that one could actually sum over the signal elements of the *dynamic signal set* instead of $\sigma_{cp}$; for the sake of clarity this approach is avoided). Over the AWGN channel, the integral in (A.12) can be reduced to (again keeping only the cross-product term, dropping terms which cancel with the numerator of (A.5), and converting to baseband as in equation (1.10)):

$$\frac{1}{2\pi}\int_0^{2\pi}\exp\left(\frac{1}{N_o}\int_{nT}^{(n+1)T}\left[\tilde{I}(t)I(\sigma_{cp},\theta_p,t) + \tilde{Q}(t)Q(\sigma_{cp},\theta_p,t)\right]dt\right)d\theta_p \tag{A.13}$$

where (normalizing the symbol power, E/T = 1/2):

$$I(\sigma_{cp},\theta_p,t) = \cos\left(\phi(t) + \theta_p\right)$$
$$Q(\sigma_{cp},\theta_p,t) = \sin\left(\phi(t) + \theta_p\right) \qquad \text{and} \quad \phi(t) = \theta_c(t) + \theta_{cp}(t) \tag{A.14}$$

Inserting these expressions into (A.13) and expanding out the cosine and sine terms yields:

$$\begin{aligned}
&\frac{1}{2\pi}\int_0^{2\pi} \exp\left[A\cos(\theta_p) + B\sin(\theta_p)\right] d\theta_p \\
&= \frac{1}{2\pi}\int_0^{2\pi} \exp\left[\sqrt{A^2+B^2}\cos\left(\theta_p - \tan^{-1}\frac{B}{A}\right)\right] d\theta_p \\
&= I_o\left(\sqrt{A^2+B^2}\right)
\end{aligned} \tag{A.15}$$

where:

$$\begin{aligned}
A &= \frac{1}{N_o}\int_{nT}^{(n+1)T} \left[\tilde{I}(t)\cos\phi(t) + \tilde{Q}(t)\sin\phi(t)\right] dt \\
B &= \frac{1}{N_o}\int_{nT}^{(n+1)T} \left[-\tilde{I}(t)\sin\phi(t) + \tilde{Q}(t)\cos\phi(t)\right] dt
\end{aligned} \tag{A.16}$$

and $I_0$ is the modified Bessel function of the first kind of order 0. The branch metric of equation (A.11) is then:

$$\lambda_{n,a} = \left[\frac{2}{N_o}\log(e)\int_{nT}^{(n+1)T} r(t)\, s_\alpha(t)\, dt\right] - \log\left[\sum_{\sigma_{cp}} \frac{1}{M^L} I_0\left(\sqrt{A^2+B^2}\right)\right] - \log(M) \tag{A.17}$$

From equation (A.17) it is apparent that to evaluate the Fano Metric for schemes with irrational modulation index, only one function evaluation is needed for each phase state independent class of signals. This is a reduction by a factor of p over schemes with modulation index h=k/p.

To achieve the same reduction for rational modulation indices, one can simply use the Fano Metric of equation (A.17). In this way, one modified Bessel function evaluation replaces p exponential function evaluations. This approach should work well for well for large p ( and for small p it is unnecessary).

### Further Complexity Reduction

Reducing the number of function evaluations from $pM^L$ to $M^L$ will not be significant if p is small and M and L are large. To further reduce the number of evaluations, recall that the terms in the exponentials are just the correlations of the received signal with the

transmitter phase functions. The phase functions are evenly spread out over the time interval [0,T) and baseband frequency interval $\left[-(M-1)\,h/2T,\ (M-1)\,h/2T\right)$, and the term we are concerned with here (equation (A.6)) is simply an average. Thus, it should be possible to use just a few representative phase functions to get a good approximation to (A.6). For example, one could use the six filters (which cover three dimensions) of Huber and Liu's matched filtering scheme to obtain a good "mix" of phase functions. This reduces the number of function evaluations from $M^L$ to three (it may, of course, be necessary to use a few more signals than just the three frequencies).

Although this approach and its effect on performance has not been investigated in this thesis, it should work well since the primary function of the term in equation (A.6) is to perform an averaging of sufficient statistics. It seems natural to expect that only a few representative signals give a good approximation to this average, especially for signal sets which are localized in time and frequency as CPM signal sets are.

# APPENDIX B. CODING FOR BANDWIDTH PERFORMANCE

The reason for placing a convolutional coder in front of a CPM modulator is normally to improve error performance. Here, the effects of the coder on bandwidth are also considered. The approach is not the same as that of Ho and McLane [21] where the PSD's of various coded CPM schemes are calculated. Rather, some simple assumptions about the bandwidth are made based on the probability distribution of the signals in the dynamic signal set. The validity of these assumptions has not been tested; as such, the discussion below serves solely to stimulate further research.

## REC CPM

To introduce the measure of bandwidth used here, consider the REC phase pulse. Optimum detection of REC CPM schemes requires a linearly increasing number of matched filters with M, the size of the signalling alphabet and L, the length of the frequency pulse. This is, of course, due to the single basis nature of rectangular pulses and during any symbol interval, the CPM transmitter sends one of the frequencies (see equation (2.5)):

$$f_n = f_c + \frac{h}{2LT}[-L(M-1) + 2n] \quad , n = 0, 1,..., L(M-1) \tag{B.1}$$

As such, REC CPM is nothing more than a FSK scheme with a convolutional encoder placed in front which controls the pattern of frequency hopping and maintains phase continuity [38] (the condition of phase continuity will, of course, increase the size of the signal set via constant phase offsets, but for the following discussion only the frequencies are considered). When $L > 1$, the following effects of partial response signaling [26] will apply to the frequencies of equation (B.1):

1) The probability distribution of the allowable frequencies changes.
2) The size of the frequency hops is limited.

For example, consider binary signaling (M=2) and enumerate the possible frequencies from 0 to L. Then the probability distribution of frequencies is binomial and the maximum size of hop is $\pm 2$ (see Figure B.1, where the *weight distribution* for L=3 is given; the weight distribution is simply the probability distribution normalized to integer values).

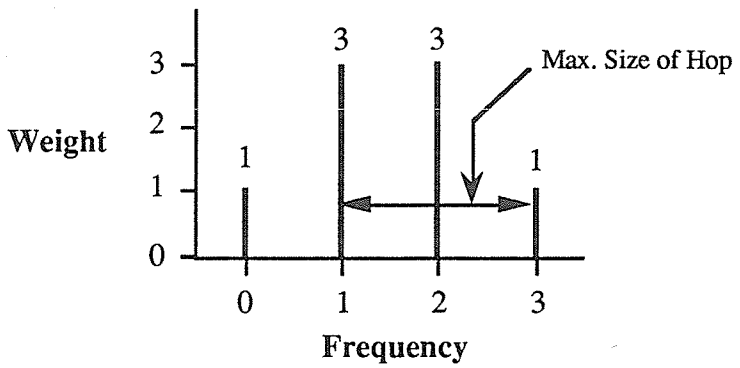


*Figure B.1. Frequency weight distribution of 3REC frequencies.*

The effect of the partial response coding is apparent from the distribution of frequencies in Figure B.1; better bandwidth performance is achieved because the center frequencies (frequencies #1 and #2) tend to occur more often. In this sense, partial response signaling performs a bandwidth shaping by changing the probability distribution of the frequencies of the REC scheme. Limiting the frequency hops will also reduce bandwidth. The same type of effect occurs for larger signaling alphabets, although now the distribution is not binomial and larger frequency hops can occur.

The results stated above apply to other CPM single basis phase pulses. The only difference is that now no simple frequency domain description is possible as for REC. It may even be possible to generalize to multiple basis CPM using the vector and matrix representations of Chapter 2, but this was not attempted. In the following, only REC schemes are considered.

### An Assumption About Bandwidth

In the following, coded CPM schemes are considered and a simple measure of their bandwidth is presented; the bandwidth measure used here will simply be the weight distribution of the frequencies. For example, if a coded CPM scheme has the same frequency weight distribution as in Figure B.1, it is assumed that the bandwidth of this CPM scheme is the same as 3REC (assuming the same separation between frequencies, of course). This measure is not completely accurate, as the size of the frequency hops and the correlations between the hops will also affect bandwidth. However, it seems intuitive that the probability distribution of the frequencies will play the major role in determining bandwidth.

## Coding

Several coding schemes for CPM have been proposed. One approach is to simply encode the incoming data sequence and then pass the coded symbols on to the CPM modulator [29,34]. Two of the more recent approaches are those given by Rimoldi [39] and Yang and Taylor [46], both of which are briefly reviewed below.

Rimoldi's coder is based on increasing the size of the signaling alphabet and then using set-partitioning principles to improve error performance. The bandwidth is kept the same as uncoded schemes by using modulation indices of the form h = 1/M [39]. Such modulation indices also make the structure of the CPM encoder more amenable to analysis. A skeletal block diagram of Rimoldi's encoder is given in Figure B.2. The complete encoder consists of two parts. The first block is a binary (or $2^m$-ary) rate 1/2 (or rate $^m/_{m+1}$) encoder. The encoded sequence is then mapped to a 4-ary (or $2^{m+1}$-ary) data sequence which is sent to the CPM modulator. Note that the combination of these two blocks does not affect the data rate, as the mapper is a rate 2 (or rate $1 + {}^1/_m$) encoder.

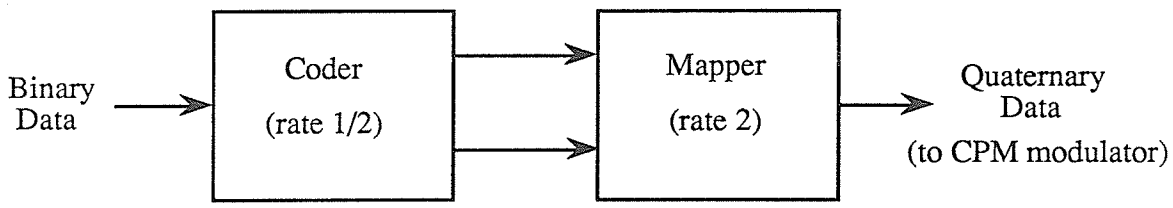


*Figure B.2. Rimoldi's encoder.*

Yang and Taylor's encoder reverses the order of the blocks of Rimoldi's encoder. Instead of coding first, the binary data is mapped to a 4-ary sequence which is subsequently encoded. Another subtle but important difference is that encoding is done over rings [46].

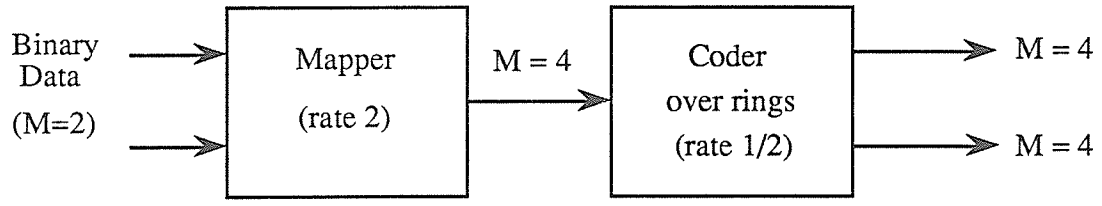


*Figure B.3. Yang and Taylor's encoder.*

It should be noted that both of the above encoders were applied specifically to CPFSK (L=1), so that no partial response bandwidth shaping (in the sense of Figure B.1) ought to occur; this is indeed the case for Rimoldi's encoder. However, in the case of Yang and Taylor's encoder (abbreviated by Y+T), the ring structure causes changes in the frequency weight distribution. Thus, in the ensuing discussion we concentrate on Y+T's encoding structure. To illustrate these ideas, consider a specific example.

### A Good Code For CPFSK

Table 2 in [46] lists several of the best codes found for the case of 4-ary, h=1/4 CPFSK. The simplest encoder listed there is a 4 state encoder with a 3.15 dB power gain over MSK. This encoder is shown in Figure B.4 (note that the mapping from binary to quaternary data has already been achieved; since the input binary data is assumed i.i.d. with a uniform distribution, the structure of this mapper is not important). All operations take place over the ring of integers modulo 4. The output symbols of the encoder are fed into the CPFSK modulator and effectively choose the frequency (labeled as in Figure B.1) which is to be transmitted. The combination of this convolutional encoder with the CPFSK modulator results in 4 states [46].

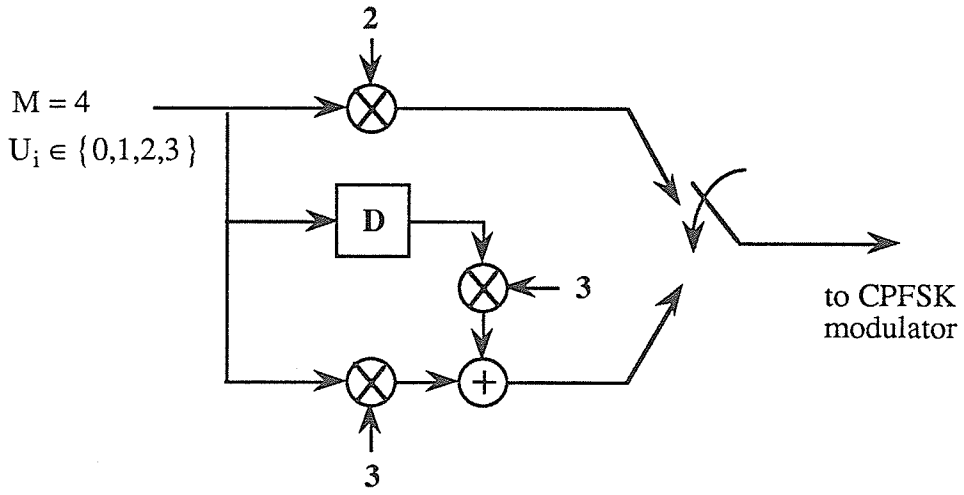


*Figure B.4. A good encoder for M=4, h=1/4 CPFSK (operations over the ring of integers mod 4).*

An interesting effect occurs with this encoder. Because the first encoded symbol of every pair of output symbols is multiplied by 2, this first symbol can only take on the values 0 and 2. The reason for this is that the first multiplier effectively forms the ideal {0,2} in the ring of integers modulo 4. In contrast, the second data symbol can take on any of the values 0,1,2 or 3. The impact of this structure is that, if the coded data symbols are mapped onto the four frequencies as in Figure B.1, the frequency weight distribution becomes as in Figure B.5. For the ensuing discussion, we abbreviate this distribution as (3,1,3,1).

The (3,1,3,1) frequency distribution will have a different PSD and bandwidth than the M=4, h=1/4 scheme. For one thing, the PSD will become skewed to a lower frequency and may actually require less bandwidth (based on calculations for an M=4, h=1/4 scheme with an input data distribution of (3,1,3,1) the bandwidth *is* actually smaller [47]; however, this calculation is not completely accurate since the coded symbols are not independent). Another factor which will affect the bandwidth is the size of the frequency hops; for the encoded scheme discussed here the maximum frequency hop allowed is ± 4. Nevertheless, if the bandwidth *is* actually smaller than the M=4, h=1/4 CPFSK scheme, then both power-*and* bandwidth-performance improvement has been achieved.

The type of frequency distribution shaping which goes on here applies generally for encoders over rings if ideals are generated by one of the encoder outputs. For example, if M=6 then a multiplication by two would generate the ideal {0,2,4} and a multiplication by 3 would generate the ideal {0,3}. Rings other than those over the integers modulo some composite number can, of course, also have this property. However, this effect does not occur for codes over fields (such as the integers modulo a prime).

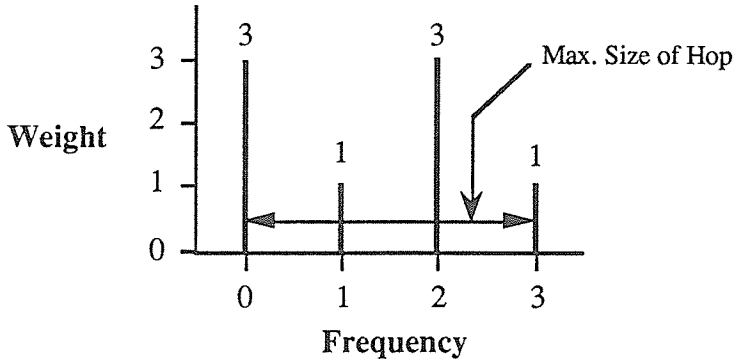


*Figure B.5. Frequency weight distribution for the encoder of Figure B.4.*

We concentrate on rings over the integers modulo some composite number, c. When ideals are generated for such rings, they will occur in multiples of one of the divisors of c, labeled d. Because the ideals are staggered (Ie. {0,d,2d,...,(c-d)}), the frequency distribution will also be staggered and not "bell-shaped" as for partial response CPM (Figure B.1). Staggered distributions will tend to have a larger bandwidth than the binomial distributions. This can be corrected by using a different symbol-to-frequency mapper than the one used for the encoder above. For example, by mapping the 0's to frequency #1 and the 1's to frequency #0, the (3,1,3,1) frequency distribution will become (1,3,3,1) and the resulting PSD will be similar to that of 3REC. In this way, the combination of the encoder over rings and the second mapper performs a bandwidth shaping of the CPM signal. Of course, this combination's error performance is not as good as the original scheme's.

The above discussion has shown that both the error and bandwidth performance must be taken into account when designing encoders for CPM. It may, in fact, be possible to find combinations of encoders and mappers which simultaneously yield good error *and* bandwidth performance. This and other avenues which can be explored are:

1) The above has been a rather ad hoc discussion on the bandwidth of coded CPM schemes. A more general and rigorous theoretical analysis should be performed; first and foremost, the assumption that coded CPM schemes with equal frequency weight distributions have similar bandwidths needs to be verified.
2) The above ideas have counterparts in the area of shaping gain for constellations[1], and it would certainly be advantageous if the power gain and (bandwidth) shaping of the codes described above could be separated in the same way as is done for constellation codes. This, however, does not seem possible since the frequency weight distribution and coding gain are directly related.
3) The discussion about the Y+T encoder was applicable to full response CPM modulators (CPFSK here). The results can be generalized to partial response CPM.
4) Rings other than the ones over the integers modulo some composite number can be used (even non-commutative rings).
5) Convolutional encoders over groups could be investigated[2].

---

[1] G.D. Forney, "Trellis shaping," *IEEE Trans. Inform. Theory*, vol. 38, pp. 281-300, March 1992.

[2] H-A. Löliger, "On Euclidean-Space group codes," D.Sc. dissertation, Swiss Federal Institute of Technology, Zürich, 1992.

# APPENDIX C. PROGRAM LISTINGS

Software written to:

i) calculate the size of the dynamic signal sets,
ii) calculate the minimum squared Euclidean distances for optimal and suboptimal detection,
iii) calculate the PSD's and out-of band power,
iv) implement the Stack Algorithm in AWGN and ACI, and
v) implement the Viterbi- and M-Algorithms in AWGN and ACI,

can be obtained by contacting:

Professor Ed Shwedyk
Department of Electrical & Computer Engineering
University of Manitoba
Winnipeg, Manitoba, Canada
R3T 2N2

All programs were written in C.