# RATE COMPATIBLE TRELLIS CODES
# FOR
# ADAPTIVE ERROR CONTROL COMMUNICATION SYSTEMS

by

Yan Wu

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE (ENGINEERING SCIENCE)
in the School of
Engineering Science


Simon Fraser University

March 1990

# APPROVAL

NAME : Yan Wu

DEGREE : Master of Applied Science ( Engineering Science )

TITLE OF THESIS : Rate Compatible Trellis Codes for Adaptive Error Control Communication Systems

EXAMINING COMMITTEE :

Chairman : Dr. R. H. S. Hardy

Dr. P. K. - M. Ho
Senior Supervisor

Dr. Vladimir Cupérman
Supervisor

Dr. James K. Cavers
Examiner

DATE APPROVED : March 19th, 1990

## PARTIAL COPYRIGHT LICENSE

I hereby grant to Simon Fraser University the right to lend my thesis, project or extended essay (the title of which is shown below) to users of the Simon Fraser University Library, and to make partial or single copies only for such users or in response to a request from the library of any other university, or other educational institution, on its own behalf or for one of its users. I further agree that permission for multiple copying of this work for scholarly purposes may be granted by me or the Dean of Graduate Studies. It is understood that copying or publication of this work for financial gain shall not be allowed without my written permission.

Title of Thesis/Project/Extended Essay

"Rate Compatible Trellis Codes for Adaptive Error Control Communication Systems"

Author: _____
(signature)

Yan WU
(name)

April 10, 1990
(date)

## ABSTRACT

To deal with the increasing demand of mobile data communications, existing error control schemes of mobile communication systems should be modified to make more efficient use of the available resources, such as power and bandwidth. This thesis provides an appealing way to achieve the above objective by integrating coded modulation design and a generalized hybrid ARQ/FEC error control protocol design into one single process. For a given bandwidth, our proposed systems provide effective unequal error protection over a wide range of signal to noise ratio, and a maximum throughput efficiency of several bits per channel symbol, while similar existing techniques can only accommodate a maximum information rate of 1 bit per channel symbol. For mobile radio applications, the analytical and computer simulation results show that the systems using our new approach yield up to 80 % overall system throughput gain over the best known similar system proposed by Hagenauer with comparable decoding complexities. A detailed performance analysis, and computer simulation results of the systems using the new technique are included in the thesis.

## ACKNOWLEDGEMENTS

I would like to thank my supervisor, Dr. Paul Ho, for introducing me to the topic and much of the material of this thesis. Moreover, his guidance, support, scholarly criticism and patience have been sincerely appreciated throughout the course of my research.

The financial support from the Natural Sciences and Engineering Research Council of Canada is gratefully acknowledged.

# Table of Contents

# List of Figures

# List of Tables

## ABBREVIATIONS

ACK - positive acknowledgement

ARQ - automatic repeat request

AWGN - additive white Gaussian noise

BPSK - binary phase shift keying

BSC - binary symmetric channel

CPM - continuous phase modulation

CRC - cyclic redundancy check

CSI - channel state information

dB - decibel

Ed - Euclidean distance

FEC - forward error correction or forward error correcting codes

FER - frame error rate

GBN - go back N

Hd - Hamming distance

i.i.d. - independent and identically distributed

MTCM - multiple trellis coded modulation

NAK - negative acknowledgement

PSD - power spectral density

PSK - phase shift keying

QAM - quadrature amplitude modulation

QPSK - quadrature phase shift keying

RCPCC - rate compatible punctured convolutional code

RC-TCM - rate compatible trellis coded modulation

RHP - right hand (complex) plane

SNR - signal to noise ratio

SNW - stop and wait

SRT - selective repeat

TCM - trellis coded modulation

## VARIABLES AND FUNCTIONS

$a_k$ - the amplitude of $g_k$, a Rayleigh distributed random variable

b - the number of input bits to the encoder at each encoding interval

$c_i$ - $i$th coded symbol in each puncturing period

$c_{ik}, c_{jk}$ - complex coded MPSK symbols with a unity normalized magnitude

$C_d$ - an $(n_1, k)$ error detecting code

$d_E$ - Euclidean distance between two codewords

$d_{fH}$ - free Euclidean distance

$d_H$ - Hamming distance between two codewords

$d_{fH}$ - free Hamming distance

$d_{\min}$ - the minimum Euclidean distance for a uncoded system

$d_k^2$ - squared Euclidean distance between two symbols in the signal space

$\delta_i$ - path metric

$\prod d^2$ - the product of squared branch Euclidean distances

$E_b$ - bit energy

$E_s$ - symbol energy

$E[\cdot]$ - statistical average

$f_c$ - carrier frequency

$\phi_k$ - the phase of $g_k$, a uniform distributed random variable

$g_k$ - a complex Gaussian random variable

g(t) - channel fading process

k - number of information bits contained in a codeword

K - the maximum number of decoding attempts for each packet

M - modulation level

$\mu$ - multiplicity of a MTCM

n - codeword length of a linear block code

$n_1$ - codeword length of an (outer) error detecting code

n(t) - channel AWGN

$n_k$ - filtered channel AWGN term in the $k$th signalling interval

$n_i$ - number of information bits per packet

$n_{av}$ - average number of bits needed to deliver $n_i$ information bits

$n_c$ - number of CRC bits per packet

$n_t$ - packet length

$N_s$ - number of encoder states

N - number of users in a cellular cell

$N_0$ - twice of the power spectral density of channel AWGN

P - puncturing period

$P_b$ - bit error probability

$P_{b_i}$ - bit error probability after the $i$th decoding attempt

$P(i \rightarrow j)$ - pairwise error event probability

$P_{C_i}$ - the successful probability of the $i$th decoding attempt

$P_{F_i}$ - the failure probability of the $i$th decoding attempt

$Q(\cdot)$ - the Q - function

$r$ - the distance between a transmitter and a receiver

$r_k$ - received sample for the $k$th symbol in the codeword sent

$r_{\max}$ - the radius of a circular cellular coverage area

$R_{av}$ - average throughput efficiency

$R_0$ - cut off rate

$R_{\max}$ - maximum effective FEC code rate

s - any number in the complex plane

$S_{av}$ - the average number of coded modulation symbols needed for delivering each information packet of $n_t$ bits

$S_i$ - the number of coded modulation symbols accumulated at the receiver for the $i$th decoding attempt

$S(t)$ - signaling waveform

$\sigma_g^2$ - variance of g(t)

t - time

T - duration of the signaling interval

v - the number of bits for terminating the decoder trellis

$x$ - information sequence of length $n_t$

$x_i$ - the $i$th bit in the information sequence $x$

$X$ - decoding complxity

# CHAPTER ONE

# INTRODUCTION

A major concern in data communications is how to control transmission errors caused by channel noise so that data can be reliably delivered to the users. One solution to this problem is the application of Forward Error Correction (FEC) coding. Another possible solution is to use Automatic Repeat Request (ARQ), an error control technique that uses error detecting code only. However, for channels with nonstationary noise and interference, such as a mobile radio channel, one has to look into some more robust error combating techniques such as the type II hybrid ARQ/FEC technique where an appropriate combination of FEC and ARQ is used. So far, a multitude of variations based on this technique, including several rate - adaptive systems, have been examined. Conventional FEC code design method (binary block code or convolutional code, optimally designed in the sense of maximizing the free Hamming distance of the code, in conjunction with binary modulation) are employed in these studies. Therefore, the throughput of these communication systems are restricted to a maximum of 1 information bit per channel symbol. In this thesis, we report an adaptive hybrid ARQ/FEC technique that uses joint power and bandwidth efficient coded modulation techniques. As will be shown later on, using coded modulation allows us to achieve much improved throughput performance, and its adaptivity makes this technique rather suitable for mobile radio applications.

## 1.1 BACKGROUND AND LITERATURE REVIEW

As mentioned earlier, to meet the requirement of reliability and throughput efficiency for various digital communication systems, two basic categories of error control strategies have been developed : Forward Error Correction (FEC) schemes and Automatic Repeat Request (ARQ) schemes. In a FEC system, an error correcting code is used for combating transmission errors. Since there are no retransmissions in a FEC system, the information throughput is a constant and equal to the code rate of the FEC, and is independent of the channel quality. However, if an error pattern beyond the error correcting capability of the code occurs, the received codeword will be decoded incorrectly and the erroneous data will be delivered to the user. Incorrect decoding occurs more often when the channel is very noisy as there is a higher chance of getting uncorrectable error patterns. In principle, one can use a low rate codes with long codeword length to reduce the chance of incorrect decoding when the channel is noisy. However, it is hard and expensive to implement decoders for long powerful error correcting codes, therefore system designers conclude that it is very difficult to achieve high data reliability with FEC alone. Therefore ARQ schemes are often preferred over FEC for error control in data communication systems, except for some systems where feedback channels are not available or retransmission is not possible for some reasons.

Compared with FEC schemes, ARQ schemes, in general, can provide a much higher reliability, and it is easy to implement. In a pure ARQ system, only an error detecting code is used. At the receiving end, an error detection algorithm ( parity check or syndrome computation ) is performed. Depending on whether errors have been detected or not, a negative ( NAK ) or a positive ( ACK ) acknowledgement

signal is sent to the transmitter, via a return channel. Upon receiving the acknowledgment signal, the transmitter will retransmit the block in error ( in the case of an NAK ) or transmit the next block ( in the case of an ACK ) in the queue. In this system, codewords with errors are delivered to the users only if the receiver fails to detect the presence of errors. It is well known that if an ( n, k ) linear block code is used for error detection only, then, the undetected error probability is always less than the decoding error probability when the same code is used for forward error correction [34], [35]. This can be observed clearly from the standard array of a ( n, k ) linear block code shown in Table 1.1.

| $C_1 = 0$ | $C_2$ | $C_3$ | . . . | $C_{2^k}$ |
|---|---|---|---|---|
| $e_2$ | $e_2 + C_2$ | $e_2 + C_3$ | . . . | $e_2 + C_{2^k}$ |
| . | . | . | . | . |
| . | . | . | . | . |
| . | . | . | . | . |
| $e_{2^{n-k}}$ | $e_{2^{n-k}} + C_2$ | $e_{2^{n-k}} + C_3$ | . . . | $e_{2^{n-k}} + C_{2^k}$ |

Table 1.1 Standard array.

In the above table, the first row contains all the codewords, denoted by the $C_j$'s, $j = 1, 2, \ldots, 2^k$. The remaining rows are the cosets of the code. The first element of each coset $e_i$, $i = 2, 3, \ldots, 2^{n-k}$ is called the coset leader. From coding theory, if a

code is used for error detection, the undetected error probability is the probability that the noise pattern is one of the vector in the first row. On the other hand, if the same code is used for error correction, the probability of an uncorrectable error is the probability that the error is not a coset leader, which includes the set of codewords in the first row. So, the undetectable error probability must be lower than the uncorrectable error probability. Thus, as far as the reliability is concerned, a pure ARQ scheme which uses only an error detection code is far superior to a pure FEC scheme which uses only an error correction code. The idea of using ARQ strategies can be traced back to the 1940's [3]. Since then various ARQ schemes have been proposed and implemented, the best known are the Stop-And-Wait (SNW) scheme, the Go-Back-N (GBN) scheme and the Selective-Repeat schemes (SRT). As the simplest retransmission strategy, the SNW scheme has been widely used. In such a system, the transmitter sends a codeword to the receiver and waits for an acknowledgment from the receiver, as can be seen from the diagram in Fig. 1.1.

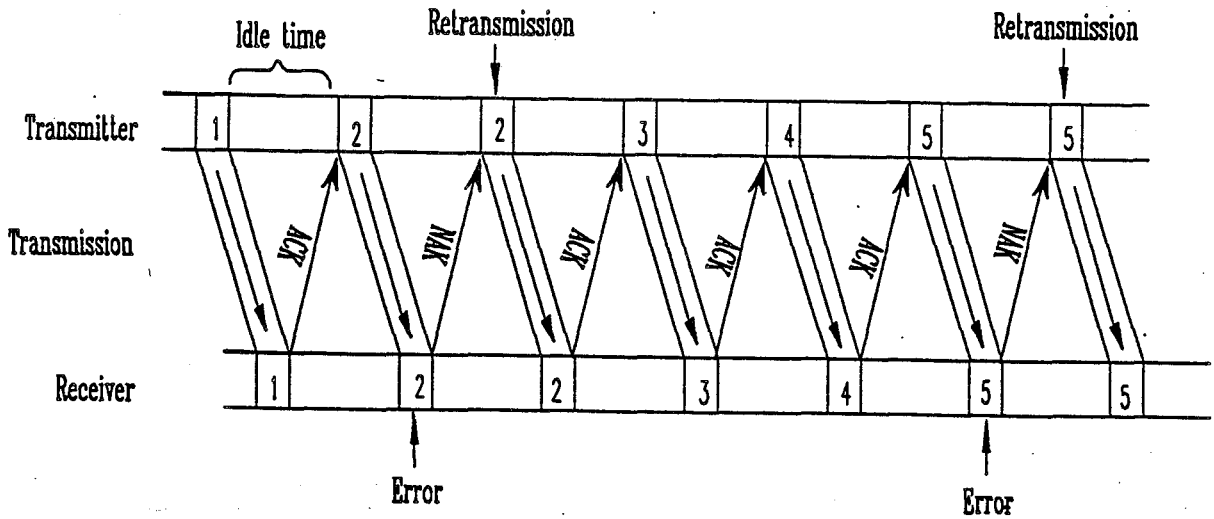



Fig. 1.1 Stop - and - wait ARQ.

As shown in Fig. 1.1, if an ACK is received, the transmitter sends the next codeword, otherwise it retransmits the codeword detected in error and again waits for an acknowledgment. This procedure will continue until the transmitter receives an ACK. The SNW scheme has a nature of inefficiency because of the idle time spent on waiting for an acknowledgment about each transmission. For some systems with small propagation delay, such as the mobile radio communication system considered in this study, the idle time has a negligible effect on the efficiency. A detailed description of the principle and the features of a SNW ARQ scheme and the other two types of ARQ systems can be found, for instance, in [4].

As mentioned earlier, the high reliability achieved by ARQ schemes is a result of retransmitting those data blocks found in error by the receiver. Consequently, as the channel error rate increases, pure ARQ schemes yield lower and lower throughput. This is a severe drawback of any pure ARQ system.

A class of hybrid schemes involving a proper combination of FEC and ARQ techniques, namely hybrid ARQ/FEC schemes, in principle, can overcome the drawbacks of both pure ARQ and pure FEC systems. The fact is that a FEC subsystem contained in a hybrid system can effectively reduce the frequency of retransmission by correcting the error patterns which occur more frequently so that a better system throughput than in a pure ARQ system is obtained. On the other hand, when a less frequent error pattern occurs and is detected, the receiver requests a retransmission rather than passing the unreliable decoded message to the user so that a better reliability than in a pure FEC system is obtained. Therefore, many efforts have been devoted to studying hybrid ARQ/FEC systems [6], [7]. Hybrid ARQ/FEC techniques are further classified into type I and type II schemes [1], [4]. In a type I hybrid ARQ/FEC

system, a code is designed for simultaneously correcting and detecting errors. This results in a requirement of significantly more redundant bits than a code used only for error detection. These parity check bits for error correction are sent no matter what the channel condition is. Because of this characteristic, see Fig. 1.2, a type I hybrid system has a lower throughput than a pure ARQ system in a low error rate channel but a higher throughput in a high error rate channel. For this reason, type I hybrid scheme is more suited for channels with an predictable fairly constant, medium to high bit error rate. Here the throughput is defined as the ratio of the number of information bits to be delivered to the average number of transmitted bits for delivering these information bits. If the bit error rate of the channel varies over a wide range, a type I system could be inefficient, due to its low throughput at low bit error rate conditions. The performance of some type I hybrid schemes, using either block codes, or convolutional codes, are reported in [6]-[9].

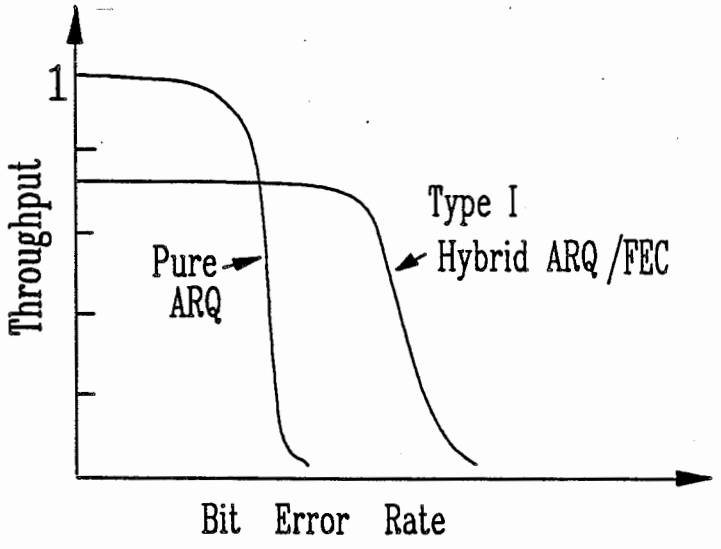



Fig. 1.2 The throughput comparison of pure ARQ and type I hybrid ARQ/FEC schemes.

To avoid the disadvantage of the type I schemes, i.e. the extra FEC parity bits are wasted under good channel conditions, type II hybrid systems were suggested, see for example, [10], [1], and [7]. This class of hybrid ARQ/FEC schemes is based on the concept of parity retransmission for FEC first introduced by Mandelbaum [5]. The basic concept of type II hybrid ARQ/FEC schemes is that the parity check bits for error correction are sent to the receiver only when they are needed. This makes a type II system quite adaptive, as oppose to a type I system. For some type II hybrid ARQ/FEC system in the literature, a codeword is divided into two parts. The first part contains k information bits and $n_c = n_1 - k$ error detection check bits or cyclic redundancy check (CRC) bits. These $n_1$ bits are obtained by passing the k information bits to a $(n_1, k)$ error detecting code. The second part consists of $n - n_1$ parity bits for error correction. These check bits are obtained by passing the $n_1$ bits in the first part into a $(n, n_1)$ error correcting code, see Fig. 1.3.

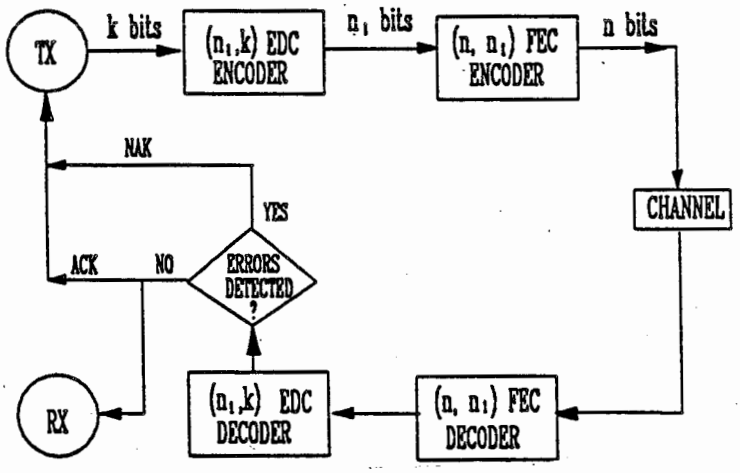



Fig. 1.3 A type II hybrid ARQ/FEC system.

Only the first part of each codeword, i.e. k information bits along with $n_c$ CRC bits, will be sent in the first transmission. If the receiver detects the presence of errors in the received data block, it saves the erroneous data block (the first part of the codeword) in a buffer and, at the same time, it notifies the transmitter to send the second part of the codeword which contains the FEC parity bits. Upon receiving the FEC parity block, the receiver then performs error correction based on the ( n, $n_1$ ) forward error correcting code, followed by error detection based on the ($n_1$, k) error detecting code, $C_d$. If errors are still detected, the receiver will discard the erroneous information block and save the FEC parity block in the buffer, and ask for a retransmission of the information block. Upon receiving the retransmitted information block, the receiver will again perform error detection based on $C_d$. If errors are detected, the receiver will combine the retransmitted information block and the FEC parity block stored in the buffer, and perform error correction based on ( n, $n_1$ ) code. If errors are again detected, the receiver will discard the FEC block and save the information block in the buffer, and ask for a retransmission of the FEC parity block, and then perform error correction based on ( n, $n_1$ ) code. If errors are still detected, the next retransmission will be the information block. This procedure of alternating between message bits (along with error detecting parity bits) in one retransmission and FEC parity bits on the next will be continued until the received word is error-free, or until a certain limit set by designers is met. Because of this message - parity alternating retransmission strategy, a type II system can overcome partially the drawbacks of a type I system.

To make a type II hybrid ARQ/FEC system even more efficient, invertible codes, which have the property that any error-free copy of the FEC parity block of the

codes can also deliver the message, have been suggested for use in type II hybrid ARQ/FEC systems [1], [4]. A system that uses an invertible code still follows the message - parity alternating retransmission strategy. For example, consider the system proposed by Lin et. al. [1].

In Lin's system, the first step of transmission is the same as that of the system given in Fig. 1.3. The k information bits and the $n_1 - k$ error detecting parity check bits are sent. At the receiver, the error detection is performed based on the code $C_d$. If errors are detected, the erroneous information block is saved in the receiver buffer and a NAK is sent to the transmitter. Similar to the system in Fig. 1.3, in this system, a FEC parity block of $n - n_1 = n_1$ bits will be sent upon receiving the first NAK. The difference is that this FEC parity block is itself a code vector of the code $C_d$, and the k information bits can be recovered from this FEC parity block by a simple inversion operation. Upon receiving the FEC parity block, the syndrome is calculated based on the code $C_d$. If the syndrome is zero, the FEC parity block is assumed to be error free, and the k information bits can be recovered from the FEC parity block by an inversion process. If the syndrome is not zero, the erroneous parity check block and the erroneous information block ( stored in the receiver buffer) are combined to perform error correction based on the $( 2n_1, n_1 )$ invertible code. If errors in the code vector of this invertible code form a correctable error pattern, they will be corrected. The decoded information block is then successfully delivered to the data sink. If the error pattern is not correctable but detectable based on $C_d$, the erroneous information block is discarded and the FEC parity block is stored in the receiver buffer, also, a NAK is sent to the transmitter. The transmitter then sends the information block again. If no error is detected from this retransmitted information

block (with the assumption that the undetectable error probability is zero), the erroneous parity block is discarded from the receiver buffer and the error - free information is delivered to the data sink. If errors are detected, the retransmitted information block together with the parity block are used for error correction based on the $( 2n_1, n_1 )$ invertible code. After error correction, if errors still exist, the erroneous FEC parity block will be discarded and the information block will be saved in the receiver buffer. The next retransmission will be the FEC parity block. This procedure of alternating between message packet in one retransmission and parity block on the next will be continued until the received word is error-free, or until a certain limit set by the designers is met.

From the above description, one can see that by using the invertible codes, the procedure of message recovery can be simplified and also, the number of retransmission required can be reduced. During a retransmission, if the FEC parity check block is successfully received, the message can then be reconstructed from the parity block by a simple inversion process rather than a more complicated decoding process. This implies no matter how many errors contained in the previous transmitted message block, no further retansmission will be needed.

A graph which combines the throughput characteristics of type I, type II hybrid ARQ/FEC schemes and a pure ARQ scheme is given in Fig. 1.4. Note that the type I and the type II schemes in Fig. 1.4 use the same forward error correcting code. Fig. 1.4 reveals that when the channel error rate is low, the FEC parity check blocks of a type II system need not be transmitted so that the effective code rate remains close to unity. In other words, a type II system behaves just like a pure ARQ system and thus it has a higher throughput than the type I system using the same forward

error correcting code. When the channel condition deteriorates, most of the time, the FEC parity block will be requested and sent to form a forward error correcting code at the receiver. Hence at high channel error rate, the throughput performance of a type II system is similar to that of a type I system. As stated earlier, a type II system removes the disadvantage of a type I system by transmitting the parity bits only when they are needed.

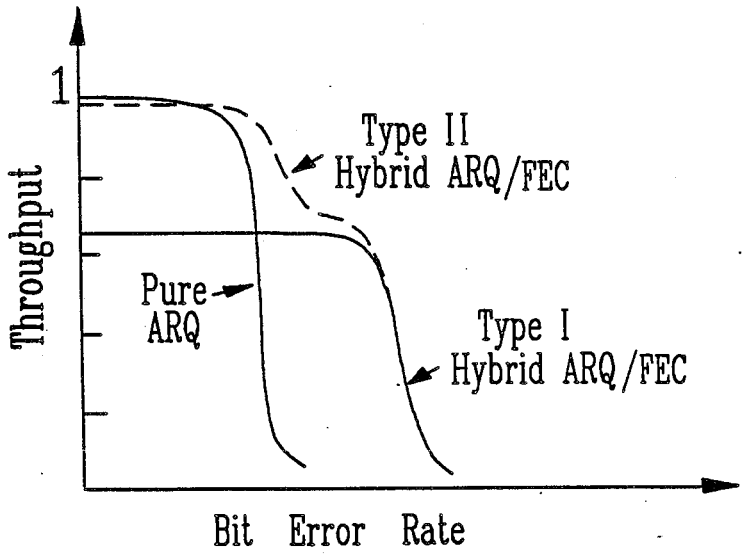



Fig. 1.4 Throughput comparison of pure ARQ, type I and type II hybrid ARQ/FEC systems.

From Fig. 1.4 we can see that there is an abrupt transition on the throughput curve of a type II system, due mainly to its limited adaptive capability. One of the main objectives of this thesis is to look for more adaptive hybrid ARQ/FEC systems with

throughput curves degrading gracefully from good channel condition to the very bad condition. We call this class of error control strategy generalized type II hybrid ARQ/FEC schemes.

Recently, several generalized type II hybrid ARQ/FEC schemes have been proposed [21],[14],[15] to allow the code rate and hence the error correction capability of the FEC to be varied over a wide range. The basic principle behind all these schemes is that the parity block of the FEC is divided into several sub - blocks and a different sub - block will be sent in each retransmission. At the receiver side, the decoder makes use of *all* previous received sub - blocks for carrying out error correction. This type of schemes are defined as *generalized* type II schemes, since in a conventional type II system, only the data block received from the very last transmission is saved. In [21], Chase introduced the code combining technique for channels with unpredictable conditions. In the context of our present discussion, Chase's code combining technique is equivalent to a generalized type II hybrid ARQ/FEC scheme that uses repetitive codes. In [15], Krishna proposed a new class of codes, the KM codes, for use in generalized type II hybrid systems. A KM code includes two linear block codes: an $(n_1, k)$ error detecting code, $C_d$, and a $(mn_1, n_1)$ invertible code which is actually a generalization of the $(2n_1, n_1)$ invertible code in [4]. The generator matrix of this $(mn_1, n_1)$ code is selected in a way that it can be partitioned into m subblocks each of dimension $(n_1 \times n_1)$. In other words, here we have m-1 FEC parity blocks of $n_1$ bits each. The information block can be obtained by an inversion process on any one of these m-1 parity blocks. One parity block is sent at each retransmission, and all the parity blocks received at the receiver are combined to create an effective error correcting code. This implies that the effective rate of the FEC can be 1/2,

1/3, ..., 1/m. As the code rate decreases, the error correcting capability of the code increases, thereby leading to an improved performance under poor channel conditions. Applying soft decision decoding to this coding scheme is suggested by Morgera et. al. [16].

Another generalized type II hybrid ARQ/FEC scheme proposed by Hagenauer [13],[14], uses Rate Compatible Punctured Convolutional (RCPC) codes instead of using block codes. RCPC codes are more flexible than the code combining technique of Chase and the KM codes of Krishna, from the point of view that the code rate can be changed in smaller steps and that a maximum likelihood soft decision decoder can be implemented easily using the Viterbi algorithm. The concept of punctured convolutional codes is first introduced by Cain et. al. [12]. They found that a high rate convolutional code can be obtained by deleting periodically encoded bits of a mother code. As such, the trellis of the mother code can be used for the decoding of the punctured code. In this way, the decoding complexity is reduced significantly as compared to that required in a regular high rate convolutional code. To apply punctured convolutional codes in an adaptive error control system, Hagenauer introduces a rate-compatibility restriction on the puncturing rules so that all code bits of a high rate punctured code are used by the lower rate codes. When RCPC codes are used with the adaptive hybrid ARQ/FEC protocol, it is quite flexible to choose the number of incremental redundancy bits transmitted in each step. Therefore, Hagenauer's proposal can attain a smoother throughput curve, which is desirable in many cases. A detailed description of RCPC codes and their application in an adaptive error control system will be given in the next chapter.

So far, all the error control schemes reviewed above use binary coding schemes followed by binary modulation schemes. This limits the system throughput to, at most 1 bit per channel symbol. These traditional channel coding techniques achieve coding gain at the cost of bandwidth expansion. In the past decade, the search for bandwidth and power efficient digital modulation techniques is a very active research area. As a result, two types of bandwidth efficient modulation schemes are proposed: Continuous Phase Modulation ( CPM ) [37] and Trellis - Coded Modulation (TCM) [23], [24]. In [37], Aulin et. al. considered M-ary digital FM (frequency modulation) with smoothing of the instantaneous frequency, which is referred to as CPM (continuous phase modulation). An ARQ scheme that includes the modulation design based on CPM can be found in [36]. On the other hand, Ungerboeck [23], [24] has developed a coding technique known as Trellis Coded Modulation (TCM). He combines coding and modulation in such a way that the noise immunity is increased without increasing the signal power and without bandwidth expansion. This is achieved by using redundant *nonbinary* modulation in conjunction with a finite state encoder. The assignment of modulation signals to the encoder transitions is based upon a concept that Ungerboeck called mapping by set partitioning, a mapping technique that will result in a large Euclidean distance between signal sequences. Using this mapping, significant coding gain of 3-4 dB in bandlimited additive white Gaussian noise (AWGN) channels with respect to the uncoded modulation system of the same data rate, can be obtained by simple, hand designed TCM.

Following Ungerboeck's pioneer work, extensive studies on TCM schemes were carried out. Recently, there are a lot of interest in applying TCM schemes in fading channels, see for example [25 - 31]. In [25], Divsalar and Simon proposed a new

class of TCM schemes which they called Multiple Trellis Coded Modulation (MTCM). These TCM schemes have multiple symbols per encoder transition, which as shown by Divsalar and Simon, provides an additionally degree of freedom to design codes that satisfy the optimum code design criteria for fading channels. As presented in [25], in order to minimize the error probability in a fading channel, not only the code's free Euclidean distance is important, but also the free Hamming distance is crucial as it directly relates to the diversity of the code. The larger the free Hamming distance, the faster the error probability decreases with an increase in the channel's signal to noise ratio. In this thesis, we will modify some MTCM schemes of [26] and apply them to an adaptive hybrid ARQ/FEC system. As in convolutional coding, a MTCM scheme has multi - symbol per trellis branch. If we puncture it in a similar way as Hagenuaer did in puncturing convolutional codes, then a promising improvement in the throughput performance can be obtained without increasing the bandwidth. This new idea will be discussed in detail in the later chapters.

## 1.2 THESIS OUTLINE

The outline of the thesis is as follows. In Chapter 2, the concept of the RCPC codes are described. The performance of a hybrid ARQ/FEC system that uses RCPC codes are given, including the throughput efficiency and the frame error rate. In Chapter 3, a new method, which integrates the generalized type II hybrid ARQ/FEC error control protocol design and the bandwidth efficient trellis coded modulation design into a single process, is discussed. The performance of systems using this new error control technique, which we called RC-TCM ( Rate Compatible Trellis Coded Modulation ), are given in Chapter 4. We also compare in Chapter 4 the

performance of our proposed systems with those that use RCPC codes. In both cases, we consider only the Rayleigh fading channel, which is an appropriate model for mobile radio channels. Finally, in Chapter 5, a summary of the results obtained in the present investigation are made and the suggestions for future study are discussed.

## 1.3 CONTRIBUTIONS OF THE THESIS

As the readers will see later, our proposed error control technique based on RC-TCM, when used in a Rayleigh fading channel, allows us to achieve a throughput that is much higher than any existing techniques in the literatures. This improvement is due to the fact that we integrate the error control protocol and the coding / modulation design into a single process. This is the main contribution of the thesis.

# CHAPTER TWO

# ADAPTIVE HYBRID ARQ/FEC PROTOCOLS USING RCPC CODES

We discuss in detail in this chapter a channel coding technique that is suitable for use in adaptive hybrid error control systems. It is the rate compatible punctured convolutional (RCPC) codes in [14]. The concept discussed in this chapter provides all the information required to understand the new adaptive hybrid ARQ/FEC schemes introduced in Chapter 3. Examples will be given in this chapter to help the explanation. The system performance, including the throughput and the frame error rate, of a hybrid ARQ/FEC system that uses RCPC codes in Rayleigh fading channel will also be given.

## 2.1 CONCEPT AND AN EXAMPLE

The block diagram of a hybrid ARQ/FEC error control system using RCPC code is shown in Fig. 2.1.

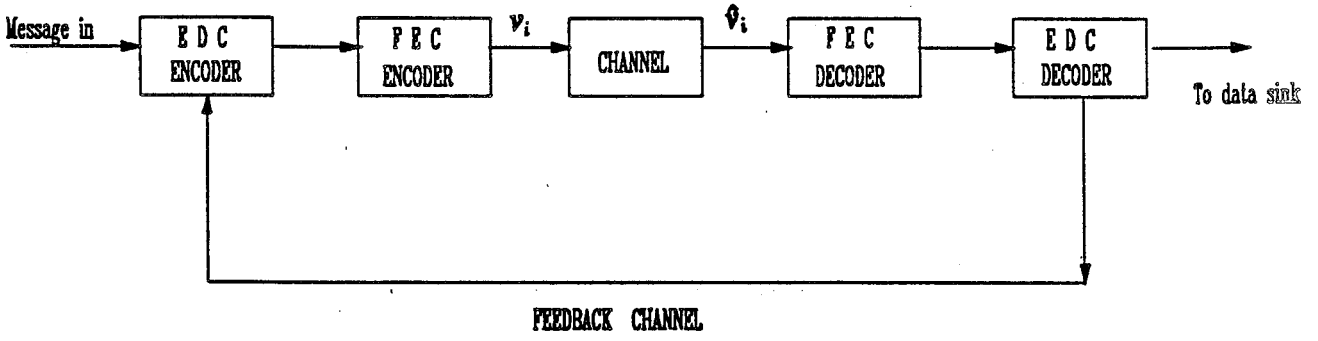



Fig. 2.1 Block diagram of an adaptive hybrid ARQ/FEC system. $v_i$ is the code vector sent in the $i$th retransmission and $\hat{v}_i$ is the corresponding received copy of $v_i$.

It consists of an error detecting code, an error correcting code, a forward channel and a feedback channel for sending acknowledgments from the receiver to the message sender. As mentioned earlier, an attractive property of a generalized type II hybrid ARQ/FEC system is its adaptive error correction capability in response to the change in channel conditions. This is achieved by making use of all the accumulated data received for the same packet during forward error correction as described below.

Let $C_0$ be the mother code with code rate $r_0 = k/n$. Here a mother code refers to a code from which a set of codes with different rates can be generated by puncturing its code symbols, k is the number of information bits in each codeword, and n is the length of the codeword. Here for convenience, the parity check bits for error detection are treated as part of the information bits of the error correcting code. Now assume that the n-k redundant bits of each codeword are partitioned into m subblocks $v_1, v_2, \ldots, v_m$ with length $l_1, l_2, \ldots, l_m$ bits respectively, see Fig. 2.2.

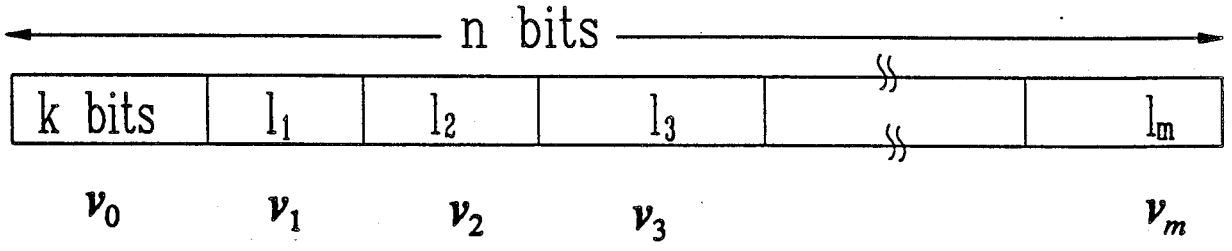



Fig. 2.2 A codeword of a mother code used in FEC. The first k bits includes information bits as well as error detection check bits.

In general, $l_1, l_2, \ldots, l_m$ are not necessarily equal. Now if we delete the last block $v_m$ from each codeword of $C_0$, we can obtain a code $C_1$ of rate $r_1 = k/\left(k + \sum_{i=1}^{m-1} l_i\right)$. Note

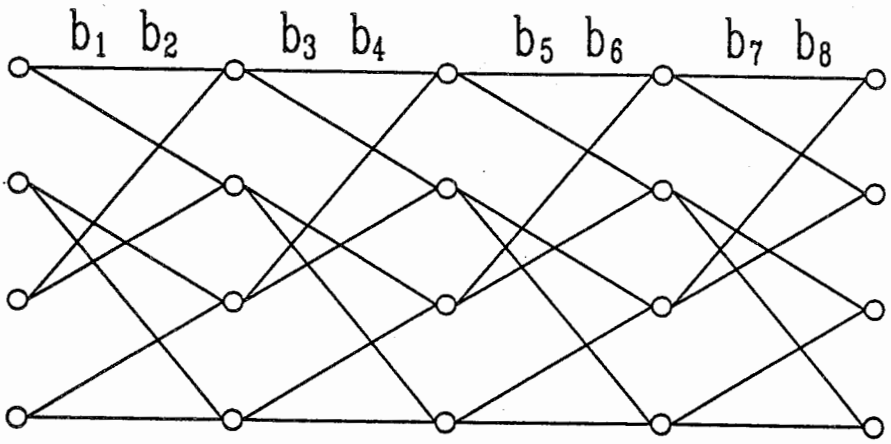
that $r_1 > r_0$. Similarly, If we delete the block $v_{m-1}$ from each codeword of the code $C_1$, code, we have a code $C_2$ of a rate $r_2 = k/\left(k + \sum_{i=1}^{m-2} l_i\right)$. Note that $r_2 > r_1$. If we continue this puncturing procedure, we can obtain a sequence of rate compatible codes,

$$C_0, C_1, \ldots C_m$$

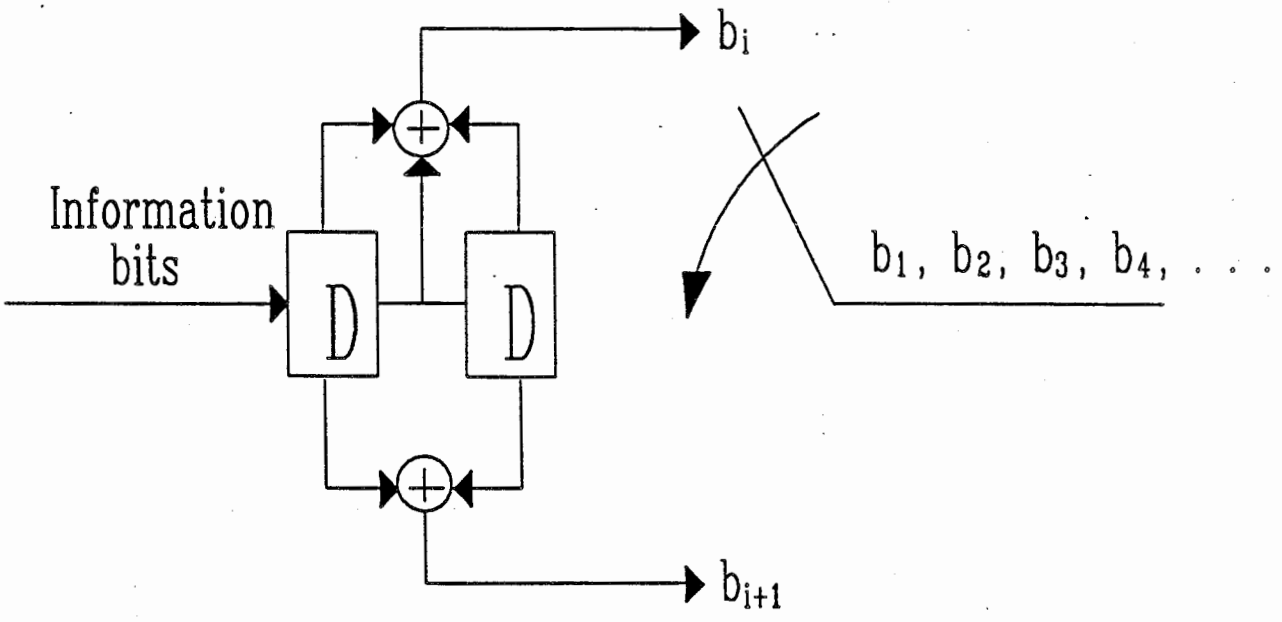
of rates $r_0, r_1, \ldots r_m$ in increasing order. Note that the code $C_m$ corresponds to the case where there is only error detection but no FEC.

This procedure of puncturing code symbols from a lower rate code to obtain a higher rate code suggests that one can vary the effective code rate of the FEC by adopting the following transmission and decoding strategy. During the first attempt, only the k information bits, denoted by the vector $v_0$ in Fig. 2.2, will be transmitted. Since in the first attempt, $v_0$ contains only error detection check bits, so the system is functioning in a pure ARQ mode, i.e. the receiver will detect whether errors are contained in the received word $\hat{v}_0$, where in the following discussion, $\hat{v}_i$ is used to represent the received copy of the transmitted word $v_i$, see Fig. 2.1. If errors are detected, the receiver will store $\hat{v}_0$ in a buffer and a negative acknowledgment (NAK) signal is sent to the transmitter. The transmitter will send the first subblock of redundant bits, $v_1$ ( with $l_1$ bits ), in the second attempt. Upon receiving $\hat{v}_1$, the receiver will perform error correction based on the concatenation of $\hat{v}_0$ and $\hat{v}_1$. The receiver is allowed to do so because the concatenation of $\hat{v}_0$ and $\hat{v}_1$ is actually a noise corrupted codeword of the code $C_{m-1}$ ( $C_{m-1}$ is obtained by deleting the last

m-1 subblocks of redundant bits from each codeword of the rate k/n mother code in Fig. 2.2 ). So although only $l_1$ bits are transmitted in the second attempt, the effective FEC code rate is $k/(k+l_1)$. If errors are still detected after the error correction, the receiver will store $\hat{v}_0$ together with $\hat{v}_1$ in the buffer and another NAK is sent to the sender. This will result in the next subblock of redundant bits, $v_2$, being transmitted in the third attempt. Again, after receiving $\hat{v}_2$, the receiver will perform forward error correction based on the concatenation of $\hat{v}_0$, $\hat{v}_1$ and $\hat{v}_2$, and the effective code rate of the FEC becomes $k/(k+l_1+l_2)$. This process will continue until there is no detectable error in the received packet or until the last block of redundant symbols has been sent. In the later case, if the decoding is still not successful, several possibilities exist : a repetition of the whole procedure described above; or code combining : repeating the whole codeword of the $r_0 = k/n$ code in subsequent attempts and combining all the received copies to form a repetition code. This is a technique suggested by Chase [21]. The procedure described above for obtaining rate compatible codes applies to both block and convolutional code. We are interested in this study in convolutional codes, or in general, codes with a trellis structure, due to the fact that maximum likelihood soft decision decoding using the Viterbi algorithm can be easily done. When a properly chosen convolutional code is employed as the mother code and when the mother code is punctured periodically according to the procedure described above, we obtain a sequence of rate compatible punctured convolutional codes [13, 14] that can be readily used in a generalized type II hybrid ARQ/FEC system. The following example illustrates how to obtain a sequence of RCPC codes.




(a)




(b)

Fig. 2.3 (a) Trellis diagram of a rate 1/2 convolutional code.

(b) A rate 1/2 convolutional encoder.

Fig. 2.3 (a) and (b) show the trellis diagram of a 4 state, rate 1/2 convolutional code and its encoder. Here nodes of the trellis represent the encoder states and the branches between the nodes represent the transitions. The label on each branch is the encoder output bits at each transition. We will use this code as the mother code. To obtain a sequence of RCPC codes, we need to specify a puncturing period P. For instance, in this example we use P=4. Subsequently, the time axis are divided into periods of 4 intervals. As shown in Fig. 2.3 (a), the encoder output within each period is denoted by the 8-tuple: $\boldsymbol{b} = (b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8)$, where $(b_i, b_{i+1})$, i odd, are the output of the rate 1/2 mother code encoder in each interval. We can delete 3, 2, 1 bits from $\boldsymbol{b}$ to form rate 4/5, 4/6, 4/7 codes respectively. Since a rate 1/2 mother code carries 1 bit of information per interval, thus intuitively, at least one coded bit should be kept in each interval during the puncturing process so as to avoid the loss of information. In other words, $b_i$ and $b_{i+1}$ can not be both deleted. Even with this constraint, there are still a large number of ways the mother code can be punctured. For purpose of demonstration, we choose to puncture the second bit in each interval in this example. Using this puncturing rule, the set of punctured codes are :

| | |
|---|---|
| rate 1/2 code | $b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8$ |
| rate 4/7 code | $b_1, b_2, b_3, b_4, b_5, b_6, b_7, \times$ |
| rate 4/6 code | $b_1, b_2, b_3, \times, b_5, b_6, b_7, \times$ |
| rate 4/5 code | $b_1, b_2, b_3, \times, b_5, \times, b_7, \times$ |

Table 2.1 A sequence of punctured convolutional codes.

where × denotes a punctured bit. Of course, there is no guarantee that this arbitrarily chosen puncturing rule will yield good rate 4/7, 4/6, and 4/5 codes. The best puncturing rule, in general, has to be obtained through an exhaustive computer search.

We have finished discussing the method used to generate the RCPC codes. Next we will demonstrate how to apply these codes in a type II hybrid ARQ/FEC error control system.

In the first attempt, the rate 4/5 punctured convolutional code specified in Table 2.1 is used, i.e. the first, second, third, fifth and seventh bit in each period ( of 4 intervals ) of the mother code will be sent. Upon receiving the noise corrupted copy, the receiver will first perform the Viterbi decoding of the rate 4/5 code. The decoded word contains information bits as well as check bits for error detection, ( see Fig. 2.1 ). If errors are detected at the receiver, one additional bit per puncturing period ( 4 intervals in this case ), i.e. $b_6$ in the above table, will be sent to form the rate 4/6 code at the receiver. If this code is still not powerful enough to correct the channel errors, one more redundant bit ( per puncturing period ), $b_4$, will be transmitted to reduce the effective code rate of the FEC to 4/7. If the receiver once again detects that an uncorrectable number of errors have occurred, then, another additional bit $b_8$ will be used. Now, the received bits accumulated at the receiver is a noise corrupted codeword of the rate 1/2 code, and the rate 1/2 code offers the highest error correction capability in this example. This transmitting and decoding process is conceptually shown in Fig. 2.4.

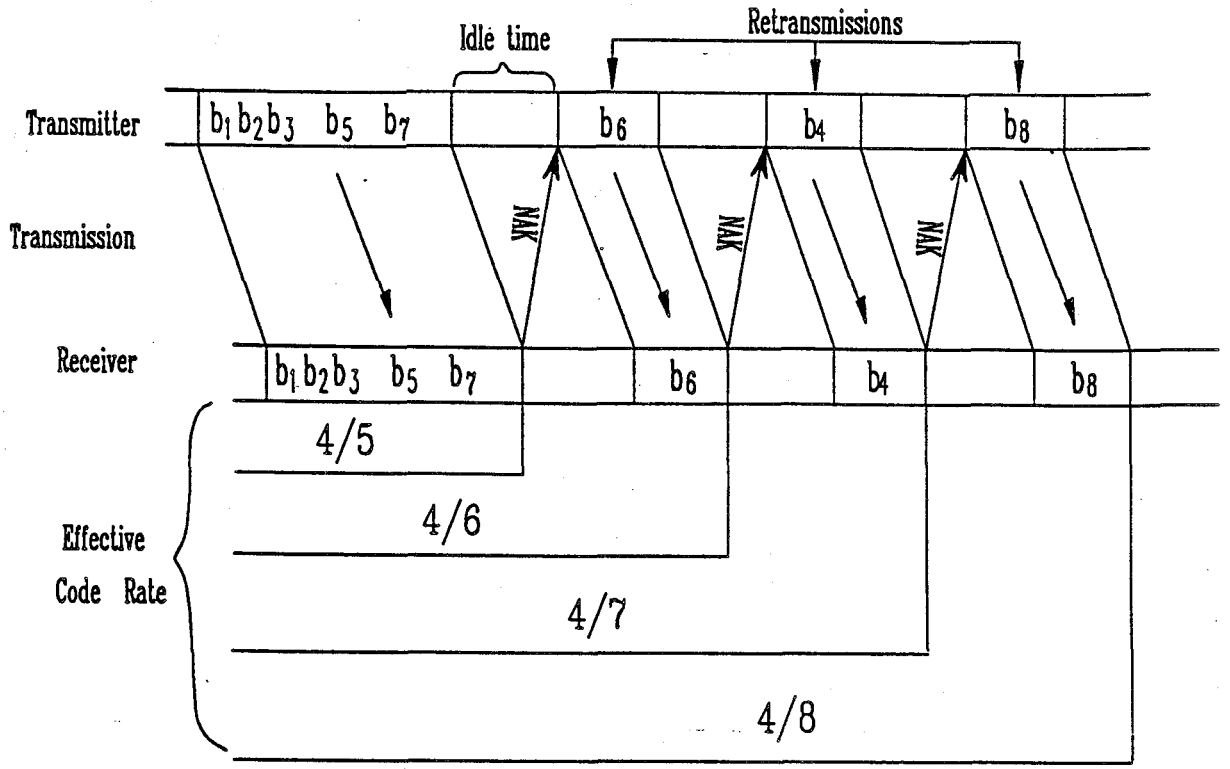



Fig. 2.4 Transmitting and decoding process of an RCPCC scheme.

It should be pointed out that in general, the number of additional bits ( per puncturing period ) transmitted in each decoding attempt is not necessarily 1 as in the above example. For instance, one can use an exponential increase in the number of additional bits sent in the successive retransmissions. In this way, the effective code rate of the FEC can be decreased rapidly so that the error protection power of the code increases substantially in consecutive decoding attempts. As mentioned earlier, only one Viterbi decoder is needed to decode all the codes obtained by puncturing the mother code. This is because all the coded bits used by a higher rate punctured code are contained in the mother code.

In order to have an efficient generalized type II hybrid ARQ/FEC scheme, we need to have a sequence of good codes. Initially one can assume a good mother code will generate good punctured codes. With this assumption, the problem of

finding a sequence of good codes is equivalent to finding a good mother code. It should be noted, though, even with a good mother code, good punctured codes can only be obtained if the puncturing rule is chosen appropriately. At present, the best puncturing rule can only be obtained through exhaustive computer search. As far as RCPC codes with Viterbi decoding is concerned, the usual optimal criterion is a large free Hamming distance $d_{fH}$. In other words, a convolutional code is good when its $d_{fH}$ is almost the largest possible for a given constraint length and a given code rate. Here the constraint length is defined as the minimum number of shift register stages required in the encoder. In [14], Hagenauer reported the results of a computer search for families of RCPC codes with rates between 1/4 and 8/9. His results show that some RCPC codes obtained are as good as the best known codes with the same rates and with the same constraint lengths.

## 2.2 SYSTEM PERFORMANCE

One of the most important performance measures for an error control system is its throughput efficiency, $R_{av}$. For the system using RCPCC, it is defined as the number of information bits per packet, $n_i$ divided by $n_{av}$, the average number of bits transmitted to deliver these $n_i$ bits to the end user:

$$R_{av} = n_i / n_{av} \tag{2.1}$$

In general, the value of $R_{av}$ depends on various factors such as the ARQ protocol used; the length of the header which contains the source and destination addresses, as well as other useful routing information; the rate of the error detecting code; the header failure probability; the round trip propagation delay; and the acknowledgement packet failure probability. For convenience, several assumptions are made in this

thesis and are applied to the analysis and comparisons in later chapters: (1) a stop - and - wait generalized hybrid ARQ/FEC with $n_i$ information bits per packet is used; (2) the undetected error probability is small and hence negligible; (3) the header is separately encoded from the packet; (4) the headers and the acknowledgements are always received correctly ( The information about the header's sensitivity, and the effect of noisy feedback channels can be found in [11] ); (5) the round trip propagation delay has no effect on the throughput performance. When the round trip delay is very small, like that in a mobile radio system, assumption (5) is definitely true for a stop - and - wait hybrid ARQ/FEC system. In addition, due to the configuration of typical mobile radio systems, the waiting time need not be wasted. It can be used to send packets to other stations.

Besides the throughput, the frame error rate (FER) is another important performance measure for the hybrid ARQ/FEC system. Let K be the maximum number of decoding attempts for each data frame. When there are still errors detected after the Kth decoding attempt, we assume this frame of data that contains errors will be delivered to the users. The frame error rate, FER, is defined as the probability that a packet which is ultimately not correctly decoded. If the effect of channel noise to each decoding attempt is statistically independent, FER is determined by the failure probability of each of the K decoding attempts.

Let us consider an example taken from [14]. It is a hybrid ARQ/FEC system that uses a sequence of RCPC codes proposed by Hagenauer. This system has a packet length of 422 bits, including $n_c = 32$ cyclic redundancy check (CRC) bits and $\nu = 4$ bits for terminating the encoder trellis. A 16 state, rate 1/3 convolutional code is chosen as the mother code. BPSK modulation is employed. The puncturing period is $P = 8$ intervals. We will use the number of information bits that can be delivered

per BPSK pulse ( bits/pulse ), or pulse efficiency, to describe the code rate of the RCPC codes. The reason is to facilitate the throughput comparison of the hybrid ARQ/FEC error control system that uses RCPC codes with our proposed hybrid ARQ/FEC error control systems that use TCM ( reported in later chapters ) based on the same bandwidth. Let $( c_1, c_2, \cdots, c_{24} )$ denotes the 24 BPSK pulses in each puncturing period of 8 encoding intervals of the mother code. The RCPC codes are given in Table 2.2.

| BPSK symbols transmitted per puncturing period in the *i*th transmission | Symbols used in the *i*th decoding attempt | The effective code rate at the *i*th decoding attempt (bit/pulse) |
|---|---|---|
| 1st $c_1 c_2 c_4 c_7 c_{10} c_{14} c_{16} c_{19} c_{22}$ | $c_1^* c_2^* c_4^* c_7^* c_{10}^* c_{14}^* c_{16}^* c_{19}^* c_{22}^*$ | 8 bits / 9 pulses $\Rightarrow$ 8/9 |
| 2nd $c_{13}$ | $c_1 c_2 c_4 c_7 c_{10}$ $c_{13}^* c_{14} c_{16} c_{19} c_{22}$ | 8 bits / 10 pulses $\Rightarrow$ 4/5 |
| 3rd $c_8 c_{20}$ | $c_1 c_2 c_4 c_7 c_8^* c_{10}$ $c_{13} c_{14} c_{16} c_{19} c_{20}^* c_{22}$ | 8 bits / 12 pulses $\Rightarrow$ 2/3 |
| 4th $c_5 c_{11} c_{17} c_{23}$ | $c_1 c_2 c_4 c_5^* c_7 c_8 c_{10} c_{11}^*$ $c_{13} c_{14} c_{16} c_{17}^* c_{19} c_{20} c_{22} c_{23}^*$ | 8 bits / 16 pulses $\Rightarrow$ 1/2 |
| 5th $c_3 c_6 c_9 c_{12} c_{15} c_{18} c_{21} c_{24}$ | $c_1 c_2 c_3^* c_4 c_5 c_6^* c_7 c_8$ $c_9^* c_{10} c_{11} c_{12}^* c_{13} c_{14} c_{15}^* c_{16}$ $c_{17} c_{18}^* c_{19} c_{20} c_{21}^* c_{22} c_{23} c_{24}^*$ | 8 bits / 24 pulses $\Rightarrow$ 1/3 |

Table 2.2 A sequence of RCPC codes with puncturing period of 8 intervals. * indicates the symbol sent at the *i*th transmission.

Fig. 2.5 shows the throughput curve (taken from [14]) of Hagenauer's system that uses the RCPC codes listed above, working on a fully interleaved Rayleigh fading channel with soft decision decoding. Fig. 2.6 shows the corresponding estimated FER. Here, perfect knowledge about the various fading amplitudes during the transmission of the packet is assumed available at the decoder.

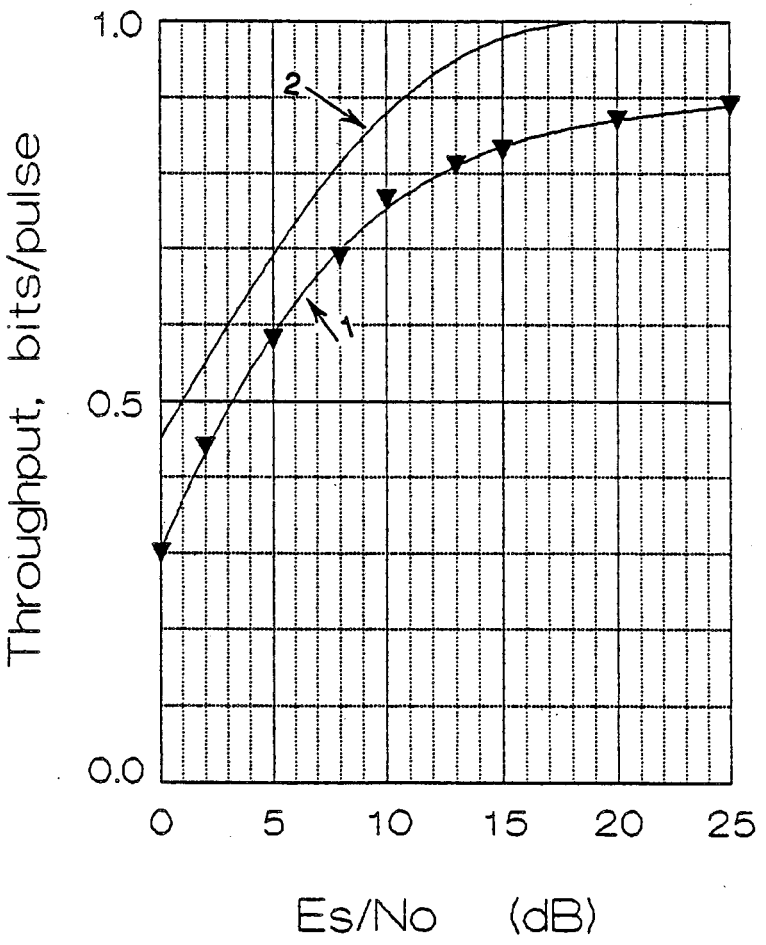



Fig. 2.5 The throughput of a hybrid ARQ/FEC system with RCPC codes (Curve 2) and the cut off rate for BPSK modulation (Curve 1). Both are for a fully interleaved Rayleigh channel with perfect channel state information.

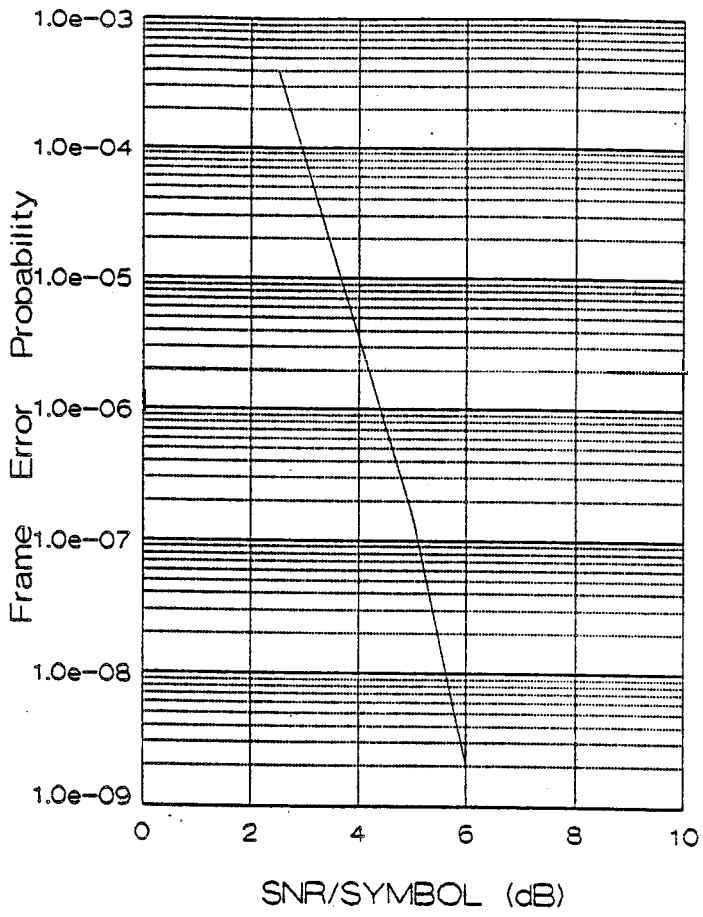



Fig. 2.6 The frame error rate of a hybrid ARQ/FEC system with RCPC codes on an interleaved Rayleigh channel.

Recall in section 1.1, we described the characteristics of the throughput performance of a pure ARQ system, a type I hybrid ARQ/FEC system and a type II hybrid ARQ/FEC system (with the same rate of FEC). Compared with these systems, Hagenauer's system gives a more efficient and smoother throughput performance, see Fig. 2.7.

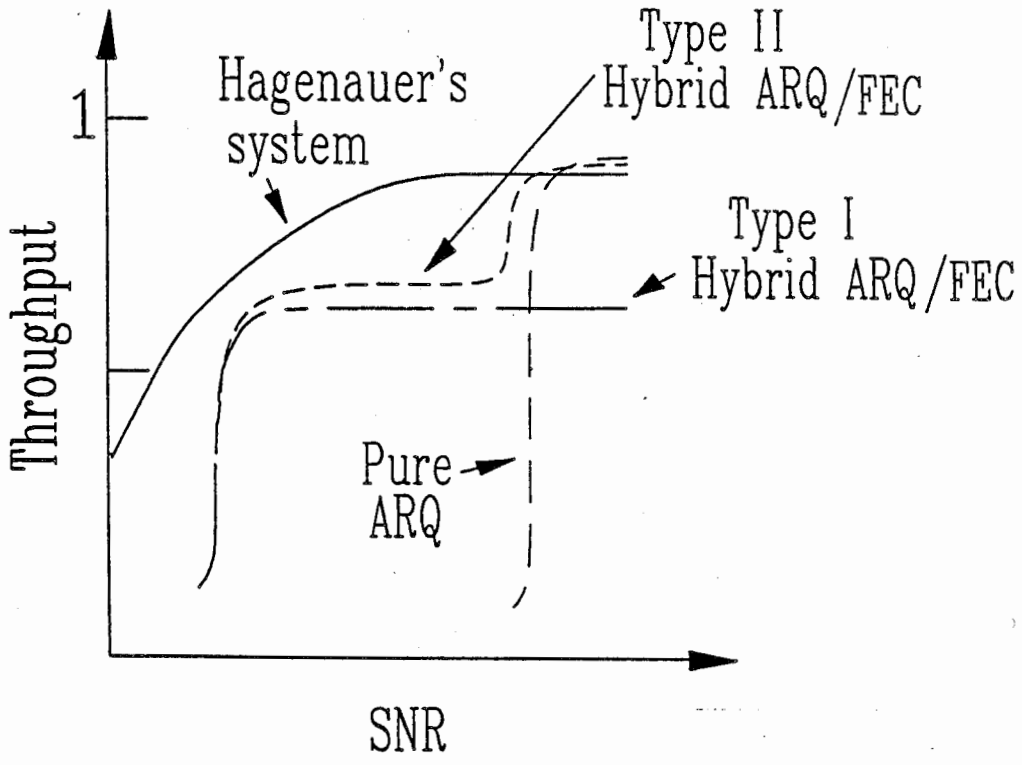



Fig. 2.7 Throughput comparisons of several error control systems.

To further point out the efficiency of Hagenauer's system, we plot in Fig. 2.5 the cut off rate curve for BPSK modulation in an interleaved Rayleigh channel.

The cut off rate, $R_0$, is a parameter similar to the channel capacity. It indicates a limit in the data rate of a channel coding system that uses a particular modulation format. In particular, $R_0$ is defined as

$$\overline{P}_e \leq 2^{-\kappa(R_0 - R)} \tag{2.12}$$

where $\overline{P}_e$ is the average error probability over all possible coding systems, and $\kappa$ is the number of modulation symbols in each codeword. The above equation says

that as long as the data rate R is less than $R_0$, $\overline{P}_e$ can be made arbitrarily small by increasing κ. Mathematically, for a fully interleaved Rayleigh fading channel with M - ary phase - shift - keying (MPSK) modulation[1], $R_0$ is given by :

$$R_0 = -\log_2 \frac{1}{M} \sum_{m=0}^{M-1} \frac{1}{1 + \left(\frac{\overline{E}_s}{N_0}\right) \sin^2 \frac{\pi m}{M}} \tag{2.13}$$

where ideal interleaving and perfect channel state information is assumed available in (2.13). As mentioned before, by perfect channel state information we mean that at each time interval, the effect of channel fading process to the signal is known at the receiver. In (2.13), $\overline{E}_s$ is the average received energy per pulse, and $N_0/2$ is the two - sided power spectral density (PSD) of the channel additive white Gaussian noise (AWGN).

Fig. 2.5 demonstrates that the performance of Hagenauer's error control system is close to the cut off rate curve for almost any channel condition and hence his RCPC codes are very efficient coding schemes. However, as observed from Fig. 2.5, the throughput of Hagenauer's system is limited to 1 bit/pulse due to the fact that binary modulation and binary coding are used. We plot in Fig. 2.8 the cut off rate curves for M = 4, and M = 8 PSK signals. They indicate that to achieve higher throughput, one has to use multi - level coded modulation schemes like those proposed by Ungerboeck [23]. In the next chapter, we show how to integrate TCM technique into hybrid error control systems.

---

1 The general representation for a set of MPSK signaling waveform is

$$S(t) = \sqrt{\frac{2}{T}} \cos\left(2\pi f_c t + \frac{2\pi}{2}(m-1)\right) \qquad m = 1, 2, \ldots, M \qquad 0 \le t \le T$$

where T is the duration of the signaling interval, $f_c$ is the carrier frequency, the energy of each MPSK pulse is 1.

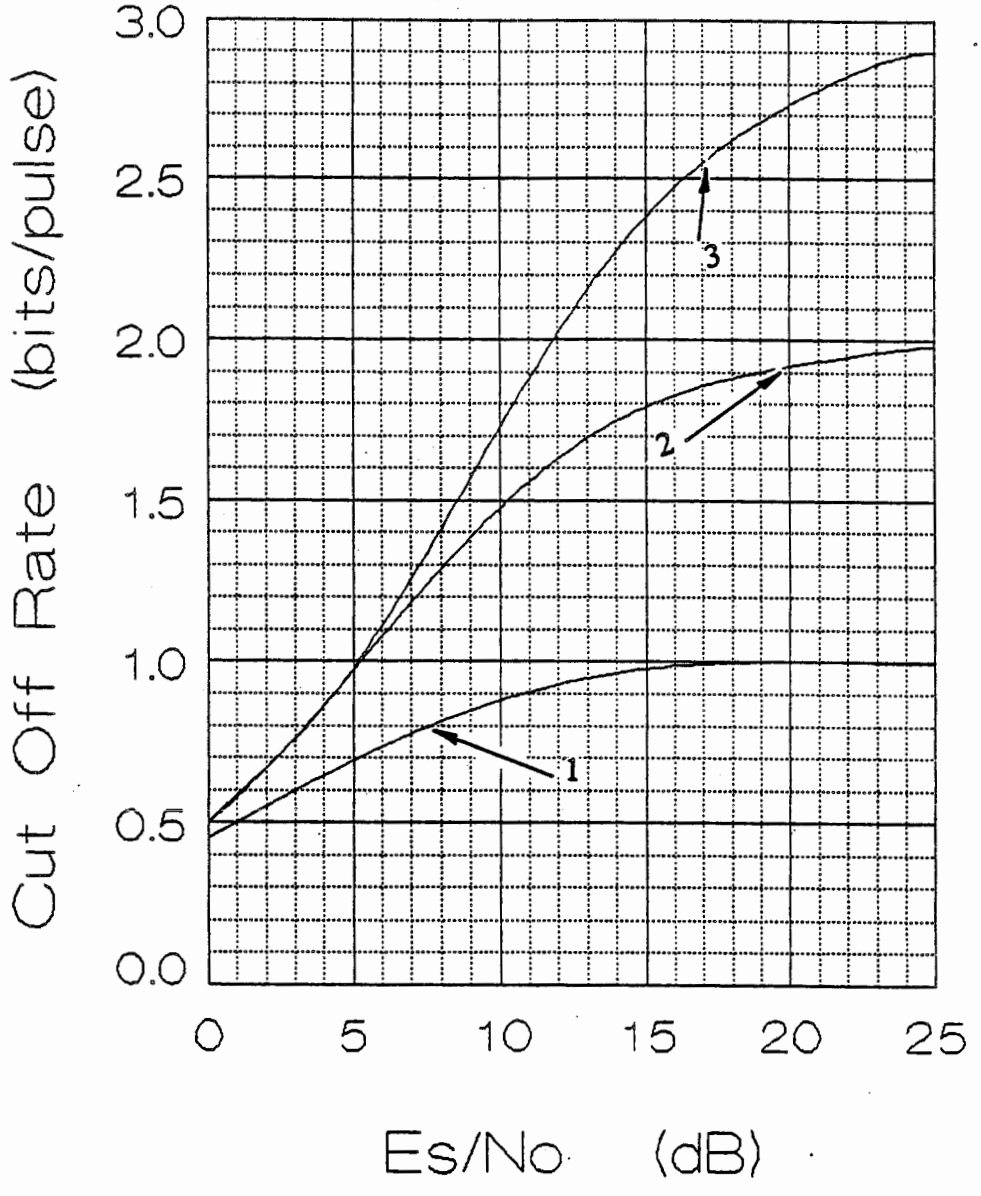



Fig. 2.8 Cut off rates for M = 2, 4, 8 PSK in an interleaved Rayleigh channel.

# CHAPTER THREE

# ADAPTIVE HYBRID ARQ/FEC PROTOCOLS USING TCM

In this chapter, a new adaptive hybrid ARQ/FEC scheme that uses Trellis Coded Modulation (TCM) is proposed and discussed. First, we will review the concept of TCM. This will be followed by a description of how to integrate TCM into a rate adaptive hybrid error control system. As mentioned earlier, the throughput of an error control system that uses traditional channel coding with binary modulation is limited to 1 bit per channel symbol. As will be shown later, the throughput efficiency can be greatly increased if we use TCM schemes in an adaptive ARQ/FEC system.

## 3.1 UNGERBOECK'S TCM CODES

### 3.1.1 TCM In An AWGN Channel

As mentioned before, in a digital communication system, immunity to noise can be provided by using forward - error - correction codes (FEC). The redundant symbols in a FEC increase the Hamming distances between information sequences and hence lower the probability of decoding error. Since redundant symbols are added, bandwidth expansion is required. In [23], Ungerboeck proposed a novel technique that can improve noise immunity without any bandwidth expansion. His idea is actually quite simple and is based on the fact that redundancy can be introduced by increasing the size of the signal set, instead of introducing redundant symbols as in the conventional approach. To see this, let us consider the following example. Suppose we want to

send two bits of information per modulation pulse. One of the modulation formats that we can use is QPSK ( or 4PSK ). The signal constellation of QPSK is shown in Fig. 3.1,

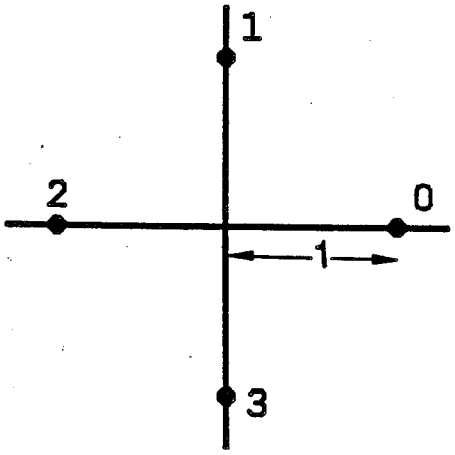



Fig. 3.1 Signal constellation of QPSK.

where signal point i, i = 0, 1, 2, 3, corresponds to the QPSK waveform of

$$S_i(t) = \sqrt{\frac{2}{T}}\cos\left(2\pi f_c t + i\frac{\pi}{2}\right) \qquad 0 \le t \le T \tag{3.1}$$

In the above equation, $f_c$ is the carrier frequency, T is the pulse duration and the energy per QPSK pulse is 1. Note that each point in the constellation can be represented by a complex number of the form

$$e^{j\frac{\pi}{2}i} \qquad i = 0, 1, 2, 3.$$

Assuming we have a white Gaussian noise (AWGN) channel with a power spectral density of $N_0/2$, then it is well known from communication theory that at large signal to noise ratio (SNR), the probability of demodulation error will be dominated by the minimum Euclidean distance error event, i.e. those events of confusing a signal with its nearest neighbors in the signal constellation. From the signal constellation of QPSK, we can show easily that the minimum squared Euclidean distance is

$$d_{\min}^2 = 2 \tag{3.2}$$

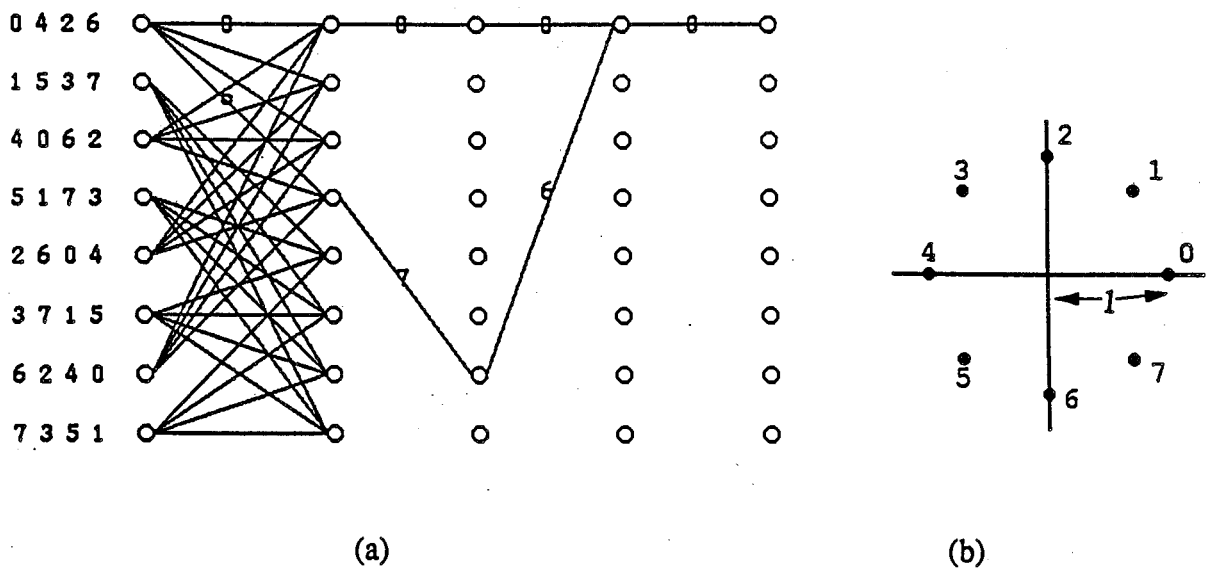
Since each QPSK symbol contains 2 bits of information, we have $E_b = 1/2$. $E_b$ is the bit energy. Hence, $d_{\min}^2 = 4E_b$. It follows that at larger SNR, the error probability is

$$Q\left(\sqrt{\frac{d_{\min}^2}{2N_0}}\right) = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = Q\left(\sqrt{\frac{1}{N_0}}\right) \tag{3.3}$$

where $Q(\alpha)$ is the Q - function defined as :

$$Q(\alpha) \equiv \frac{1}{\sqrt{2\pi}} \int_{\alpha}^{\infty} \left\{ e^{-\gamma^2/2} \right\} d\gamma \tag{3.4}$$

Now, we want to show how we can increase the minimum Euclidean distance by trellis coding with a redundant signal set. By a redundant signal set we mean that we use 8PSK ( or higher level modulation ) instead of using 4PSK to send 2 bits of information. As its name may suggest, a trellis encoder is a finite state sequential machine, like the one shown in Fig. 3.2(a).




(a) (b)

Fig. 3.2 (a) Trellis structure of an Ungerboeck's 8 state code.
(b) 8PSK signal constellation.

Here we use the same notation as in representing convolutional code, i.e. the circles represent the states, and the branches represents transitions between states. There are four branches originating from each of the 8 states, due to the fact that 2 bits are encoded per interval. The transition that the encoder will take depends on the present encoder state as well as the input. During a transition, the encoder will emit a modulation waveform chosen from the 8PSK signal set in Fig.3.2(b). Here signal point $i$, $i = 0, 1, \ldots, 7$, in Fig. 3.2 (b) corresponds to a modulation waveform of

$$S_i(t) = \sqrt{\frac{2}{T}} \cos\left(2\pi f_c t + i\frac{\pi}{4}\right) \qquad 0 \le t \le T \tag{3.5}$$

Since there are 8 modulation waveforms available while there are only 4 possible transitions from each state, coding here then refers to the proper assignment of waveforms to the encoder transitions so as to maximize the free squared Euclidean distance, which is defined as:

$$d_{fE}^2 = \min_{C_i \neq C_j} \sum_k d_E^2(c_{ik}, c_{jk}) \tag{3.6}$$

In the above equation, $C_i = (c_{i1}, c_{i2}, \cdots, c_{ik}, \cdots)$ and $C_j = (c_{j1}, c_{j2}, \cdots, c_{jk}, \cdots)$ are 2 sequences of 8PSK symbols associated with 2 different paths in the trellis, $d_E^2(c_{ik}, c_{jk})$ means the Euclidean distance between the complex 8PSK symbols $c_{ik}$ and $c_{jk}$, and the minimum is taken over the set of all possible $C_i$ and $C_j$ that originate from the same state and that terminate at the same state.

Ungerboeck proposed a heuristic rule of assigning 8PSK symbols to the various transitions to obtain a large $d_{fE}^2$. He called this method "mapping by set partitioning". It is based on the fact that although the minimum square distance between signal points in Fig. 3.2(b) is $\Delta_0^2 = 0.586$, the signal subsets A and B obtained by taking the odd signal points and the even signal points each have a larger minimum squared distance of $\Delta_1^2 = 2$; see Fig. 3.3.

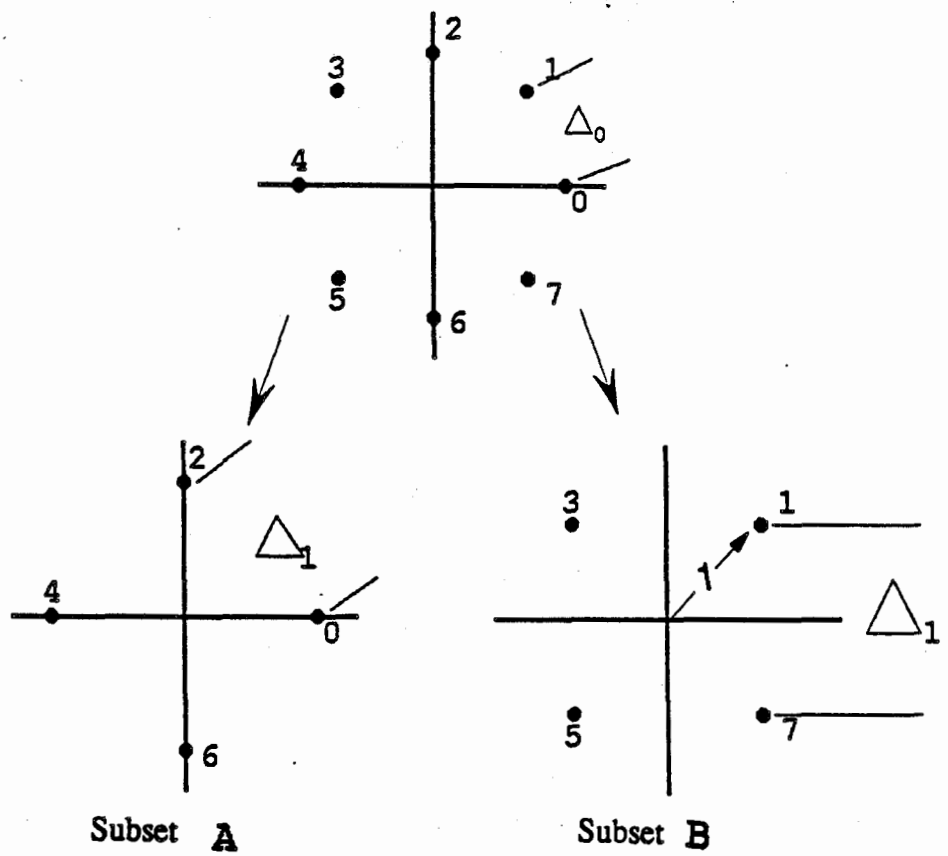



Fig. 3.3 Partitioning of 8PSK signals into subsets with increasing minimum Euclidean distance in subsets.

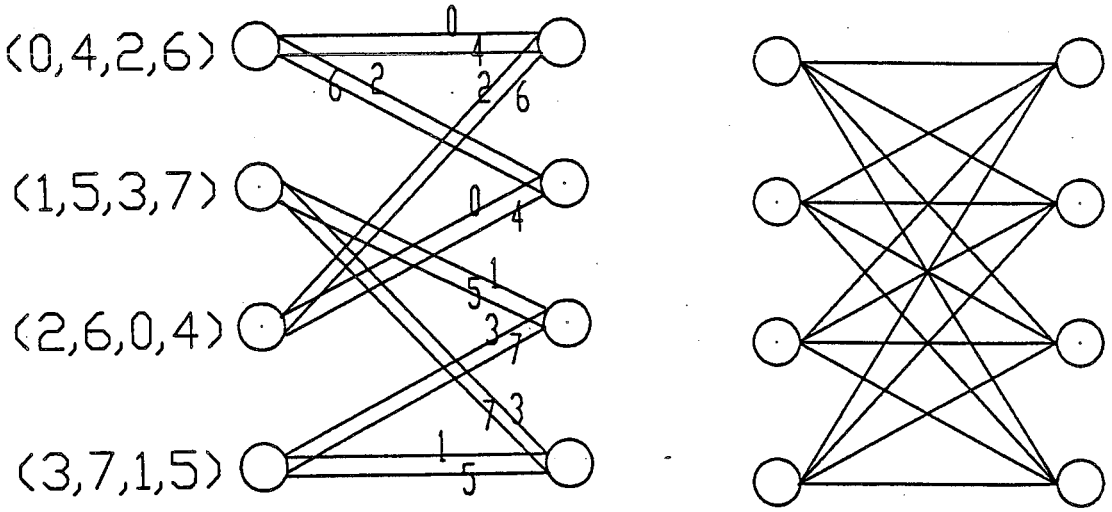
If we assign signals from the same subset to those transitions originating from the same state and those terminating at the same state, we will have at least a squared free distance of

$$d_{fE}^2 \geq \Delta_1^2 + \Delta_1^2 = 4 = 8E_b \tag{3.7}$$

where again $E_b = 1/2$, due to the fact that each pulse carries 2 bits of information. The reason why $8E_b$ is a lower bound on the squared free distance is that it only represents the squared distances accrued during the initial split and the final remerge of 2 paths in the trellis and there may be additional distances accrued during other

intervals. In any case, comparing the coded distance of $8E_b$ with the uncoded QPSK distance of $4E_b$, we will have a coding gain of 3 dB, i.e. trellis coding requires 3 dB less energy to achieve the same bit error probability. This coding gain comes without any increase in bandwidth. This is because we are sending one pulse for every 2 bits of information, just as in the case of QPSK. This example clearly indicates the attractiveness of trellis coded modulation. One can see from Fig. 3.2 that the transitions originating from and terminating at each state indeed receive signals from the same subsets. For example, the numbers 0, 4, 2, 6 associated with the first state indicates that the 8PSK symbols sent in the first, second, third, and fourth transitions are the signal points labelled 0, 4, 2, 6 respectively in Fig. 3.3.

In some cases, the subsets A and B in Fig. 3.3 are required to be further partitioned into subsets. Consider once again a TCM that sends 2 bits of information per 8PSK pulse. If the number of encoder states is 4, then we have two possible encoder trellis structures, see Fig. 3.4(a), (b).




(a) (b)

Fig. 3.4 Possible trellis structures for TCM with 4 states, 2 input bits per interval, and 8PSK modulation.

The trellis structure in Fig. 3.4(a) represents an encoder with parallel transitions, since there are always 2 possible transitions between pairs of states. Since parallel transitions represents error events of length 1 interval, they must be assigned signal points that have the largest distance separated in the signal constellation. Fig. 3.5 shows how the subsets A and B in Fig. 3.3 are further partitioned into subsets, A0, A1, B0, and B1.

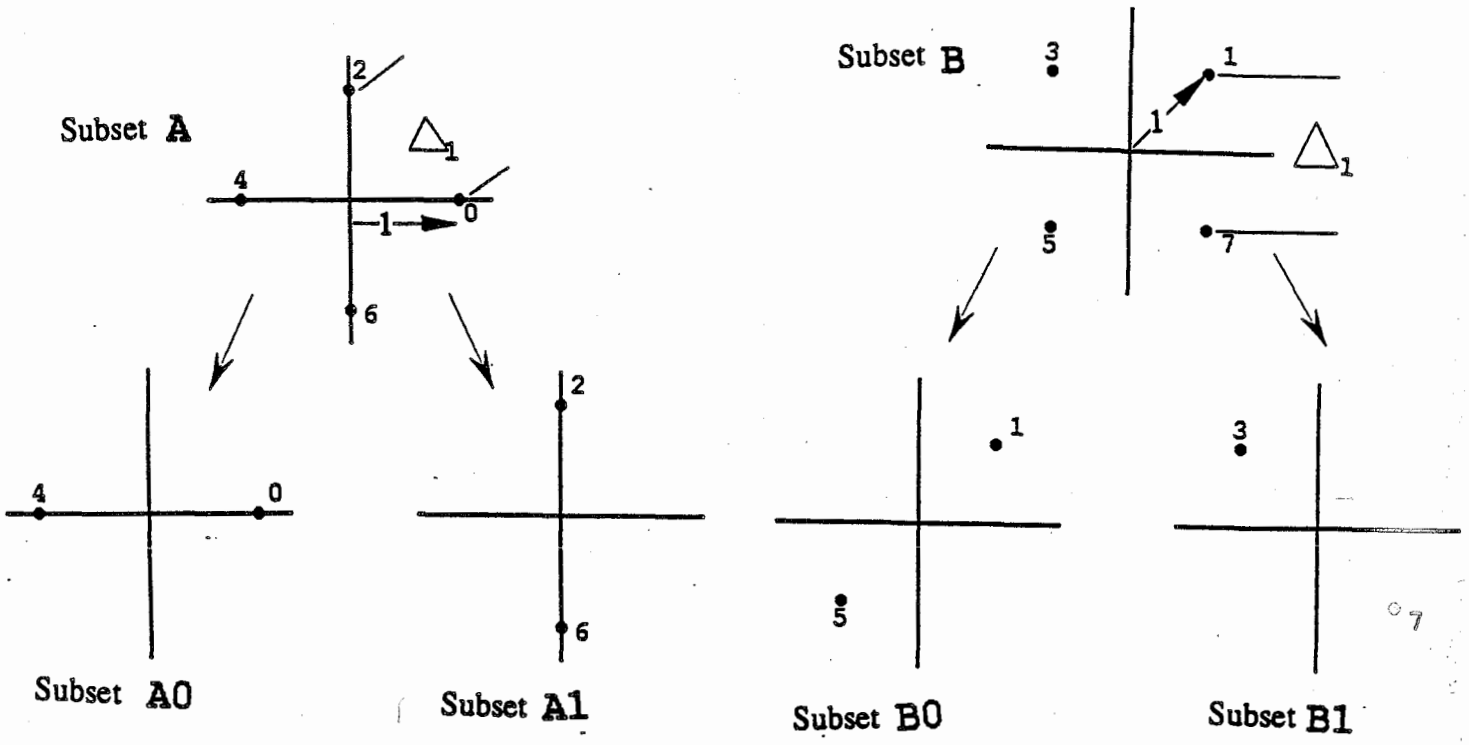



Fig. 3.5 Further partitioning of the subsets in Fig. 3.3

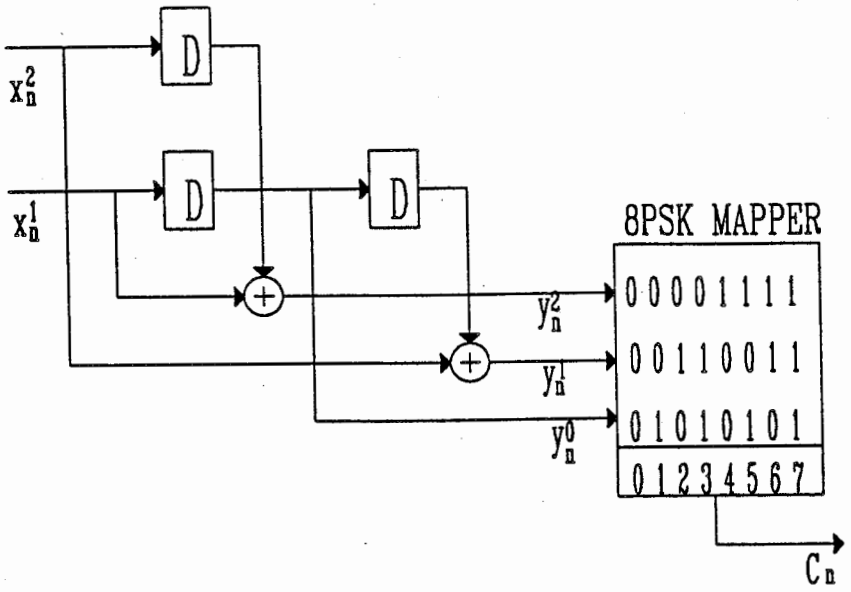
From Fig. 3.5 we see that each of the two signal points in A0 or A1 or B0 or B1 give the largest distance separation in the 8PSK signal constellation. We can observe from Fig. 3.4(a) that parallel transitions indeed always receive signals from subset A0 or A1 or B0 or B1.

Below is a summary of Ungerboeck's heuristic code design rules for AWGN channels [23]:

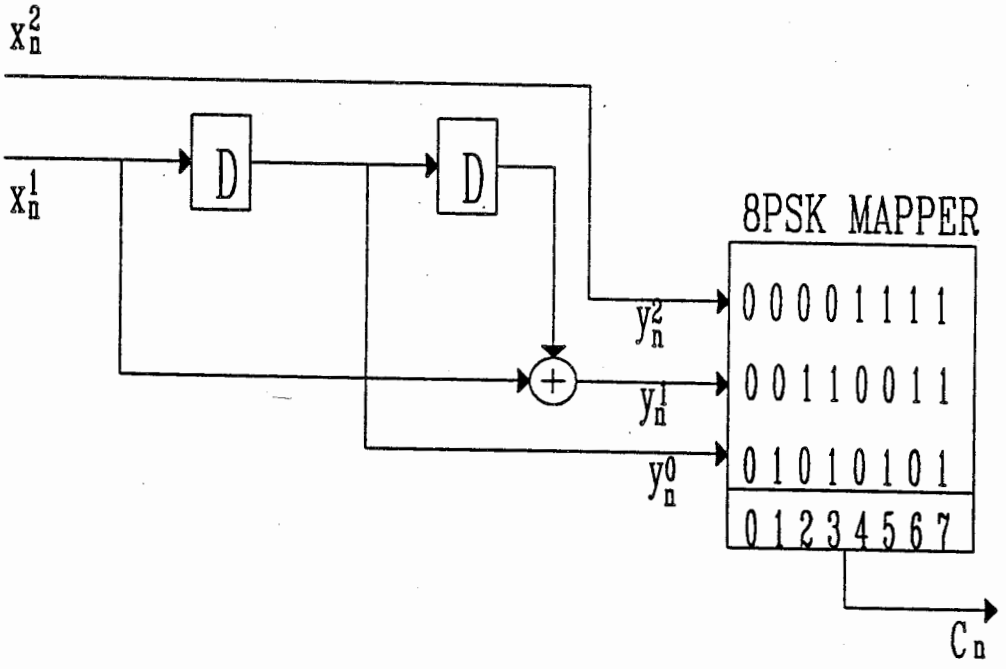
- All channel (8PSK) symbols should occur with equal frequency and with a fair amount of regularity and symmetry.
- Transitions originating from or merging in the same state receive signals either from subset A or B.
- Parallel transitions receive signals either from subset A0 or A1 or B0 or B1.

These rules guarantee reasonably good codes for the white Gaussian noise channel. However, to find out the best code for a given encoder structure, one has to rely on computer search.

The trellis encoders in Fig. 3.2(a) and Fig. 3.4(a) can be realized in terms of rate 2/3 convolutional encoders followed by an 8 level mapper; see Fig. 3.6.




(a)




(b)

Fig. 3.6 (a) Convolutional encoder realization of the TCM scheme in Fig. 3.2 (a).

(b) Convolutional encoder realization of the TCM scheme in Fig. 3.4 (a).

Here in Fig. 3.6, $x_n^i, i = 1, 2$, are the binary information bits encoded during the $n$th encoding interval, and $y_n^j, j = 0, 1, 2$, are the corresponding encoded binary bits which will be mapped into a 8PSK symbol. The mapper is a natural mapper with $y_n^2$ being the most significant bit and $y_n^0$ the least significant bit. The result of the mapping is the number $c_n$ which is taken from the set { 0, 1, 2, . . ., 7 }. If $c_n = j$, it means the $j$th waveform in the signal constellation in Fig. 3.2 (b) will be sent during the $n$th interval.

### 3.1.2 TCM In A Rayleigh Fading Channel

In the last section, we mentioned that the error performance of a trellis coded modulation system operating in an AWGN channel is dominated by $d_{fE}^2$, the free squared Euclidean distance of the code. We will consider next the transmission of TCM over a Rayleigh fading channel. The channel model is depicted in Fig. 3.7.

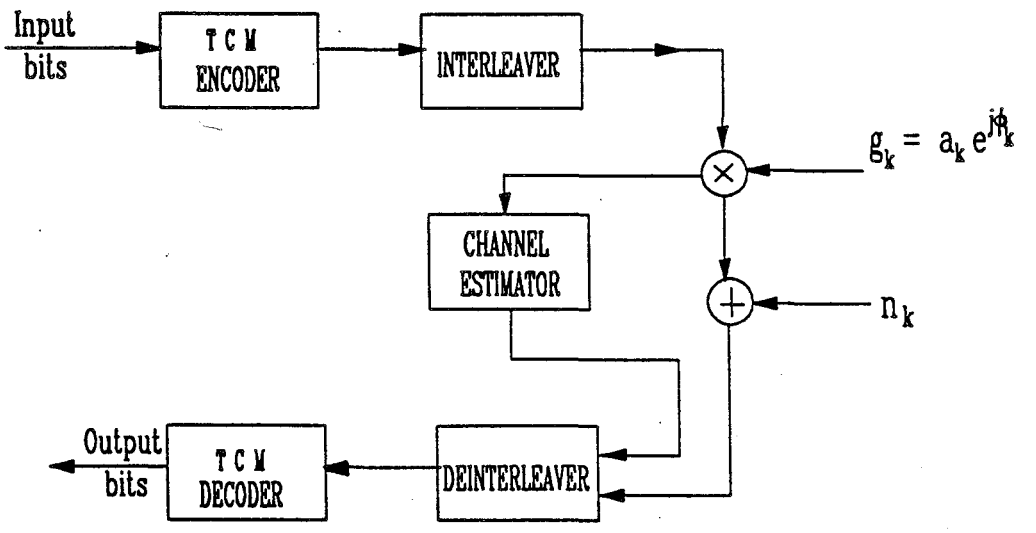



Fig. 3.7 Channel model of TCM transmitted over a Rayleigh fading channel.

In this figure, $g_k = a_k e^{-j\phi_k}$ is a distortion experienced by the $k$th modulation symbol sent. $g_k$ is a complex Gaussian random variable. Its amplitude, $a_k$, is Rayleigh distributed with the probability density function :

$$P(a) = 2ae^{-a^2} \qquad a \geq 0 \tag{3.8}$$

Its phase, $\phi_k$, is uniform distributed in the interval $[0, 2\pi]$, and $E[|g_k|^2] = 1$, where $E[\cdot]$ means the statistical average. Note that the $g_k$'s are correlated and the correlation depends on the fading spectrum. The purpose of the interleaver and de-interleaver in the system is to de-correlate the distortion experienced by the modulation symbols sent in different intervals. The following assumptions are used in this study: (1) perfect channel state information is available, i.e. the $g_k$'s are known at the receiver; (2) fading is slow relative to the symbol rate so that $g_k$ accurately represents the fading in the $k$th symbol interval; (3) the interleaving depth is so large that each deinterleaved symbol is affected independently by the channel fading process. With assumption (1), we are implicitly saying that we are able to achieve perfect coherent reception at the receiver. This is not an unrealistic assumption, since techniques like pilot tone [43] and pilot symbols aided detection provide reasonably good results in phase recovery. Assumption (2) is easily justified for mobile radio applications, because the fading rate is usually less than 1% of the symbol rate. Assumption (3) is also quite reasonable since we have found [31] that as far as error probability is concerned, a finite, large interleaving depth is almost as good as an infinite interleaving depth.

It has been found [25] that when trellis codes are used in a Rayleigh fading channel, the design of the codes for optimum error performance is only weakly dependent on the minimum squared Euclidean distance, $d_{fE}^2$, at large SNR, as opposed to the case of an AWGN channel. Specifically, the error performance at large SNR is dominated by $d_{fH}$, the length of the shortest error event path, as well as $\Pi d^2$, the product of squared branch Euclidean distances along the shortest error event path. What we mean by the "length" of an error event path is actually the Hamming distance between two coded sequences, counted by the channel symbols. $d_{fH}$ is therefore the free Hamming distance of the code. For example, Fig. 3.8 illustrates the shortest error event path of the 8 state Ungerboeck's code shown in Fig. 3.2(a). In this case, $d_{fH} = 2$. Obviously, it is different from the Hamming distance of 3 associated with the minimum Euclidean distance path.

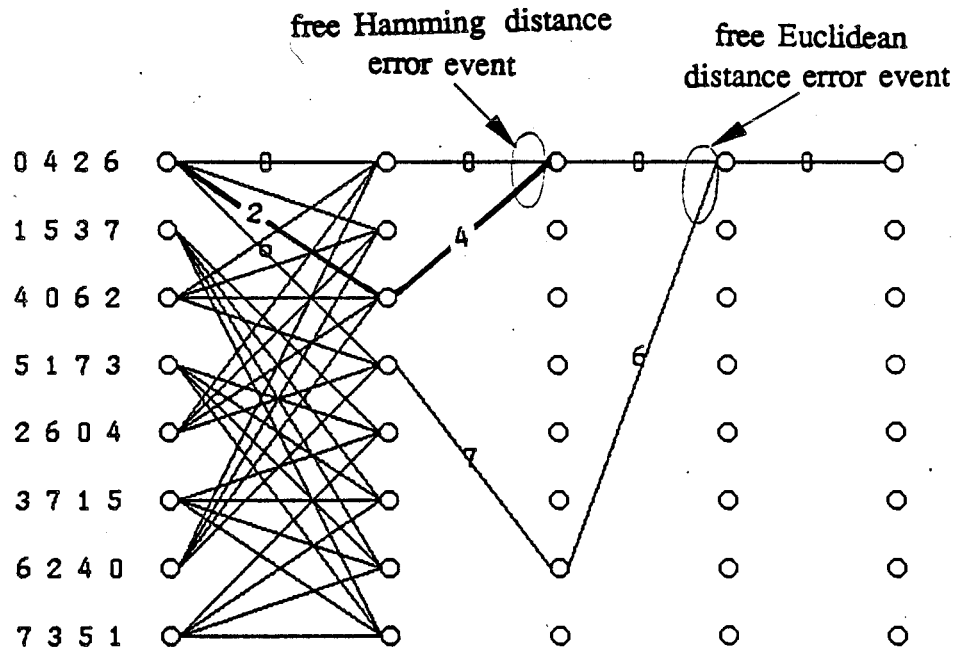



Fig. 3.8 The shortest Hamming distance error event and the free Euclidean diatance error event of the 8 state Ungerboeck's code.

As mentioned above, The other factor that influence the error probability in the fading channel is $\Pi d^2$, the product of squared branch Euclidean distances along the short error event paths. This can be explained with the expression for the pairwise error event probability $P(i \rightarrow j)$ for trellis coded modulation schemes operating in Rayleigh fading channels. A way to derive the exact expression of $P(i \rightarrow j)$ has been found by Cavers and Ho [22] and it will be used in the analysis of the throughput performance in the next chapter. Here, for simplicity in the explanation, let's look at the Divsalar and Simon's upperbound of the pairwise error event probability obtained by using the Chernoff bound [25]:

$$P(i \rightarrow j) \leq \left\{ \prod_{k \in \eta} \left( \frac{E_s}{4N_0} \right) | c_{ik} - c_{jk} |^2 \right\}^{-1} \tag{3.9}$$

where $C_i = (c_{i1}, c_{i2}, \cdots, c_{ik}, \cdots)$ is the transmitted codeword, $C_j = (c_{j1}, c_{j2}, \cdots, c_{jk}, \cdots)$ is another codeword, $\eta$ means all those values of $k$ such that $c_{ik} \neq c_{jk}$, the $c_{ik}$'s and the $c_{jk}$'s are complex coded MPSK symbols with a normalized magnitude equal to 1, $E_s$ is the average energy of each received symbol, and $N_0/2$ is the power spectral density (PSD) of the channel white Gaussian noise. At high SNR, an upper bound on the bit error probability can be obtained by simplifying (3.9) to :

$$P_b \leq C \frac{4^{d_{fH}}}{(E_s/N_0)^{d_{fH}}} \tag{3.10}$$

where C is a constant related to the distance structure and the bit assignments of the codewords. This upper bound says that the error performance of the code decays at a rate of $1/(E_s/N_0)^{d_{fH}}$ at high SNR. We call $d_{fH}$ the code diversity order. From

(3.9) and (3.10), it should be clear that for a given complexity, a code (at a given rate and a given number of encoder states) which is optimized for the fading channels satisfies the following two criteria.

- The code has the largest diversity order $d_{fH}$.
- The code also has the largest product of branch Euclidean distances along the shortest error event path.

In [25], [26], Divsalar and Simon suggest a class of TCM, called multiple trellis coded modulations (MTCM), which for any given number of states, have larger $d_{fH}$ than that of Ungerboeck's TCM. This class of TCM is described in the next section.

## 3.2 MULTIPLE TRELLIS CODED MODULATION - MTCM

The essential difference between a multiple trellis code and a conventional trellis code is that a MTCM has more than one modulation symbol assigned to each trellis branch. In fact, if the multiplicity of a MTCM code, $\mu$, is defined as the number of symbols per trellis branch, a conventional trellis code can be viewed as a special case of MTCM with $\mu = 1$. According to (3.10), the higher the code diversity order, the better the code performance. Consider the trellis structure of a TCM code: if $2^b > N_s$, where b is the number of input bits to the encoder at each encoding interval and $N_s$ is the number of trellis states, there must be parallel paths between pairs of states. Whenever parallel paths exist, the free Hamming distance $d_{fH}$ or the code diversity order is restricted to 1 for conventional trellis code where only one channel symbol is assigned to each trellis branch. This implies that the error event probability of the code will asymptotically vary inversely with $E_s/N_0$ (for sufficiently large SNR).

In the case of multiple trellis code, since there are multiple symbols per transition, then even with the presence of parallel transitions, the diversity order $d_{fH}$ can be larger than 1, so that the error performance of the system can decrease at a faster rate. This is the main reason that Divsalar and Simon recommended the use of MTCM schemes.

As an example, let us consider the MTCM in Fig. 3.9. This code has 2 states, 16 transitions per state, a multiplicity of 2, and it uses 8PSK modulation. In each encoding interval, there are 4 input bits and 2 output 8PSK symbols. Since 2 pulses are sent every 4 input bits, the pulse efficiency is 2 bits/pulse. Given this encoder structure, the next step is to map 8PSK symbol pairs of the form (i, j), where i, j $\in$ { 0, 1, 2, . . ., 7 } are signal points from the signal constellation in Fig. 3.2(b), to the various transitions. Since the code has parallel transitions, the signal pairs assigned to parallel transitions should have a Hamming distance of 2. Also, the inter-set Hamming distance must at least equal 1, i.e. if $a_{ij}$ is the $j$th signal pair in set $i$, and $a_{kl}$ is the $l$th signal pair in set $k$, then

$$d_H(a_{ij}, a_{kl}) \geq 1 \tag{3.11}$$

for any $i \neq k$, and any $j$ and $l$. This criterion guarantees that error events which span more than one interval have at least a Hamming distance of 2. Finally, we would like to maximize the minimum product of squared branch distances within each set of signal pairs, as indicated by equation (3.9). With the help of heuristic mapping rules and computer search, Divsalar and Simon came up with the assignment in Fig. 3.9.

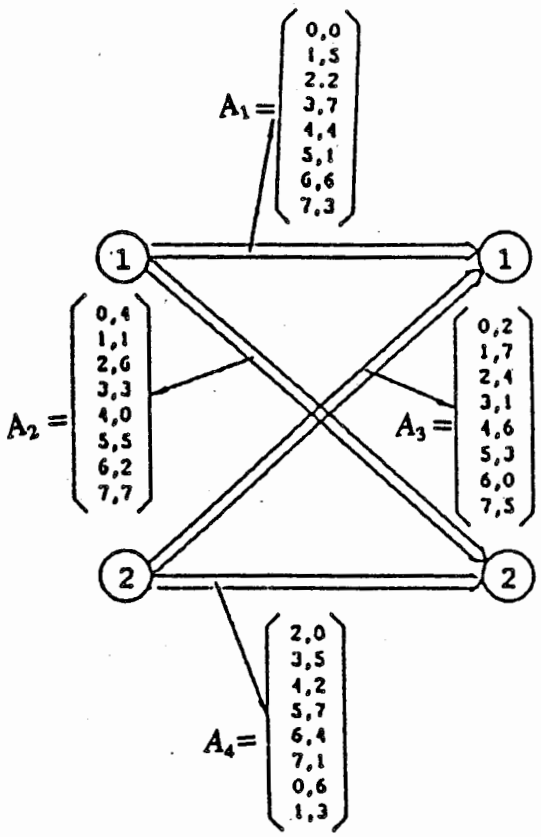



Fig. 3.9 A 2 state, rate 2 (bits/pulse) MTCM.

This signal assignment indeed provides a free Hamming distance of 2. However, the product of squared distance of each set $A_1$, $A_2$, $A_3$ and $A_4$ ( for example, between the signal pairs $(0, 0)$ and $(2, 2)$ of the set $A_1$ ) is not guaranteed to be the largest possible value for this trellis structure. A more detailed description about signal mapping of MTCM can be found in [25], [26]. In this study, we will only make use of existing good MTCM.

## 3.3 A NEW CODING APPROACH FOR HYBRID ARQ/FEC SYSTEMS

### 3.3.1 The Motivation

As described in Chapter Two, a sequence of good rate compatible punctured convolutional (RCPC) codes can provide unequal error protection over a wide range of SNR. However, these codes only allow us to achieve a throughput of at most 1 bit per channel symbol, no matter how good the channel condition is. Now, we are aiming at a more desirable error control system which is as adaptive as systems using RCPC codes, yet allowing us to achieve a throughput of more than 1 bit per channel symbol without bandwidth expansion. We saw in Chapter 2 that to achieve this, multilevel PSK modulation should be employed. Can multilevel PSK modulation like TCM be used in such a system ? An easy way to answer this question is to look at some examples. First, consider one of Ungerboeck's codes. It is the rate 2/3, 16 state trellis coded 8PSK modulation scheme with the trellis diagram shown in Fig. 3.10.

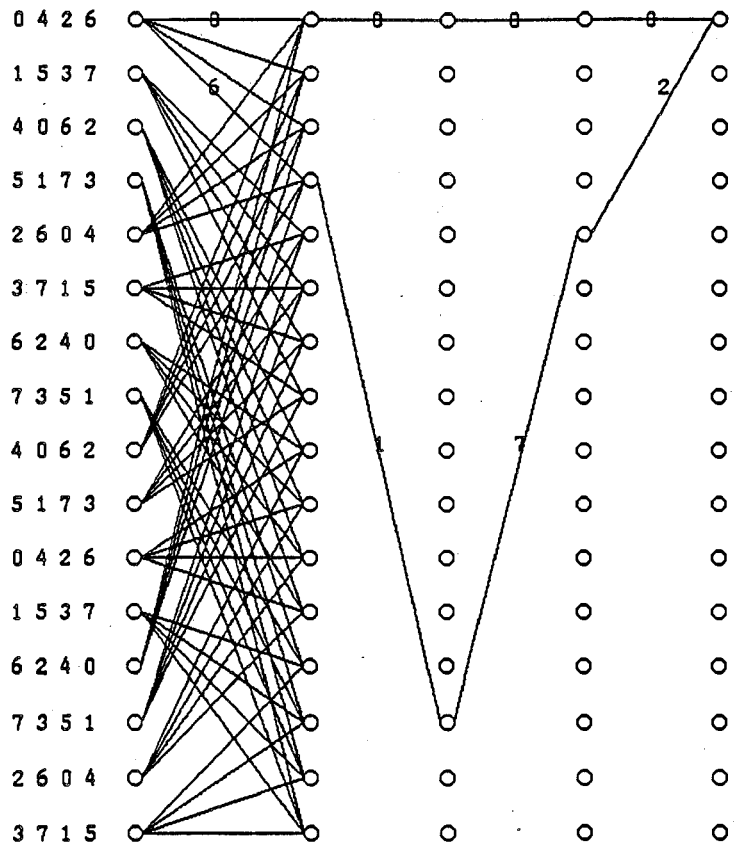



Fig. 3.10 Trellis structure of a 16 state Ungerboeck's code.

Although this code is optimally designed for an AWGN channel with $d_{fE}^2 = 5.172$, it is also desirable for fading channels since it has a code diversity order of 3, and the minimum product of the branch distances along the shortest error event path is 4.686. So this is a good coded modulation scheme for pure FEC. However, we found that this code, as well as other of Ungerboeck's codes, (or conventional trellis codes) can not be used as mother codes to obtain a sequence of rate compatible codes with higher rates. The reason is that Ungerboeck's codes, such as the one we see in Fig. 3.10, have only a single modulation symbol per trellis branch. Recall that in an adaptive hybrid ARQ/FEC system, like the one proposed by Hagenauer, in order to vary the FEC code rate adaptively according to the change in channel conditions, a mother code which can be punctured to obtain a sequence of rate

compatible codes is used. Convolutional codes can be used in such systems as mother codes because they all have multiple encoded bits ( or symbols ) per transition. When one bit is punctured, the remaining bit(s) can still be used to recover the information. In the case of Ungerboeck's codes, there is only one symbol per transition. If the symbol is punctured, the information in that interval is lost and there is no way that we can recover that piece of information. So the conclusion is that Ungerboeck's TCM can not be used as a mother code to obtain higher rate codes. But it can be used as a mother code to obtain a sequence of rate compatible codes with lower rates by way of periodically selective repeating its coded symbols. This will be considered later in Chapter 4.

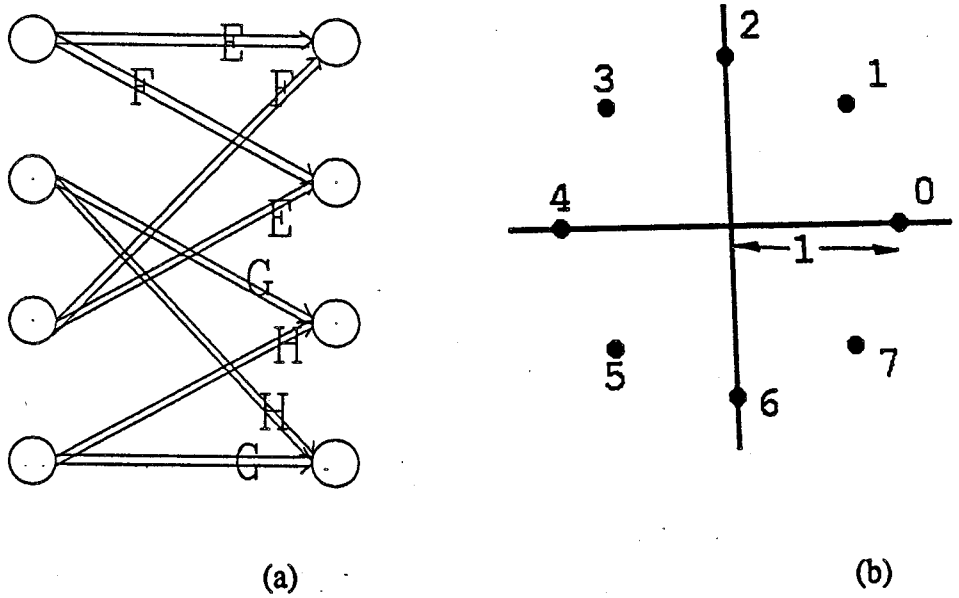
Now let us look carefully into the structure of MTCM, like that shown in Fig. 3.9. First we see a MTCM has a similar property as that of a mother convolutional code used in a Hagenaur's system, i.e. it has multiple symbols per trellis branch. This makes the puncturing procedure possible. In fact, as long as a MTCM with multiplicity $\mu$, modulation level M, and b information bits encoded per interval, satisfies the condition

$$\frac{b}{(\mu-1)\log_2 M} \leq 1 \quad , \tag{3.12}$$

it can be punctured at least one M - ary channel symbol from each interval without losing any information. Secondly, as a special class of trellis coded modulation, MTCM has the advantage of bandwidth efficiency. We will now examine the concept of using MTCM in an adaptive hybrid ARQ/FEC system in detail. For convenience, we call our new technique rate compatible trellis coded modulations ( RC - TCM ).

### 3.3.2 An Example

To illustrate the new technique, let's consider the following example. A 4 state, rate 1 bit/pulse trellis coded 8PSK scheme is chosen as the mother code of a RC - TCM system. The state diagram of this code is given in Fig. 3.11.




(a) (b)

Fig. 3.11 (a) Trellis diagram of a 4 state MTCM.

(b) 8PSK signal constellation.

There are 4 bits of information entering the encoder at each time interval, so there are 16 branches diverging from each of the four states. There are 8 parallel paths existing between the present states and the successor states in the trellis. Similar to the expression in the example of Section 3.2, E, F, G, and H in Fig. 3.11 represent the sets of channel symbols assigned to each of the 8 parallel transitions, and they are shown in equation (3.13), where once again the integers $0, 1, \ldots, 7$ are 8PSK symbols in Fig.3.2 (b). Note that each row in E, F, G, H has 4 8PSK symbols and hence the throughput of this code is 1 bit/pulse. As before, the E, F, G, H are chosen

so as to give the largest possible free Hamming distance and a large product of squared Euclidean distances for the encoder structure in Fig. 3.11. The intraset Hamming distance for each of the 4 sets is 4, the minimum interset Hamming distance is 2, and the product of squared Euclidean distances along the minimum intraset Hamming distance path in each set is 4.0.

$$E=\begin{bmatrix}0&0&0&0\\1&5&1&5\\2&2&2&2\\3&7&3&7\\4&4&4&4\\5&1&5&1\\6&6&6&6\\7&3&7&3\end{bmatrix} \quad G=\begin{bmatrix}0&2&0&2\\1&7&1&7\\2&4&2&4\\3&1&3&1\\4&6&4&6\\5&3&5&3\\6&0&6&0\\7&5&7&5\end{bmatrix} \tag{3.13}$$

$$F=\begin{bmatrix}0&4&0&4\\1&1&1&1\\2&6&2&6\\3&3&3&3\\4&0&4&0\\5&5&5&5\\6&2&6&2\\7&7&7&7\end{bmatrix} \quad H=\begin{bmatrix}0&6&0&6\\1&3&1&3\\2&0&2&0\\3&5&3&5\\4&2&4&2\\5&7&5&7\\6&4&6&4\\7&1&7&1\end{bmatrix}$$

For convenience, we denote the four symbols in each interval by $c_1, c_2, c_3, c_4$. A sequence of trellis codes with different rates can be obtained by periodically puncturing and repeating of the mother code. Assuming with a maximum of 6 attempts, the sequence of codes we used are obtained as follows. First, in obtaining higher rate codes, we puncture the mother code in Fig. 3.11 with a puncturing period of 1 interval. If we delete $c_4$ from each puncturing period ( of 1 interval ), we have a rate 4/3 bits/pulse code. If we delete one more, namely $c_3$, we have a rate 4/2

bits/pulse code. We can not puncture any further, due to the fact we have 16 transitions per state and only 8 different signal points available. In other words, by puncturing, we can vary the code rate from 1 to 2 bits/pulse, in two steps. To obtain lower rate codes from the mother code in Fig. 3.11, we have to repeat symbols of the mother code. If we repeat $c_1, c_2$ once in each interval, we can form a rate 4/6 = 2/3 bit/pulse code. If we repeat all 4 symbols once, we have a rate 4/8 = 1/2 bit/pulse code. If we repeat all 4 symbols twice, we can obtain a rate 4/12 = 1/3 bit/pulse code. Therefore, with both puncturing and repeating, we can vary the code rate from 2 to 1/3 bit/pulse in 6 steps. The following table summarizes our coding strategy.

| Symbols transmitted per puncturing period in the $i$th transmission | Symbols used in the $i$th decoding attempt | The effective FEC code rate in the $i$th decoding attempt (bit/pulse) |
|---|---|---|
| 1st $c_1\, c_2$ | $c_1^*\ \ c_2^*$ | 4 bits / 2 pulses $\Rightarrow$ 2 |
| 2nd $c_3$ | $c_1\ \ c_2\ \ c_3^*$ | 4 bits / 3 pulses $\Rightarrow$ 4/3 |
| 3rd $c_4$ | $c_1\ \ c_2\ \ c_3\ \ c_4^*$ | 4 bits / 4 pulses $\Rightarrow$ 1 |
| 4th $c_1\, c_2$ | $c_1\, c_1^*\, c_2\, c_2^*\, c_3\, c_4$ | 4 bits / 6 pulses $\Rightarrow$ 2/3 |
| 5th $c_3\, c_4$ | $c_1\ \ c_1\ \ c_2\ \ c_2$<br>$c_3\ \ c_3^*\ \ c_4\ \ c_4^*$ | 4 bits / 8 pulses $\Rightarrow$ 1/2 |
| 6th $c_1\, c_2\, c_3\, c_4$ | $c_1\, c_1\, c_1^*\, c_2\, c_2\, c_2^*$<br>$c_3\, c_3\, c_3^*\, c_4\, c_4\, c_4^*$ | 4 bits /12 pulses $\Rightarrow$ 1/3 |

Table 3.1 A sequence of rate compatible trellis codes. * indicates the symbol sent at the $i$th transmission.

It should be pointed out that an exhaustive search is not used to arrive at the above sequence of codes. We rely on the assumption that good codes generate good codes. The parameters that dominate the performances of these codes, such as free Hamming distance and product of Euclidean distances will be given in the next chapter. As one will see there, the above code sequence indeed yields rather good performance.

The use of the code sequence in Table 3.1 in a rate adaptive hybrid ARQ/FEC error control system is illustrated with the diagram below.

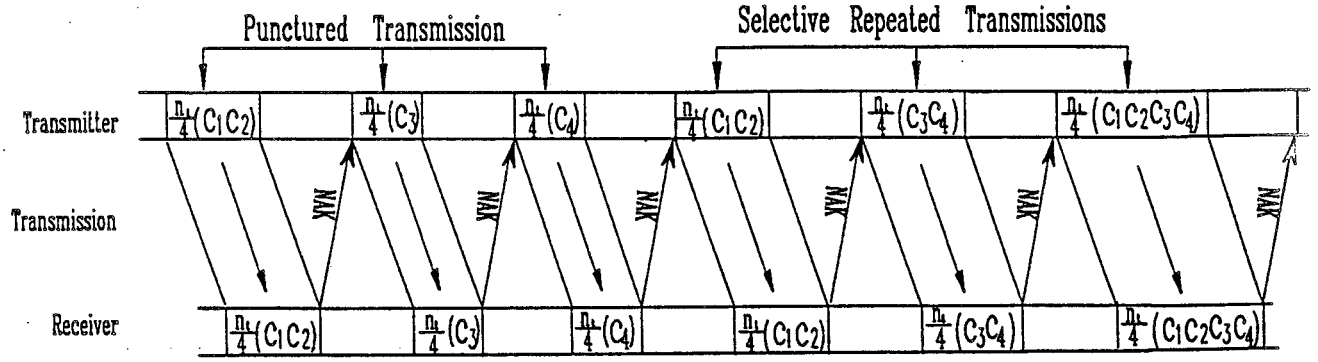



Fig. 3.12 The use of a sequence of rate compatible trellis codes in a hybrid ARQ/FEC error control system.

In the above figure,

$$n_t = n_i + n_c + \nu \tag{3.11}$$

where $n_t$ is the total packet length, $n_i$ is the number of information bits per packet, $n_c$ is the number of CRC bits in each packet, and $\nu$ is the number of trellis

terminating bits added to each packet. Since the puncturing period is P = 1 interval and there are 4 bits encoded per interval, there are altogether $\frac{n_t}{4}$ periods for each packet. In most situations, $n_t$ can vary from a few hundred to a few thousand bits.

As shown in Fig. 3.12, initially, the third and fourth trellis encoder output symbols, $c_3, c_4$, of each puncturing period, are punctured from the mother code in Fig. 3.11 and stored in the transmitter's buffer. Only $c_1, c_2$ are sent to the receiver. Since two symbols are transmitted for every 4 bits of information, the code with the highest effective rate, 2 bits/pulse, is used in the first attempt. When the channel is quiet, this high rate code will be powerful enough to deliver the packet free from errors. In this case, a positive acknowledgement (ACK) is sent to the transmitter, and the next packet is transmitted. However, if after decoding, errors are detected, a negative acknowledgement (NAK) is sent to the transmitter, and the transmitter will then send the third symbol, $c_3$, of each period. At the receiving end, the erroneous data of the first attempt are combined with the data received for the second attempt to form a rate 4/3 bits/pulse trellis code. If errors are detected at the receiver after decoding based on this rate 4/3 code, the fourth symbol of each period, $c_4$, will be sent. At this point, 4 symbols are sent for each 4 bits of information, and the receiver performs decoding based on the rate 1 bit/pulse code, i.e. the mother code. If the decoding is still not successful, then in the 4th attempt, $c_1, c_2$ in each period will be repeated and the rate 2/3 bit/pulse code is now in effect. If this code is still not powerful enough, then in the 5th attempt, $c_3, c_4$ are repeated to get a rate 1/2 bit/pulse code. Finally, if required, all the four symbols are repeated in the last attempt, bringing into effect the most powerful code in the sequence, the rate 1/3 bit/pulse code. One may note that until the 3rd attempt, the code rate is higher than the rate

of the mother code. It is the result of puncturing the mother code. After that, from 4th to 6th attempt, a set of lower rate codes, obtained by selectively repeating the mother code, are in effect. The last three attempts are actually based on the concept of code combining of Chase [21].

### 3.3.3 ARQ/FEC Protocol with RC - TCM

In general, the procedures for implementing an adaptive hybrid ARQ/FEC system that uses RC - TCM are summarized as follow.

1) For delivering a packet of information with $n_i$ bits, add $n_c$ CRC bits to the information sequence for purpose of error detection.

2) Add ν known bits to terminate the encoder trellis for each data block with $(n_i + n_c)$ bits.

3) Choose a mother trellis code, a puncturing ( same as the selective repeating ) period, a maximum number of retransmissions for each data packet, and a set of code rates.

4) Choose ( heuristically or by computer search ) a puncturing rule that results in a sequence of good high rate codes by puncturing the mother code, and (or) choose a selective repeating rule that yields a sequence of good low rate codes from the mother code. "Good" here means each rate compatible trellis code should have an error performance close to that of the best possible TCM with the same code rate, and similar complexity.

5) Use the encoder of the mother code to encode the data sequence of length $n_t = n_i + n_c + \nu$ bits and store the encoded sequence at the transmitter.

6) Transmit the codewords of the highest rate code in the first attempt. Transmit the previously punctured symbols or repeat the symbols of the mother code when retransmissions are required. Accumulate *all* received symbols at the receiver for the purpose of forming a more powerful code in the next decoding attempt.

7) Decode the received sequence with a soft decision Viterbi decoder based on the current effective error correction code.

8) After FEC, check the syndrome of the error detecting code. If the syndrome is zero, output the decoded message to the user. An ACK signal is sent to the transmitter and the transmitter releases the next packet of information. If the syndrome is not zero, repeat steps (6) - (8) until the maximum number of retransmission is reached.

From the analysis and simulation results given in the next chapter, one can see that the adaptive error control systems using RC - TCM schemes provide desirable throughput performances.

# CHAPTER FOUR

# PERFORMANCE OF HYBRID ARQ/FEC SYSTEMS USING RC - TCM

In this chapter, we present analytic and simulation results for a rate adaptive, hybrid ARQ/FEC system that uses RC - TCM schemes. First, we present the system model for our analysis. Since our intended area of application of this new technique is mobile radio systems, we will analyze the bit error probability of RC - TCM in the Rayleigh fading channel, using a calculation technique recently developed by Cavers and Ho [22]. Next, we make use of the bit error probability to determine the throughput of our error control system. The calculated throughput is then compared with simulation results. The throughput performance of several error control systems that use different RC - TCM are given, including some cases that use selective repeating of Ungerboeck's codes. Finally, we compare the performance of our system with that of Hagenauer's.

## 4.1 THROUGHPUT EVALUATION

### 4.1.1 System Model

For mobile radio application, our system model is the one shown in Fig. 4.1.

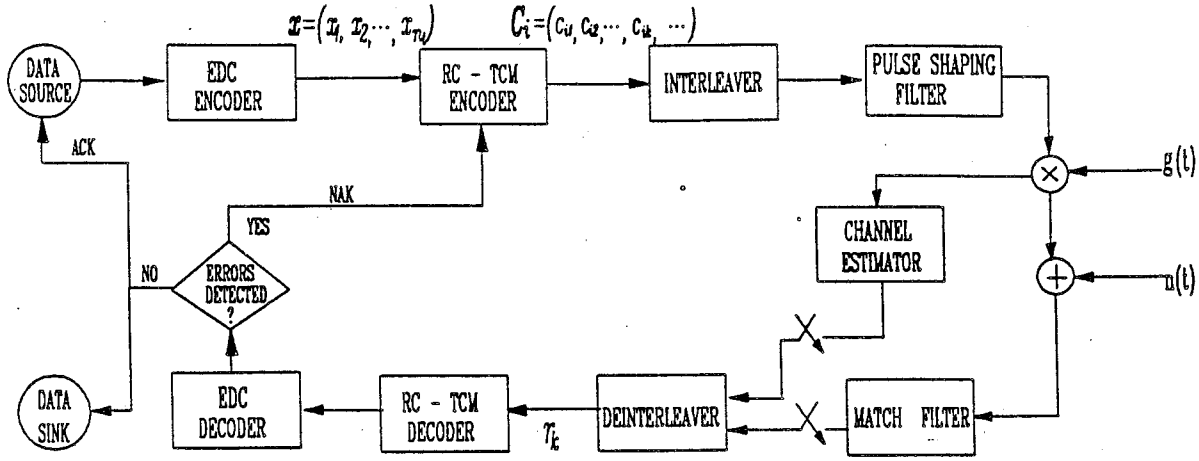



Fig. 4.1 System model for RC - TCM in mobile radio channels.

where in Fig. 4.1, $n_c$ CRC bits are first added to $n_i$ information bits through a $(n_c + n_i,\ n_i)$ error detecting code encoder, and then ν bits for terminating the encoder trellis at a known state are added to form the data sequence $\boldsymbol{x} = (x_1, x_2, \cdots, x_{n_t})$. Therefore, $\boldsymbol{x}$ includes $n_i$ information bits, $n_c$ CRC bits and ν terminating bits, and $n_t = n_i + n_c + \nu$. For convenience, we call $\boldsymbol{x}$ the information sequence to be transmitted for each message packet. Then, $\boldsymbol{x}$ is encoded through the encoder of a mother RC - TCM, and the encoded sequence is stored at the transmitter buffer. A puncturing and (or) selective repeating rule controls the transmission and the decoding of data in each attempt. Although the FEC used in different attempts are different, the same RC - TCM encoder and decoder pair (of the mother code) is used in different attempts. This is because of the rate compatibility of these FEC. Since only incremental redundancy are sent in each retransmission, encoding of the message $\boldsymbol{x}$ needs only to be done once. In other words, no matter which attempt we are referring to, $\boldsymbol{x}$ is the information sequence corresponding to the codeword $\boldsymbol{C}_i = (c_{i1}, \cdots, c_{ik}, \cdots)$. Here in

this study, the $c_{ik}$'s are 8PSK symbols shown in the signal constellation in Fig. 3.2 (b). As such, each $c_{ik}$ is a complex number of the form $(\cos(\frac{2\pi}{8}i), \sin(\frac{2\pi}{8}i))$, $i = 0, 1, \ldots, 7$. We assume the interleaver in Fig. 4.1 has an infinite interleaving depth, so that after de - interleaving, the $c_{ik}$'s are affected independently by the channel fading process g(t). It has been found that in many cases [31], a reasonably large interleaving depth is almost as good as an infinite interleaving depth. The impulse response of a pulse shaping filter in Fig. 4.1 satisfies Nyquist's criterion for zero intersymbol interference. Since we are dealing with a Rayleigh channel, the fading process g(t) is a zero mean, complex Gaussian random process with a variance of $\sigma_g^2 = \frac{1}{2}\overline{|g(t)|^2}$. The overbar here represents the expected value. We assume the process g(t) is slow enough compared to the signalling rate so that g(t) can be represented by a constant value in each signalling interval. Let $g_k$ be the value of g(t) in the $k$th signalling interval, i.e. $[kt, (k+1)T]$. Then, after match filtering, sampling and deinterleaving, the received sample for the $k$th symbol in the codeword sent is

$$r_k = g_k c_{ik} + n_k \tag{4.1}$$

where $n_k$ is a filtered noise term. Since the unfiltered (complex baseband) channel additive white Gaussian noise n(t) has a power spectral density (PSD) of $N_0$, it follows that each $n_k$ has a variance of $N_0$ as well. All the $n_k$'s are independent and identically distributed (i.i.d.) random variables. Due to the interleaving / de-interleaving process, the $g_k$'s are also i.i.d. random variables. The $g_k$'s are independent of the $n_k$'s.

We assume in this study that the receiver has perfect knowledge of the $g_k$'s, i.e. ideal channel state information (CSI). Usually, CSI can be obtained via pilot tone

or pilot symbol assisted detection. With a small expense of bandwidth, the pilot symbols assisted detection technique in [43], can provide almost the performance of an ideal channel estimator. Note that by assuming perfect CSI, we are implicitly assuming also perfect coherent reception at the receiver. Under the assumptions of ideal interleaving and perfect CSI, the decoding rule used by the decoder in Fig. 4.1 is to compute the metrics

$$\delta_j = \sum_k | r_k - g_k c_{jk} |^2 \qquad j = 1, 2, \cdots \tag{4.2}$$

and select the codeword $\hat{C}_i = (\hat{c}_{i1}, \hat{c}_{i2}, \cdots, \hat{c}_{ik}, \cdots)$, whose metric is the largest. The probability of a decoding error will be given later.

### 4.1.2 Throughput Analysis

As mentioned earlier, we use the stop - and - wait (SNW) protocol in our present study. In half duplex channels, the waiting time for an acknowledgement in SNW ARQ systems are wasted. However, the distinguishing features of a land mobile radio system is that a pair of frequencies is shared by many users in different time periods. This makes the SNW protocol suitable for this application, because a transmitter can make use of the idle time for waiting the feedback acknowledgement from one receiver to transmit to another receiver. For example, after the base station transmits a message to a mobile, it may immediately send a message to any other mobile without interfering with the previous mobile's acknowledgement on the feedback channel. The same is true for the messages transmitted from a mobile to other mobiles through a base station.

We analyze below the throughput performance of a rate adaptive hybrid ARQ/FEC scheme that uses a stop - and - wait transmission strategy and our proposed RC - TCM technique. The throughput analysis will be based on the error probability analysis of TCM of Cavers and Ho [22]. The assumptions here are the same as that we used in analyzing the throughput performance of systems that use RCPCC. These assumptions can be found in Section 2.2

As shown in Fig. 4.1, $n_t$ is the total number of bits to be transmitted in each message packet. It includes $n_i$ information bits, $n_c$ CRC bits and ν bits for terminating the encoder trellis. Let $S_{av}$ denote the average number of transmitted M-ary coded modulation symbols for delivering each packet with $n_t$ bits. Then, the average throughput efficiency of the system that uses RC - TCM can be written as:

$$R_{av} = \frac{n_i}{n_t} \cdot \frac{n_t}{S_{av}} = \frac{n_i}{S_{av}} \qquad \text{(bit/pulse)} \qquad (4.3)$$

where $n_t/S_{av}$ is the average effective code rate of the FEC. To obtain $S_{av}$, let's define $S_i$ as the number of symbols accumulated at the receiver for the $i$th decoding attempt, and let $P_{F_i}$ be the probability that after the $i$th attempt, errors are still present in the decoded word. Also let $P_{C_i} = 1 - P_{F_i}$ be the probability that the $i$th attempt is successful, i.e. after FEC, no errors are detected. Note that the index $i$ in $P_{F_i}$ and $P_{C_i}$ can vary from 1 to K, where K is the designed maximum number of transmissions for each message packet. K should be chosen so that the maximum decoding delay introduced is tolerable and the frame error rate (FER) is sufficiently small. Here, by frame error rate, we mean the probability that errors still exist after the last decoding attempt. Since in our proposed systems, all the data received for previous decoding

attempts are used for the current decoding attempt, the errors associated with the K decoding attempts are not independent of each other. However, it is very hard to find the correlation function between the results of these K decoding attempts. To carry on our throughput analysis, we use the approximation that the results of the success (or failure) of the K decoding attempts are independent. From the analytical results we obtained (shown in the later sections), we found that the inaccuracy caused by this optimistic approximation is small. Let $P_{F_0} = 1$, then we have:

$$S_{av} \cong \sum_{i=1}^{K} S_i P_{C_i} \left( \prod_{j=0}^{i-1} P_{F_j} \right) + S_K \prod_{i=1}^{K} P_{F_i} \tag{4.4}$$

The remaining work is to find the $P_{F_i}$'s, the failure probabilities.

Recall $P_{F_i}$ is defined as the probability that after the $i$th attempt in FEC, the Viterbi decoder's output still contains errors, or the failure probability of the $i$th decoding attempt. Generally speaking, the bit errors at the output of a Viterbi decoder are not independent and the distributions of error patterns are very difficult to explore. For convenience, we use the approximation that the bit errors are independent. This means

$$P_{F_i} \cong 1 - (1 - P_{b_i})^{n_t} \tag{4.5}$$

where $P_{b_i}$ is the bit error probability after the $i$th decoding attempt, $n_t = n_i + n_c + \nu$ is the number of bits per packet.

The above approximation is quite appropriate, since at large signal to noise ratio condition, the error probability will be dominated by the shortest error events and

usually only a single bit error is associated with the shortest error events.

The probability that the *ith* decoding attempt is successful is :

$$P_{C_i} = 1 - P_{F_i} = (1 - P_{b_i})^{n_i} \tag{4.6}$$

and the frame error probability is:

$$FER = \prod_{i=1}^{K} P_{F_i} = \prod_{i=1}^{K} \left[ 1 - (1 - P_{b_i})^{n_i} \right] \tag{4.7}$$

for a maximum of K decoding attempts per packet.

Equations (4.4) to (4.6) indicate that to find the throughput of our rate - adaptive error control system, we must first find the bit error probability, $P_{b_i}$, of each rate compatible trellis code used by the system. As mentioned in Section 3.1.2, for any two codewords $C_i = (c_{i1}, c_{i2}, \cdots, c_{ik}, \cdots)$ and $C_j = (c_{j1}, c_{j2}, \cdots, c_{jk}, \cdots)$ of a trellis code, (the i, j here no longer denote different codes or different attempts ), we have the upperbound on the pairwise error event probability

$$P(i \to j) \le \left\{ \prod_{k \in \eta} \left( \frac{E_s}{4N_0} \right) | c_{ik} - c_{jk} |^2 \right\}^{-1}$$

From $P(i \to j)$, we can obtain an upperbound on the bit error probability of a trellis codes by taking the sum of the product of the probability of each error event and the number of bit errors associated with that error event. Due to the nature of the Chernoff bound [17], the resulting upperbound on the bit error probability is usually quite loose. If we use this loose upperbound on the bit error probability in our throughput analysis, we will get a loose lowerbound on $R_{av}$. In this study, we make

use of the exact pairwise error event probability expression derived by Cavers and Ho [22] to obtain a tight lower bound on the bit error probability. This tight lower bound in turns give us tight upper bound on the throughput $R_{av}$.

For the coding / decoding system shown in Fig. 4.1, if $C_i = (c_{i1}, c_{i2}, \cdots, c_{ik}, \cdots)$ and $C_j = (c_{j1}, c_{j2}, \cdots, c_{jk}, \cdots)$ are two codewords of the trellis coded PSK system, then according to [22], the probability of confusing these two codewords is

$$P(i \to j) = - \sum_{RHP\ poles} Residues\left\{\frac{\Phi(s)}{s}\right\} \tag{4.8}$$

where s represents any number in the complex plane, RHP means the right hand complex plane,

$$\Phi(s) = \left\{\prod_{k \in \eta}\left(\frac{E_s}{4N_0}\right)d_k^2\right\}^{-1}\left\{\prod_{k \in \eta}\frac{-1}{4(s-p_{1k})(s-p_{2k})}\right\} \quad , \tag{4.9}$$

$$d_k^2 = |\, c_{ik} - c_{jk}\,|^2 \quad , \tag{4.10}$$

$$p_{1k} = \frac{1}{2} - \sqrt{\frac{1}{4} + \frac{1}{(E_s/N_0)d_k^2}} \quad , \tag{4.11}$$

$$p_{2k} = \frac{1}{2} + \sqrt{\frac{1}{4} + \frac{1}{(E_s/N_0)d_k^2}} \quad , \tag{4.12}$$

and $\eta$ is the set of k for which $d_k^2 \neq 0$. If $C_i$ and $C_j$ are different only in $d_H$ places, i.e. if the Hamming distance between $C_i$ and $C_j$ is $d_H$, then, equation (4.8) can be written as

$$P(i \to j) = \left(\frac{E_s}{N_0}\right)^{-d_H} \left\{ \sum_{RHP\ poles} Residues\left[ \frac{1}{s} \prod_{k \in \eta} \frac{-1}{d_k^2 (s - p_{1k})(s - p_{2k})} \right] \right\} \tag{4.13}$$

As can be seen from (4.11) and (4.12), $p_{1k}$ is always less than 0 while $p_{2k}$ is always greater than 0. As such, the $p_{2k}$'s are all the right hand plane poles. To find $P(i \to j)$, the residues at the $p_{2k}$'s must be calculated. It is well known that if $\Phi(s)/s$ has a pole $p$ of order m, i.e. $\Phi(s)/s$ is a multiple of $1/(s-p)^m$, then the residue at $p$ is

$$Residue\left\{ \frac{\Phi(s)}{s}\bigg|_{s=p} \right\} = \frac{1}{(m-1)!} \lim_{s \to p} \frac{d^{m-1}}{ds^{m-1}} \left\{ (s-p)^m \frac{\Phi(s)}{s} \right\} \tag{4.14}$$

For those $\Phi(s)/s$'s having only 1*st* order poles, equation (4.14) can be determined by hand. If $\Phi(s)/s$ has a large number of higher order poles, a computer routine can be written to calculate the residue. Because of the form of $\Phi(s)/s$, the routine is recursive and hence the residues can be calculated rather easily.

Recall that our main interest here are the bit error probabilities, $P_{b_i}$, of different decoding attempts, or in other words, the bit error probability of each of the different RC - TCM schemes used by the system. As suggested in [22], a good estimate of the bit error probability of a TCM scheme can be obtained by taking only a small set of short error events into account:

$$P_b \approx \frac{1}{b} \sum_j m_{ij} P(i \to j) \tag{4.15}$$

where b is the number of input bit per encoding interval, and $m_{ij}$ is the error coefficient, or the number of bit errors associated with each error event.

Unlike an ordinary trellis code, which has a " stationary " distance structure, ( same distance structure can be observed by looking into the trellis starting from any point), a rate compatible trellis code with puncturing ( and/or repeating ) period of P intervals may have P different distance spectra and hence P different sets of error coefficients. Therefore, one should consider each of the P starting points for diverging paths in the decoding trellis, calculate the corresponding bit error probability, and then take the average over the P bit error probabilities to get $P_{bP}$, the bit error probability of a RC - TCM scheme, as shown below

$$P_{bP} \approx \frac{1}{P} \sum_{i_P=1}^{P} \frac{1}{b} \sum_{j} m_{ij}(i_P) P_{i_P}(i \rightarrow j) \tag{4.16}$$

Here $P_{i_P}(i \rightarrow j)$ is the pairwise error event probability and $\{m_{ij}(i_P)\}$ is the set of error coefficients associated with the error events when the $i_P th$ starting point for diverging paths within each puncturing and repeating period is considered.

By applying the above method to several examples, we found that in most cases that at relatively low SNR, the free Euclidean distance error event has the strongest effect on $P_b$ (or $P_{bP}$). On the other hand, at large SNR, as mentioned before, the free Hamming distance error event dominates the bit error probability. Thus we concluded that by considering one free Hamming distance error event and one free Euclidean distance error event, a fairly indicative estimate of the system throughput efficiency curve can be obtained. This will be discussed in more detail in the next section. In summary, the analytical result of the throughput performance of a hybrid ARQ/FEC system using RC - TCM can be obtained by utilizing equations (4.3) to (4.16).

## 4.2 EXAMPLES AND SIMULATION RESULTS

We will now analyze the throughput performance of rate adaptive hybrid ARQ/FEC systems that use different RC - TCM schemes.

### 4.2.1 System I --

### RC - TCM Schemes Obtained From A 4 State MTCM

We showed in Table 3.1 a sequence of RC - TCM schemes obtained from a 4 state multiple trellis code. To analyze the throughput performance of the hybrid ARQ/FEC system that uses the rate compatible codes in Table 3.1 ( for convenience, we will call this system *system I* in later discussion ), we must first calculate the bit error probability of each of these codes.

When we calculate the bit error probability of a coded PSK system, for convenience, we may choose the all zero-phase codeword as the transmitted codeword $C_t$. However, we would like to point out that in general, TCM is a nonlinear modulation scheme and the bit error probability is codeword dependent.

According to Fig. 3.11 and Table 3.1, the puncturing period P is one encoding interval. In this case, the free Hamming distance error event path for each of the 6 rate compatible codes has a length measured by the number of non-zero 8PSK symbols within one branch. This is because of the existence of parallel paths. The free Euclidean distance error event paths for all but the second code ( rate 4/3 ) in this example are the same, and it is highlighted in Fig. 4.2.

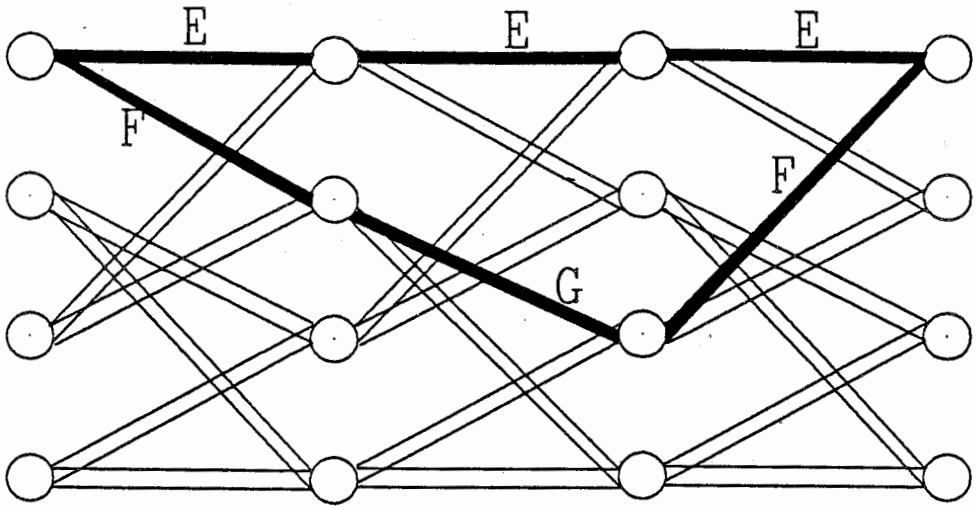



Fig. 4.2 The free Euclidean distance error event paths of a set of rate compatible trellis codes.

For the second code ( rate 4/3 ) in Table 3.1, the minimum Euclidean distance error event path is the same as the shortest error event path.

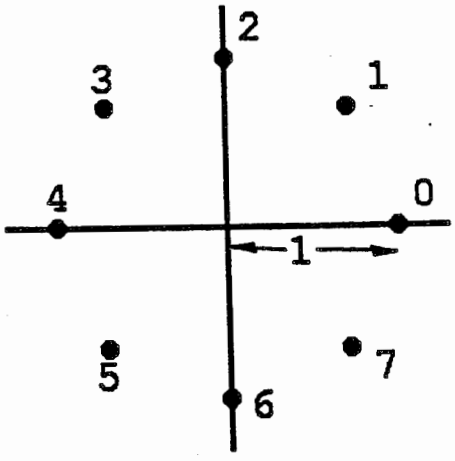



Fig. 4.3 8PSK signal constellation.

Recall the first code in Table 3.1 is obtained by deleting the last 2 coded symbols in each interval of the rate 1 bit/pulse mother code in Fig. 3.11. In this case, we have found, with the aid of a computer program, the shortest error event ( with the smallest product of squared Euclidean distance ) with respect to the all zero - phase word ( 0, 0, . . . ) in Fig. 3.11 is the word ( 1, 5 ) or ( 7, 3 ), or ( 5, 1 ), or ( 3, 7 ). In complex notation, we have the transmitted codeword

$$C_i = (1, 1, \cdots)$$

and the erroneous codeword

$$C_j = (e^{j\pi/4}, e^{j5\pi/4}, 1, \cdots)$$

where $e^{jn\pi/4}$ corresponds to the $n$th signal point in the 8PSK signal constellation in Fig. 4.3. According to (4.13), the pairwise error event probability is

$$P(i \to j) = -\left(\frac{E_s}{N_0}\right)^{-2} \left\{ \sum_{RHP\ poles} Residues\left[ \frac{1}{s} \prod_{k=1}^{2} \frac{-1}{d_k^2 (s - p_{1k})(s - p_{2k})} \right] \right\} \qquad (4.17)$$

where the right hand side poles are $p_{2k}$'s with k = 1, 2, and they are all first order poles. From (4.10), we have

$$d_1^2 = | 1 - e^{j\pi/4} |^2 = 0.586 \qquad (4.18a)$$

$$d_2^2 = | 1 - e^{j5\pi/4} |^2 = 3.414 \qquad (4.18b)$$

and hence

$$\begin{bmatrix} p_{11} \\ p_{21} \end{bmatrix} = \frac{1}{2} \mp \sqrt{\frac{1}{4} + \frac{1}{0.586(E_s/N_0)}} \qquad (4.19a)$$

$$\begin{bmatrix} p_{12} \\ p_{22} \end{bmatrix} = \frac{1}{2} \mp \sqrt{\frac{1}{4} + \frac{1}{3.414(E_s/N_0)}} \qquad (4.19b)$$

For all the rate compatible codes used in this system, 4 information bits are encoded at each time interval. For the first code ( rate 2 bits/pulse ) which we are now concerned about, the mapping of information bits to channel symbols at each state are shown in Table 4.1.

| Information bits | Channel symbols $1 \rightarrow 1,\ 3 \rightarrow 2$ | Information bits | Channel symbols $2 \rightarrow 3,\ 4 \rightarrow 4$ |
|---|---|---|---|
| 0000 | 00 | 0000 | 02 |
| 0001 | 15 | 0001 | 17 |
| 0010 | 22 | 0010 | 24 |
| 0011 | 37 | 0011 | 31 |
| 0100 | 44 | 0100 | 46 |
| 0101 | 51 | 0101 | 53 |
| 0110 | 66 | 0110 | 60 |
| 0111 | 73 | 0111 | 75 |
| | $1 \rightarrow 2,\ 3 \rightarrow 1$ | | $2 \rightarrow 4,\ 4 \rightarrow 3$ |
| 1000 | 04 | 1000 | 06 |
| 1001 | 11 | 1001 | 13 |
| 1010 | 26 | 1010 | 20 |
| 1011 | 33 | 1011 | 35 |
| 1100 | 40 | 1100 | 42 |
| 1101 | 55 | 1101 | 57 |
| 1110 | 62 | 1110 | 64 |
| 1111 | 77 | 1111 | 71 |

Table 4.1 The mapping of information bits to channel symbols for a rate 2 bits/pulse RC - TCM code, where $i \rightarrow j$, ( i = 1, 2, 3, 4, and j = 1, 2, 3, 4 ) denotes the signal set assignment from state i to state j, see Fig. 3.11.

From Table 4.1 we see there are 4 free Hamming distance error events, (1, 5), (3, 7), (5, 1), and (7, 3), that have the minimum product of squared distances $\Pi d^2$. Each of these 4 error events corresponds to different number of bit errors. By taking the average, we have 2 bits of information error in the shortest error event of this code. Since P = 1 in this example, equation (4.16) is equivalent to equation (4.15). If we only consider the shortest error event ( free Hamming distance error event ) in our bit error probability analysis in equation (4.15), then we have the lower bound

$$P_{b_1} > 2 \cdot \frac{1}{4} P(i \to j) \tag{4.20}$$

where $P(i \to j)$ is given in equation (4.17). Next, we will consider the free Euclidean distance error event. Once again, with the help of a computer search program, we found

$$C_j = (e^{j\pi/4}, e^{j\pi/4}, e^{j\pi/4}, e^{j7\pi/4}, e^{j\pi/4}, e^{j\pi/4}, 1, \cdots) \quad ,$$

$$C_j = (e^{j7\pi/4}, e^{j7\pi/4}, e^{j\pi/4}, e^{j7\pi/4}, e^{j\pi/4}, e^{j\pi/4}, 1, \cdots) \quad ,$$

$$C_j = (e^{j\pi/4}, e^{j\pi/4}, e^{j\pi/4}, e^{j7\pi/4}, e^{j7\pi/4}, e^{j7\pi/4}, 1, \cdots) \quad ,$$

and

$$C_j = (e^{j7\pi/4}, e^{j7\pi/4}, e^{j\pi/4}, e^{j7\pi/4}, e^{j7\pi/4}, e^{j7\pi/4}, 1, \cdots)$$

are closest to the transmitted codeword $C_i = (1, 1, \cdots)$. These four error events have the same product of squared distance, 0.04, and the same squared Euclidean distance of 3.515. The pairwise error event probability $P(i \to j)$ corresponding to each of these free Euclidean error events is

$$P(i \to j) = -\left(\frac{E_s}{N_0}\right)^{-6} \left\{ \underset{on\ p_{21}}{Residue} \left[ \frac{1}{s} \left( \frac{-1}{d_1^2 (s - p_{1k})(s - p_{2k})} \right)^6 \right] \right\} \qquad (4.21)$$

where $d_1^2$, $p_{11}$, and $p_{21}$ are the same as those given in (4.18a) and (4.19a). As can be seen from (4.21), we have to find the residue at a $6th$ order pole. From Table 4.1, there are on average 7 bits of information errors associated with the free Euclidean distance error event (i.e. the term $m_{ij}$ in equation (4.15) becomes 7). So if only the free Euclidean distance error event is used in calculating the bit error probability of the 1st decoding attempt, $P_{b_1}$, we have the following lower bound

$$P_{b_1} > 7 \cdot \frac{1}{4} P(i \to j) \qquad (4.22)$$

where $P(i \to j)$ is given in (4.21). A tight lower bound on $P_{b_1}$ can be obtained as follows

$$P_{b_1} > \max[\ P_1,\ P_2\ ]$$

where $P_1$ and $P_2$ are the lower bounds in (4.20) and (4.22) respectively. Note that the error curves of $P_1$ and $P_2$ will intersect due to the difference in diversity order and distance parameters. In summary, we use the following approach in estimating the bit error probability. For each TCM scheme, we identify the shortest error event (free Hamming distance error event with smallest product of squared distance) and the free Euclidean distance error event. We then calculate the exact bit error probability of these two events. A tight lower bound is then obtained by taking at each SNR, $E_s/N_0$, the maximum of these two bit error probabilities. Finally, the tight lower bound on bit error probability will be converted to a tight upper bound on the throughput according to equations (4.3) to (4.5). For a RC-TCM scheme with P >1,

we first calculate the lower bound of bit error probability associated with each of the P starting points in the decoding trellis. And then we use the average of these P lower bounds on $P_b$ to calculate the throughput efficiency of the system. We list in Table 4.2 the distance parameters of the RC - TCM schemes in Table 3.1.

| | Along the free **Hd** path | | | Along the free **Ed** path | | |
|---|---|---|---|---|---|---|
| | $d_{fH}$ | $\prod_{k=1}^{d_{fH}} d_k^2$ | $d_E^2$ | $d_H$ | $\prod_{k=1}^{d_H} d_k^2$ | $d_{fE}^2$ |
| 1st | 2 | 2.0 | 4.0 | 6 | 0.04 | 3.515 |
| 2nd | 3 | 1.1715 | 4.586 | 3 | 1.1715 | 4.586 |
| 3rd | 4 | 4.0 | 8.0 | 12 | $1.633\times10^{-3}$ | 7.03 |
| 4th | 6 | 8.0 | 12.0 | 18 | $6.596\times10^{-5}$ | 10.545 |
| 5th | 8 | 16.0 | 16.0 | 24 | $2.665\times10^{-6}$ | 14.06 |
| 6th | 12 | 64.0 | 24.0 | 36 | $4.351\times10^{-9}$ | 21.09 |

Table 4.2 Distance parameters of the RC - TCM schemes in system I.

In the above table, $d_H$ is the Hamming distance ( in channel symbols ) between the two codewords in the free Euclidean distance error event, $d_E^2$ is the squared Euclidean distance between the two codewords in the free Hamming distance error event. From (4.13), one can predict that at relatively low SNR, the free Euclidean distance error event has a stronger effect on the system error performance, but at high SNR, the error performance is dominated by the shortest error event due to the fact that these two error events have different Hamming distances, and hence they will intersect at

some SNR. We plot in Fig. 4.4 (a) and (b) the bit error probability associated with the free Hamming distance and free Euclidean distance error events for the 1*st* and the 3*rd* code in Table 3.1. From Fig. 4.4, we see a lower bound on bit error probability can be obtained by taking the larger of the 2 error curves at each SNR.

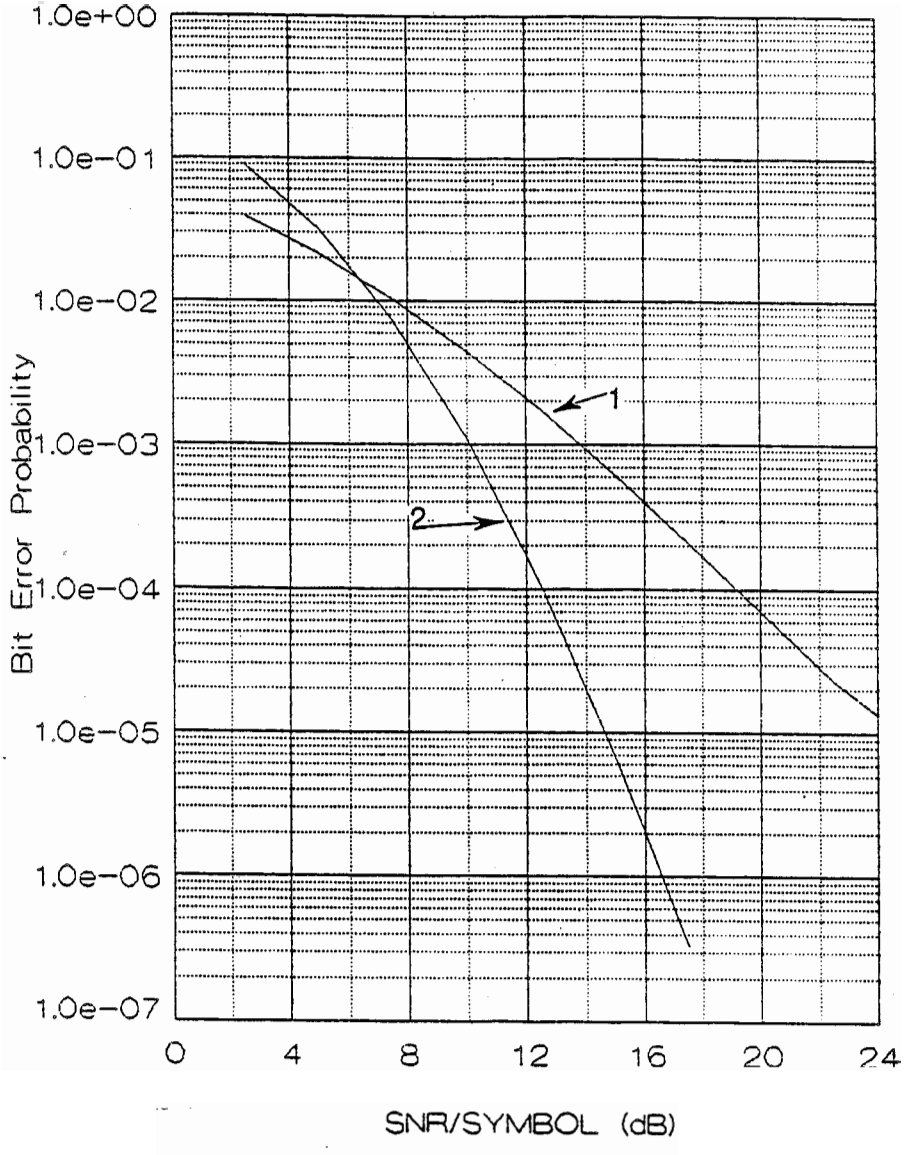



Fig. 4.4(a) Bit error probabilities associated with the free Hamming distance error event (curve 1) and the free Euclidean distance error event (curve 2) for the 1*st* code in Table 3.1.

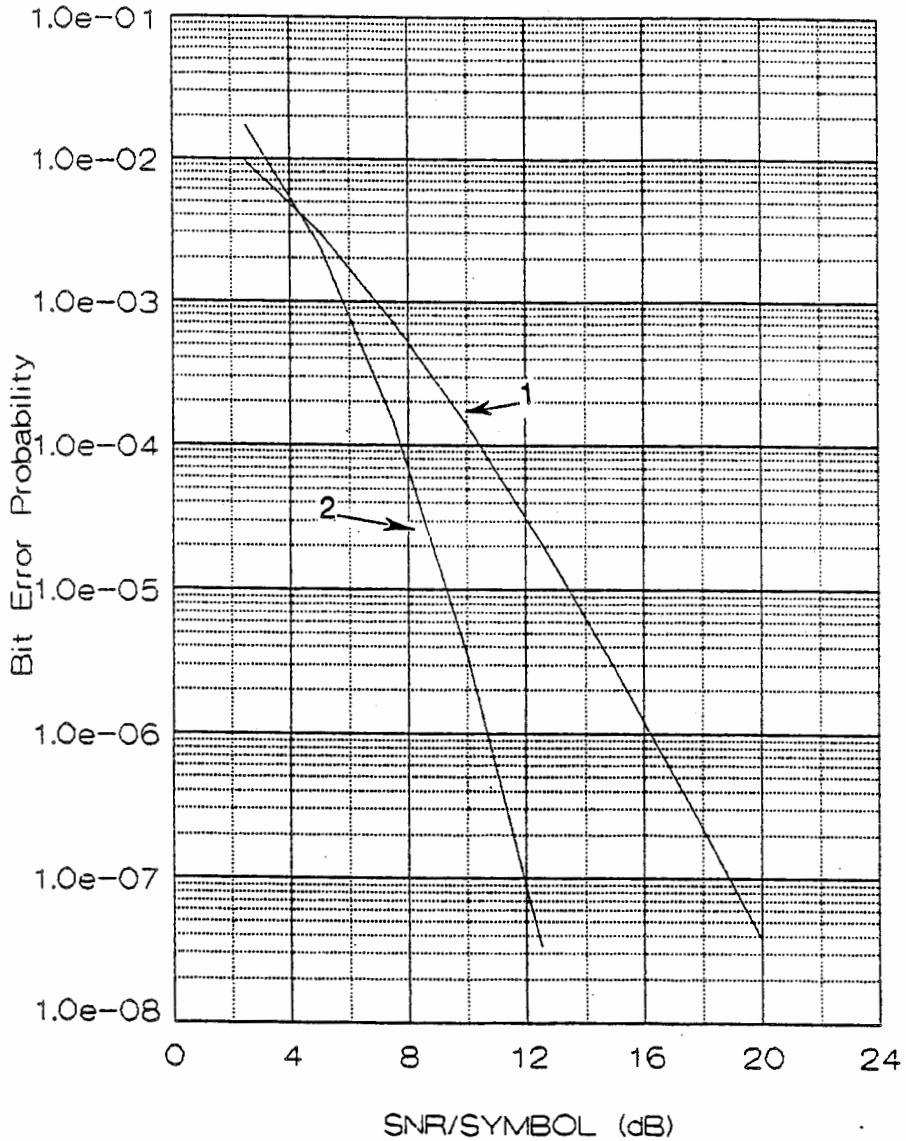



Fig. 4.4(b) Bit error probabilities associated with the free Hamming distance error event (curve 1) and the free Euclidean distance error event (curve 2) for the 3*rd* code in Table 3.1.

Once we have calculated the lower bounds $P_{b_i}$ for each code in Table 3.1, we can then lower bound $P_{F_i}$ in equation (4.5). These lower bounds on $P_{F_i}$ can in turn be used in (4.4) to obtain the lower bound on $S_{av}$, the average number of symbols transmitted in this hybrid ARQ/FEC system for successfully delivering each message packet with $n_i$ bits. For the RC - TCM schemes in Table 3.1, the associated $S_{av}$ is

$$
\begin{aligned}
S_{av} = & \frac{n_t}{2} P_{C_1} \\
& + \left( \frac{n_t}{2} + \frac{n_t}{4} \right) P_{F_1} P_{C_2} \\
& + \left( \frac{n_t}{2} + \frac{n_t}{4} + \frac{n_t}{4} \right) P_{F_1} P_{F_2} P_{C_3} \\
& + \left( \frac{n_t}{2} + \frac{n_t}{4} + \frac{n_t}{4} + \frac{n_t}{2} \right) P_{F_1} P_{F_2} P_{F_3} P_{C_4} \\
& + \left( \frac{n_t}{2} + \frac{n_t}{4} + \frac{n_t}{4} + \frac{n_t}{2} + \frac{n_t}{2} \right) P_{F_1} P_{F_2} P_{F_3} P_{F_4} P_{C_5} \\
& + \left( \frac{n_t}{2} + \frac{n_t}{4} + \frac{n_t}{4} + \frac{n_t}{2} + \frac{n_t}{2} + n_t \right) P_{F_1} P_{F_2} P_{F_3} P_{F_4} P_{F_5}
\end{aligned}
\tag{4.23}
$$

$S_{av}$ can then be substituted in (4.3) to obtain an upperbound on $R_{av}$. This upperbound is shown in Fig. 4.5 for the case where $n_i = 380$, $n_c = 32$, and $\nu = 8$ (i.e. $n_t = 420$). Note that in [14], Hagenauer uses the same set of parameters in his study of RCPCC. Also shown in Fig. 4.5 are the simulation results of system I (using the rate compatible trellis codes in Table 3.1) as well as the simulation result taken from [14]. Clearly, our upperbound on throughput is quite close to our simulation results, indicating our analytical technique for estimating the bit error probability of trellis coded PSK is

quite appropriate.

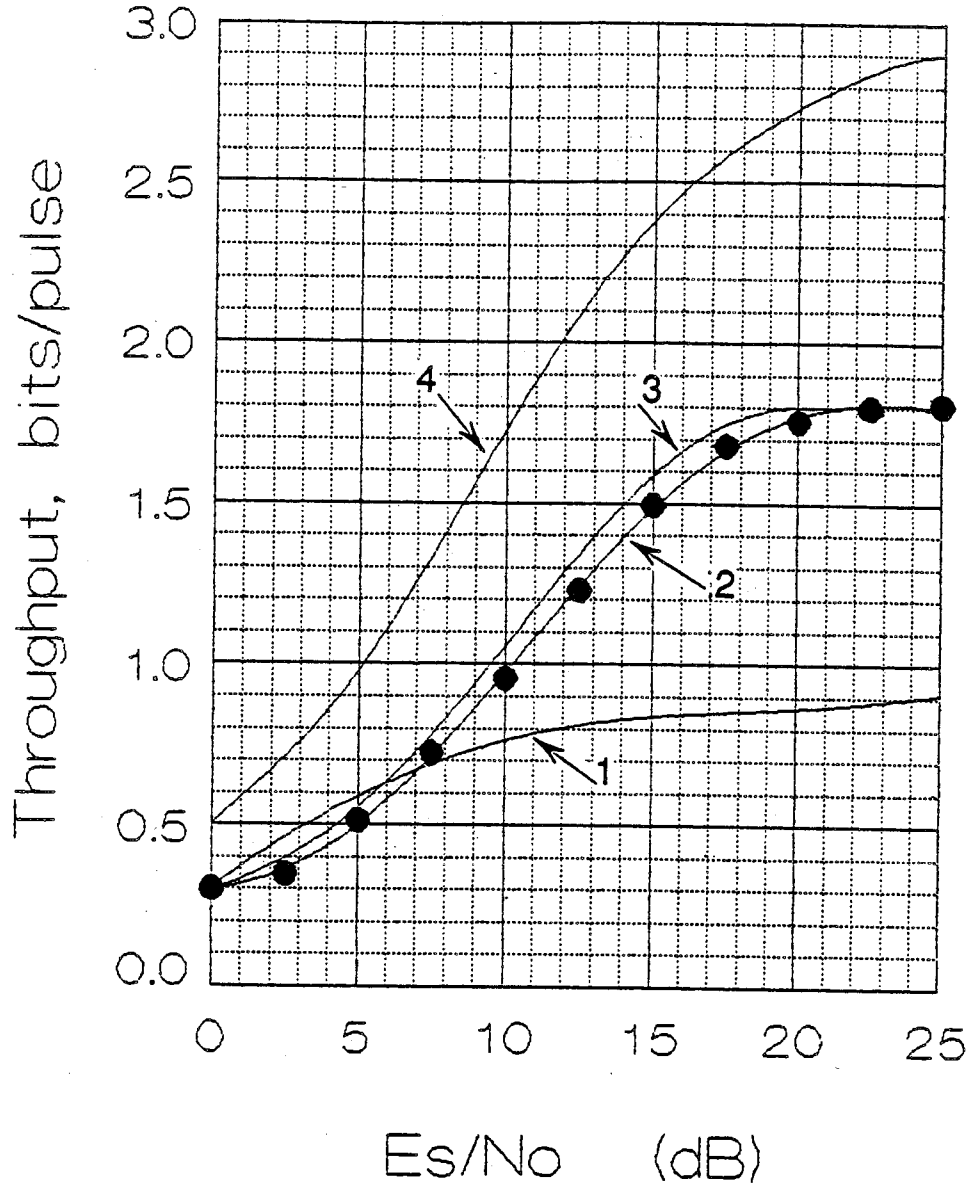



Fig. 4.5 Throughput performance of system I, simulation result (curve 2); analytical result (curve 3); Hagenauer 's system (curve 1); cut off rate for 8PSK (curve 4).

Note that in our simulation, we always generate 1000 packets at each SNR. By comparing the throughput curves of system I ( using the rate compatible trellis codes in Table 3.1 ) with that of Hagenauer's system, we see a significant improvement on throughput at SNR above 7 dB. As a matter of fact, for 15 dB and up, the throughput of system I is about twice that of Hagenauer's system. For 7 dB or less, though, system I has a slightly poorer throughput.

Suppose that both systems are used in a mobile radio environment, where the energy of the signal deteriorates with the increase in distance ($r$) between the transmitter and the receiver at a rate of $r^{-4}$. Assume N users are uniformly distributed around a base station in a circular cellular coverage area with radius $r_{max}$, ( see Fig. 4.6 ), and the marginal signal to noise ratio, $E_{min}/N_0$, is 5 dB. Then, $\frac{2\pi r\,dr}{\pi r_{max}^2} \cdot N$ is the number of users that lie in the annular region bounded by the circles of radii $r$ and $r + dr$, and the signal to noise ratio, $E_s/N_0$, at

$$r = \frac{3}{4} r_{max}, \quad r = \frac{1}{2} r_{max}, \quad r = \frac{7}{16} r_{max}, \quad \text{and} \quad r = \frac{1}{4} r_{max}$$

turns out to be 10dB, 17dB 20dB and 29dB respectively. In such a system, different subsystems (each of them is made up of one of the users and the base station) may have different throughput which depends on the location of the users. Therefore it is appropriate to calculate the system wide throughput by using equation (4.24).

$$\int_0^{r_{max}} \frac{2\pi r}{\pi r_{max}^2} \cdot N \cdot R_{av}(r) dr \tag{4.24}$$

The results show that system I has an average system throughput gain of 51 % over Hagenauer's system.

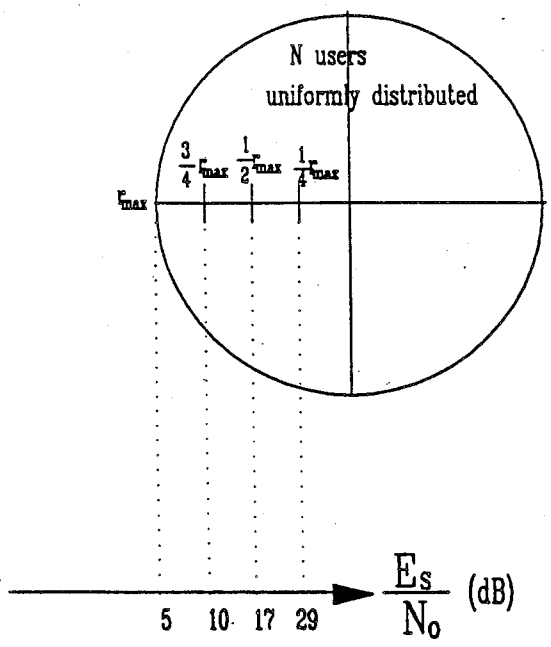



Fig. 4.6 A circular cellular coverage area with radius $r_{max}$

Compared to the cut off rate, or $R_0$ curve of 8PSK, which is also shown in Fig. 4.5, the throughput curve of system I is relatively far away from $R_0$. Recall that the throughput curve of Hagenauer's system is very close to the $R_0$ curve for 2PSK, see Fig. 2.6. This relatively large separation from $R_0$ prompted us to search for better RC - TCM schemes for our rate adaptive hybrid ARQ/FEC system.

As mentioned before, besides throughput, an important performance measure of a hybrid ARQ/FEC system is its frame error rate. For this purpose, we show in Fig. 4.6 the analytical result of the frame error rate of system I by using equation (4.7). Also shown in Fig. 4.7 is the frame error rate of Hagenauer's system. We see that the expense we paid for having a large throughput at larger SNR is a larger frame error rate. This point further motivates us to search for better RC - TCM schemes.

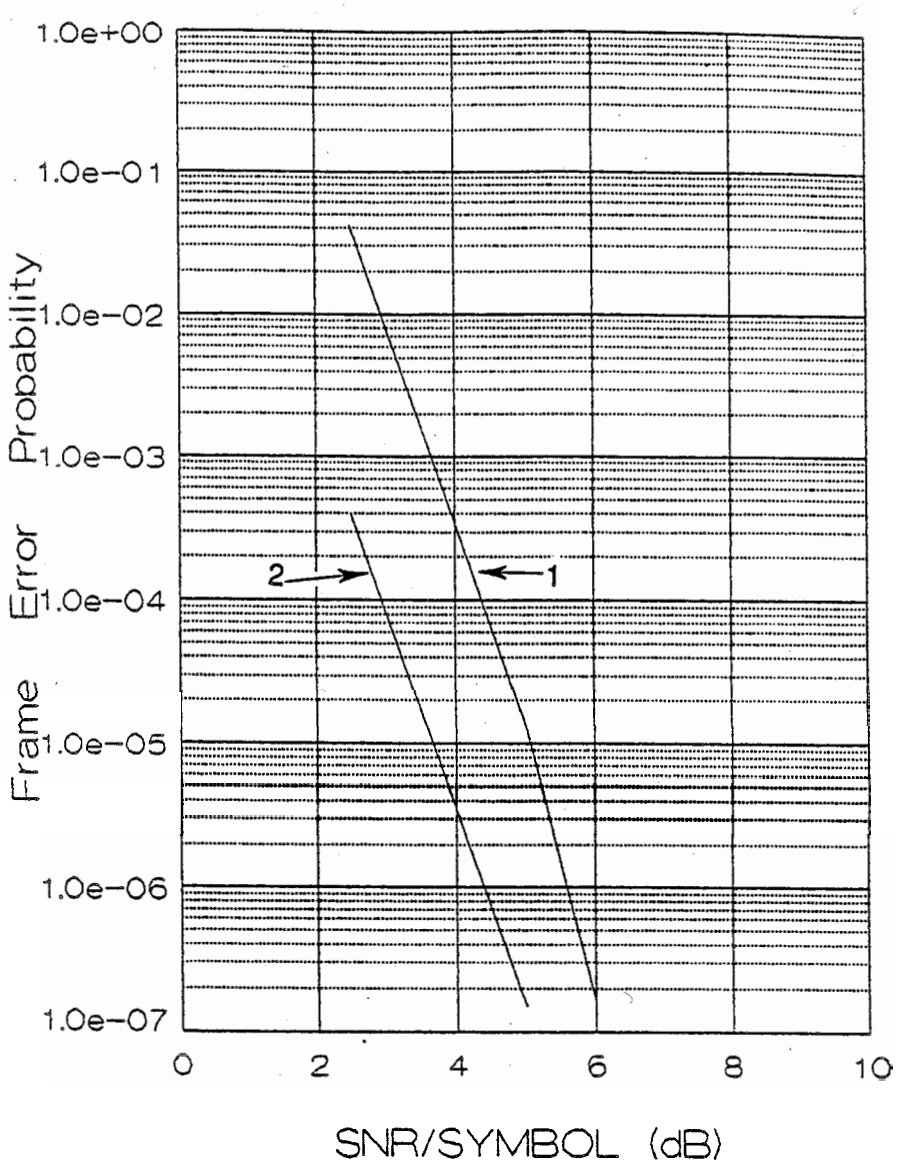



Fig. 4.7 Comparison of the frame error rate of system I (curve 1) and Hagenauer's system (curve 2).

### 4.2.2 System II --

### RC - TCM Schemes obtained From An Ungerboeck's 8 State Code

In Section 3.3.1, we pointed out that Ungerboeck's codes, due to their "non-puncturable" property, are not eligible to be used as mother codes to generate

sequences of rate compatible codes with higher rates. However, we found a class of RC - TCM schemes, formed by simply selective repeating Ungerboeck's codes, provide rather good performance.

Let us look at the trellis diagram of the 8 state Ungerboeck's code in Fig. 3.2(a). We choose this code as the mother code to generate a sequence of rate compatible trellis codes with lower rates by selectively repeating this mother code. The resulting RC - TCM schemes can then be applied to our adaptive hybrid ARQ/FEC system. Here suppose we want the code rates of the rate compatible trellis codes to range from 2 bits/pulse to 1/3 bit/pulse. The code rate can be adaptively changed from the highest rate to the lowest rate in a maximum of 6 attempts by adopting the selective repeating strategy given in Table 4.3. Here the " repeating period " P is set to 2, i.e. the encoding intervals are divided into periods of 2 intervals and the pulses in each period are selectively repeated during different attempts.

| Symbols transmitted per puncturing period in the *i*th transmission | Symbols used in the *i*th decoding attempt | The effective code rate at the *i*th decoding attempt (bit/pulse) |
|---|---|---|
| 1st $c_1\, c_2$ | $c_1^*\ \ c_2^*$ | 4 bits / 2 pulses $\Rightarrow$ 2 |
| 2nd $c_1$ | $c_1\ \ c_1^*\ \ c_2$ | 4 bits / 3 pulses $\Rightarrow$ 4/3 |
| 3rd $c_2$ | $c_1\ \ c_1\ \ c_2\ \ c_2^*$ | 4 bits / 4 pulses $\Rightarrow$ 1 |
| 4th $c_1\, c_2$ | $c_1\, c_1\, c_1^*\, c_2\, c_2\, c_2^*$ | 4 bits / 6 pulses $\Rightarrow$ 2/3 |
| 5th $c_1\, c_2$ | $c_1\ \ c_1\ \ c_1\ \ c_1^*$<br>$c_2\ \ c_2\ \ c_2\ \ c_2^*$ | 4 bits / 8 pulses $\Rightarrow$ 1/2 |
| 6th $c_1\, c_1\, c_2\, c_2$ | $c_1\, c_1\, c_1\, c_1\, c_1^*\, c_1^*$<br>$c_2\, c_2\, c_2\, c_2\, c_2^*\, c_2^*$ | 4 bits /12 pulses $\Rightarrow$ 1/3 |

Table 4.3 RC - TCM schemes obtained from the 8 state Ungerboeck's code. * indicates the symbol sent in the *i*th transmission.

Let $c_1 c_2$ be the two symbols in each repeating period of the mother code shown in Fig. 3.10. To obtain a sequence of lower rate codes, the approach is the same as the repeating procedure used in the example of section 3.3 2.

Applying the same method as we used in Section 4.2.1, we calculated the exact bit error probabilities for both the free Hamming distance error event and the free Euclidean distance error event for each of the 6 rate compatible trellis codes in this hybrid ARQ/FEC system ( we will call it *system II* ). Based on these two bit error probabilities, a lower bound on the overall bit error probability, $P_{b_1}$, was obtained. Since P = 2, we should consider 2 starting points for diverging paths in the decoding trellis when we calculate the bit error probabilities. In this example, all but the second code have the same set of error events for the 2 starting points of the decoding trellis. For the *2nd* code, the average of the bit error probabilities of the 2 free Hamming distance error events and the average of the bit error probabilities of the 2 free Euclidean distance error events were used to calculate $P_{b_2}$. Then, by applying (4.5) and (4.4), we obtained the upperbound on the throughput as shown in Fig. 4.8. The corresponding simulation result plotted in Fig. 4.8 agrees well with the analytical result.

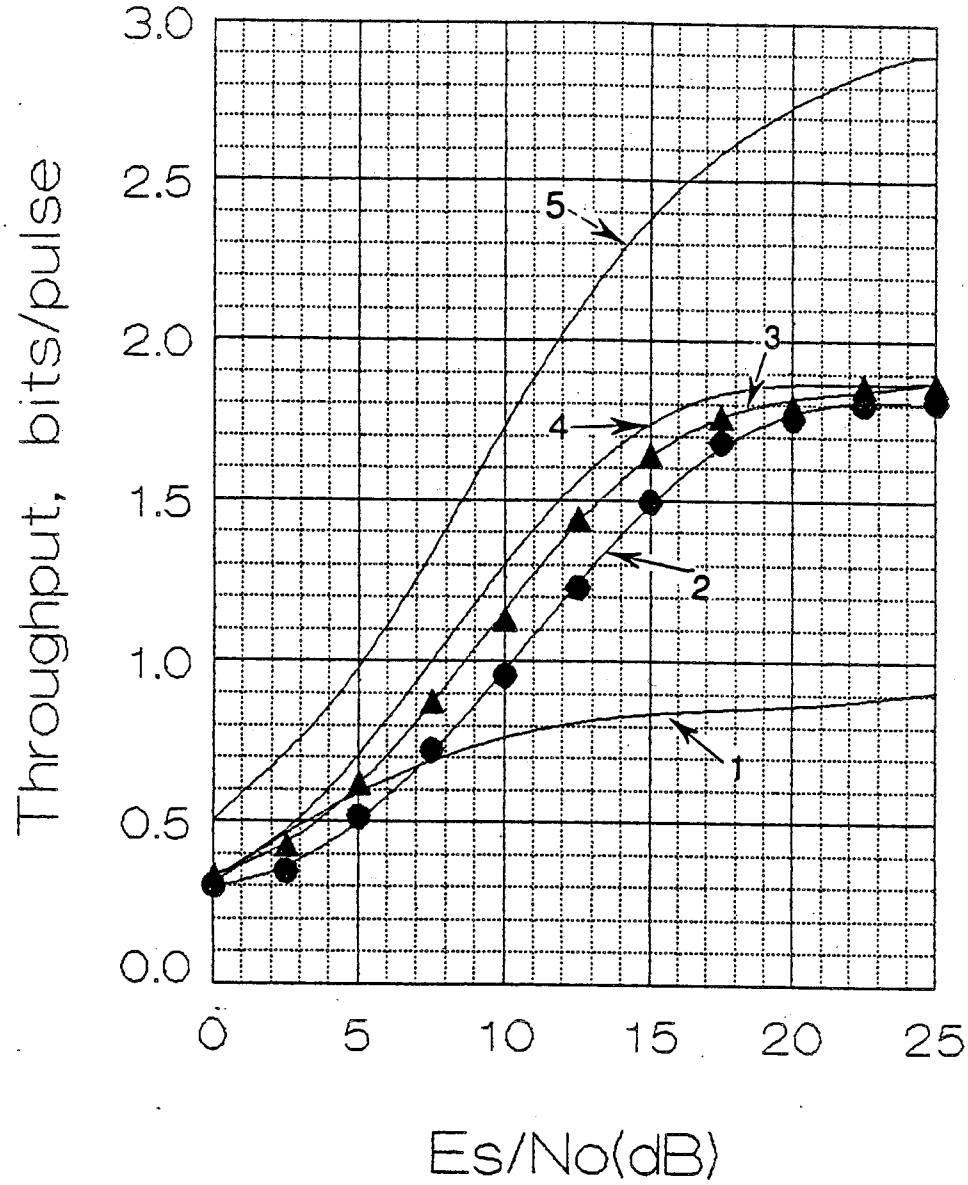



Fig. 4.8 Throughput performance of system II, simulation result (curve 3); analytical result (curve 4); the throughput of Hagenauer's system (curve 1); throughput of system I (curve 2); cut off rate for 8PSK (curve 5).

From this figure, we see that system II, which uses the RC - TCM schemes in Table 4.3, achieves some improvement in throughput over system I given in section 4.2.1. To see the reason of this improvement, we list in Table 4.4 the distance properties of the RC - TCM schemes in Table 4.3.

| | Along the free **Hd** path | | | Along the free **Ed** path | | |
|---|---|---|---|---|---|---|
| | $d_{fH}$ | $\prod_{k=1}^{d_{fH}} d_k^2$ | $d_E^2$ | $d_H$ | $\prod_{k=1}^{d_H} d_k^2$ | $d_{fE}^2$ |
| 1st | 2 | 8.0 | 6.0 | 3 | 2.343 | 4.586 |
| 2nd | 3 | 16.0 | 8.0 | 5 | 2.745 | 7.172 |
| | 3 | 32.0 | 10.0 | 4 | 1.373 | 5.172 |
| 3rd | 4 | 64.0 | 12.0 | 6 | 5.490 | 9.172 |
| 4th | 6 | 512.0 | 18.0 | 9 | 12.865 | 13.757 |
| 5th | 8 | 4096.0 | 24.0 | 12 | 30.144 | 18.343 |
| 6th | 12 | 262114.0 | 36.0 | 18 | 165.499 | 27.515 |

Table 4.4 Distance parameters of the RC - TCM schemes obtained from the 8 state Ungerboeck's code.

According to the design criteria for TCM schemes for Rayleigh fading channels given in section 3.1.2, the larger the distance parameters, $d_{fH}$ and $\prod_{k=1}^{d_{fH}} d_k^2$, the better the system performance. When we compare Table 4.4 with Table 4.2, we find that the RC - TCM schemes used in system I and system II have the same set of free Hamming distances. In this case, the second important parameter,

$$\prod_{k=1}^{d_{fH}} d_k^2$$

makes a difference in the performances of the two systems. As shown in Table 4.2 and Table 4.4, system II has a set of much larger $\prod_{k=1}^{d_{fH}} d_k^2$'s than that of system I.

As shown in Fig. 4.8, system II outperforms Hagenauer's system in throughput efficiency at SNR above 4 dB. At SNR less than 4 dB, system II has about the same throughput efficiency as that of system I. At SNR above 16 dB, the throughput of system II is twice that of Hagenauer's system. The throughput curve of system II tracks the $R_0$ curve better than that of system I. Based on Fig. 4.8 and equation (4.24), we calculated the system - wide throughput gain of system II over Hagenauer's system, when they are both used in the mobile radio environment of Fig. 4.6. The result shows that an 68 % throughput gain can be achieved by choosing system II over Hagenauer's system.

Also, the analysis indicated that the frame error rate obtained by system II is about the same as that of Hagenauer's system, see Fig. 4.9.

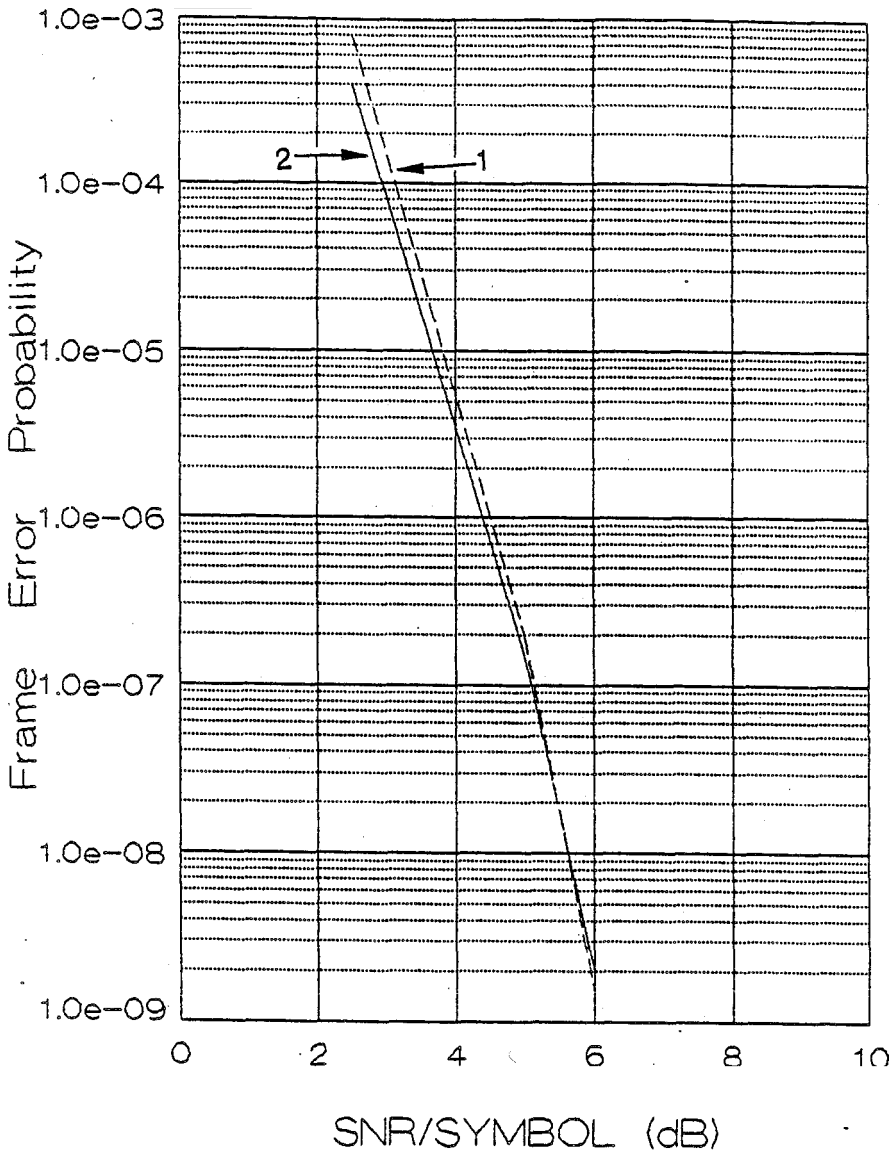



Fig. 4.9 Frame error rate of system II (curve 1); Hagenauer's system (curve 2).

From the above description, we see that a sequence of fairly good RC - TCM can be obtained by periodically selective repeating the 8 state Ungerboeck's code ( in Fig. 3.2(a) ).

### 4.2.3 System III --

### RC - TCM Schemes obtained From An Ungerboeck's 16 State Code

Recall in Section 3.3.1, we mentioned that the 16 state, rate 2 bits/pulse, Ungerboeck's code shown in Fig. 3.10 is a very attractive coded modulation scheme for both Gaussian and Rayleigh fading channels. Suppose we use the same selective repeating rule as used in system II to generate a sequence of RC - TCM based on this 16 state Ungerboeck's code. The resulting RC - TCM schemes have code rates ranging from 2 bits/pulse to 1/3 bit/pulse. The effective code rate can be adaptively changed from the highest rate to the lowest rate in a maximum of 6 attempts by adopting the selective repeating strategy given in Table 4.5. Here again, the " repeating period " P is set to 2, i.e. the encoding intervals are divided into periods of 2 intervals and the pulses in each period are selectively repeated during different attempts.

| Symbols transmitted per puncturing period in the $i$th transmission | Symbols used in the $i$th decoding attempt | The effective code rate in the $i$th decoding attempt (bit/pulse) |
|---|---|---|
| 1st $c_1\, c_2$ | $c_1^*\ \ c_2^*$ | 4 bits / 2 pulses $\Rightarrow$ 2 |
| 2nd $c_1$ | $c_1\ \ c_1^*\ \ c_2$ | 4 bits / 3 pulses $\Rightarrow$ 4/3 |
| 3rd $c_2$ | $c_1\ \ c_1\ \ c_2\ \ c_2^*$ | 4 bits / 4 pulses $\Rightarrow$ 1 |
| 4th $c_1\, c_2$ | $c_1\, c_1\, c_1^*\, c_2\, c_2\, c_2^*$ | 4 bits / 6 pulses $\Rightarrow$ 2/3 |
| 5th $c_1\, c_2$ | $c_1\ \ c_1\ \ c_1\ \ c_1^*$<br>$c_2\ \ c_2\ \ c_2\ \ c_2^*$ | 4 bits / 8 pulses $\Rightarrow$ 1/2 |
| 6th $c_1\, c_1\, c_2\, c_2$ | $c_1\, c_1\, c_1\, c_1\, c_1^*\, c_1^*$<br>$c_2\, c_2\, c_2\, c_2\, c_2^*\, c_2^*$ | 4 bits /12 pulses $\Rightarrow$ 1/3 |

Table 4.5 RC - TCM schemes obtained from the 16 state Ungerboeck's code. * indicates the symbol sent at the $i$th transmission.

Same as what we did to system II, we calculated the exact bit error probabilities for both the free Hamming distance error event and the free Euclidean distance error event for each of the 6 rate compatible trellis codes in this hybrid ARQ/FEC system (we will call it *system III*). Based on these two bit error probabilities, a lower bound on the overall bit error probability, $P_{b_i}$, was obtained. Since P=2, we should consider 2 starting points for diverging paths in the decoding trellis when we calculate the bit error probabilities. In this example, all but the second code have the same set of error events for the 2 starting points of the decoding trellis. For the second code, the average of the bit error probabilities of the 2 free Hamming distance error events and the average of the bit error probabilities of the 2 free Euclidean distance error events were used to calculate $P_{b_2}$. Then, by applying (4.5) and (4.4), we obtained the upperbound on the throughput of system III as shown in Fig. 4.10. Again, the corresponding simulation result plotted in Fig. 4.10 agrees well with the analytical result.

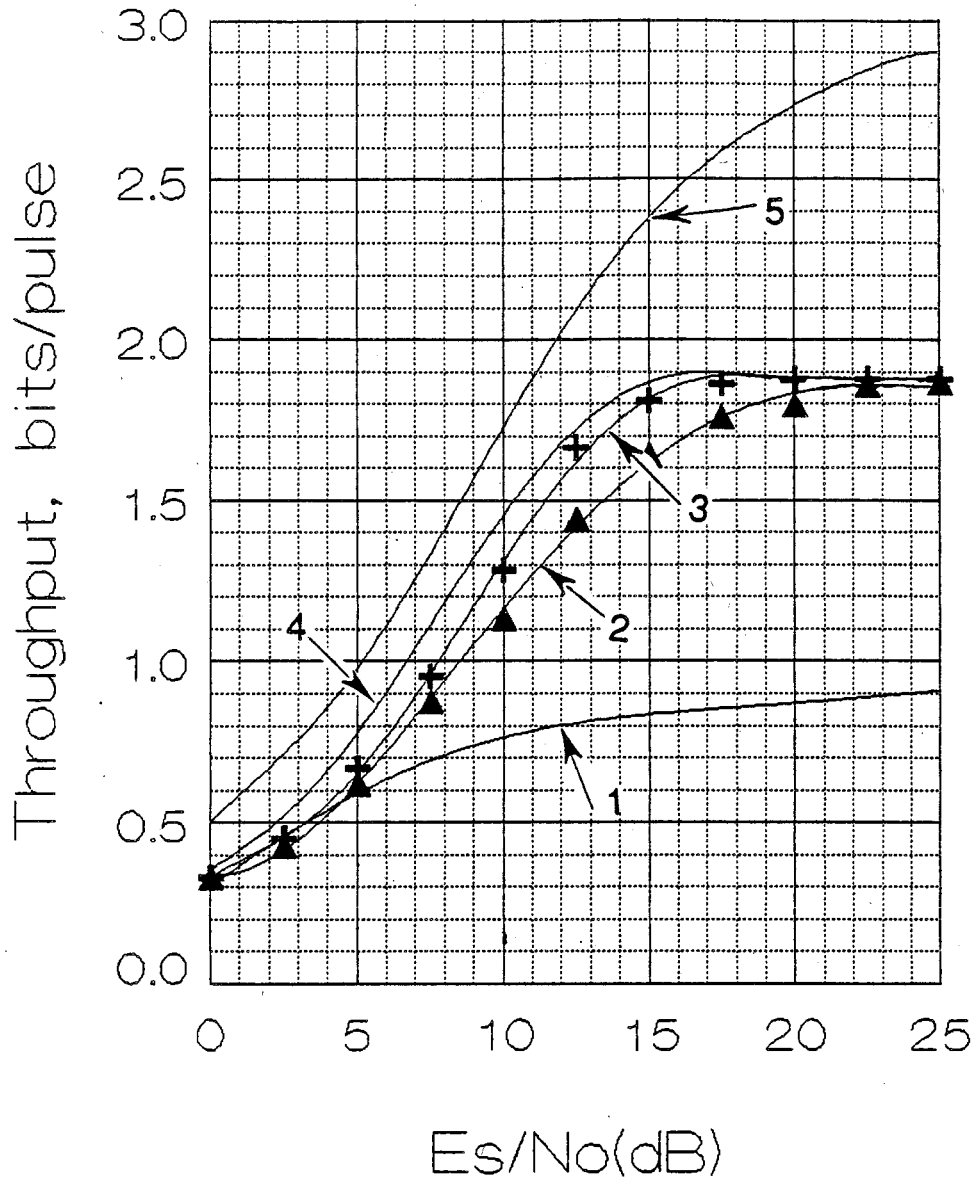



Fig. 4.10 Throughput performance of system III, simulation result (curve 3); analytical result (curve 4); the throughput of Hagenauer's system (curve 1); throughput of system II (curve 2); cut off rate for 8PSK (curve 5).

From Fig. 4.10, we see that system III has a better throughput performance than that of system II. This is because, with the increase in the number of encoder states,

the RC - TCM schemes used in system III have a set of larger $d_{fH}$'s than that in system II. Obviously, the trade - off in this case is the increase in decoding complexity. We will discuss and compare the decoding complexities of several adaptive hybrid ARQ/FEC systems later in section 4.3. Table 4.6 lists the distance parameters of system III.

| | Along the free **Hd** path | | | Along the free **Ed** path | | |
|---|---|---|---|---|---|---|
| | $d_{fH}$ | $\prod_{k=1}^{d_{fH}} d_k^2$ | $d_E^2$ | $d_H$ | $\prod_{k=1}^{d_H} d_k^2$ | $d_{fE}^2$ |
| 1st | 3 | 4.686 | 6.586 | 4 | 1.373 | 5.172 |
| 2nd | 5 / 4 | 37.488 / 2.745 | 12.586 / 7.172 | 6 / 6 | 1.61 / 1.61 | 7.758 / 7.758 |
| 3rd | 6 | 21.96 | 13.17 | 8 | 1.885 | 10.344 |
| 4th | 9 | 102.9 | 19.76 | 12 | 2.588 | 15.516 |
| 5th | 12 | 482.18 | 26.34 | 16 | 3.554 | 20.686 |
| 6th | 18 | 10588.0 | 39.51 | 24 | 6.70 | 31.032 |

Table 4.6 Distance parameters of the RC - TCM schemes obtained from the 16 state Ungerboeck's code.

Fig. 4.10 shows that system III outperforms Hagenauer's system in throughput efficiency over the entire range of SNR. At SNR above 12 dB, the throughput of system III is about twice that of Hagenauer's system. At SNR less than 15 dB, the throughput curve of system III tracks the $R_0$ curve quite well. Based on Fig. 4.10 and equation (4.24), we calculated the system - wide throughput gain of system III

over Hagenauer's system, when they are both used in the mobile radio environment of Fig. 4.6. The result shows that an 81 % throughput gain can be achieved by choosing system III over Hagenauer's system.

Also, the analysis indicated that a lower frame error rate can be obtained by system III when compared with Hagenauer's system, see Fig. 4.11.

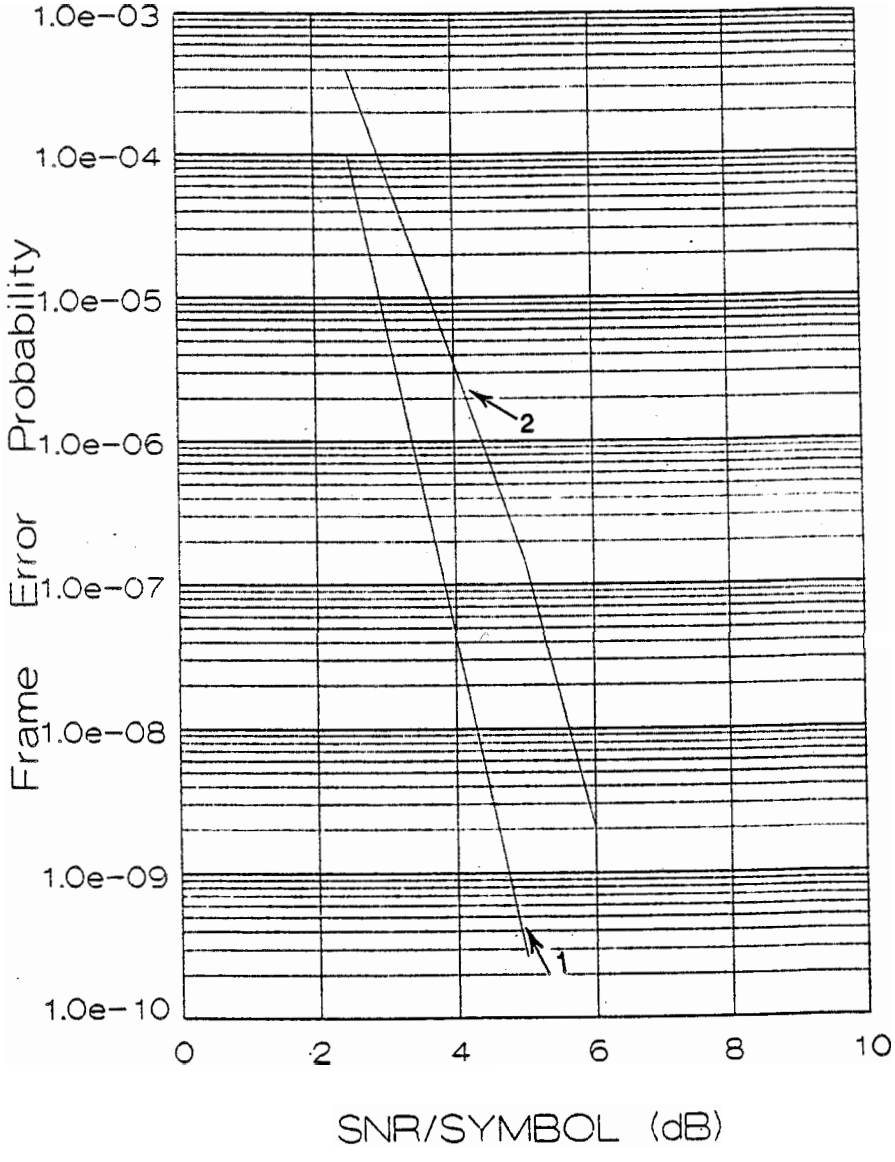



Fig. 4.11 Frame error rate of system III (curve 1); Hagenauer's system (curve 2).

So far we have shown 3 ARQ/FEC systems that use RC - TCM schemes outperform Hagenauer's system. However, we noted from Fig. 4.5, Fig. 4.8 and Fig. 4.10, the throughput efficiency of system I, system II and system III flatten out to 2 bits/pulse at high SNR, whereas the $R_0$ curve tells us that if we use 8PSK, a higher maximum throughput can be achieved. This is because the maximum effective rate of FEC, $R_{max}$, is 2 bits per channel symbol for these three systems. The actual maximum system throughput is slightly less than $R_{max}$ due to some inevitable overhead, such as the CRC bits.

### 4.2.4 System IV --

### RC - TCM Schemes Obtained From A 8 State MTCM

Following the results of the previous section, a sequence of RC - TCM with a larger $R_{max}$ should be designed in order to have a rate adaptive hybrid ARQ/FEC system with a throughput curve closer to the $R_0$ curve at high SNR than the three systems given in section 4.2.1, 4.2.2 and 4.2.3. Let's consider the following example.

It is a sequence of RC - TCM schemes obtained from an 8 state MTCM scheme proposed by Divaslar and Simon [26]. We will call the hybrid ARQ/FEC system that uses these RC - TCM schemes *system IV*. Fig. 4.12 gives the trellis diagram of Divsalar and Simon 's MTCM.

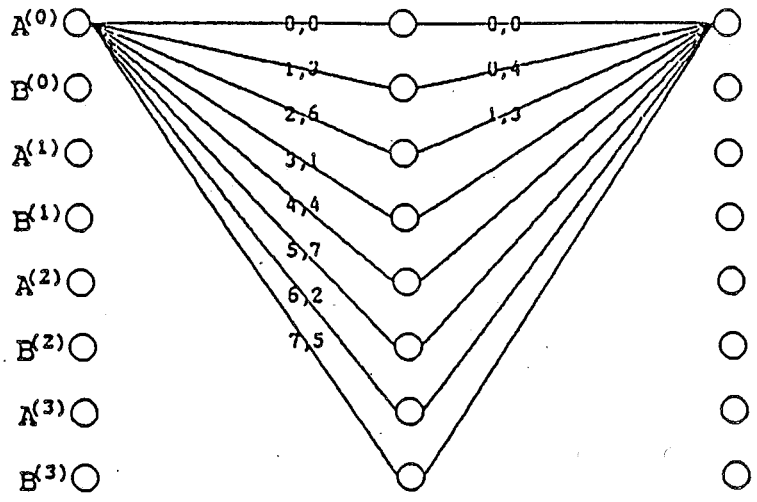



Fig. 4.12 Trellis structure of an 8 state MTCM scheme.

In this MTCM system, 3 information bits are encoded at each encoding interval and they are mapped into two 8PSK symbols. Thus the code rate is 1.5 (bits/pulse). We designed a sequence of RC - TCM schemes by using this code as the mother code and by using a puncturing (as well as repeating) period, P, of 4 intervals. With a maximum of 6 attempts for delivering each packet, we obtained the sequence of RC - TCM listed in Table 4.7.

| Symbols transmitted per puncturing period in the $i$th transmission | Symbols used in the $i$th decoding attempt | The effective code rate in the $i$th decoding attempt (bit/pulse) |
|---|---|---|
| 1st $c_1\ c_2\ \ c_4\ \ c_6\ \ c_8$ | $c_1^*\ c_2^*\ c_4^*\ c_6^*\ c_8^*$ | 12 bits / 5 pulses $\Rightarrow$ 2.4 |
| 2nd $\ \ c_3$ | $c_1\ c_2\ c_3^*\ c_4\ c_6\ c_8$ | 12 bits / 6 pulses $\Rightarrow$ 2 |
| 3rd $\ \ \ \ c_5\ \ c_7$ | $c_1\ c_2\ c_3\ c_4\ c_5^*\ c_6\ c_7^*\ c_8$ | 12 bits / 8 pulses $\Rightarrow$ 1.5 |
| 4th $\ \ c_2\ \ c_4\ \ c_6\ \ c_8$ | $c_1\ c_2\ c_2^*\ c_3\ c_4\ c_4^*$<br>$c_5\ c_6\ c_6^*\ c_7\ c_8\ c_8^*$ | 12 bits / 12 pulses $\Rightarrow$ 1 |
| 5th $\ c_1\ \ c_3\ \ c_5\ \ c_7$ | $c_1\ c_1^*\ c_2\ c_2\ c_3\ c_3^*\ c_4\ c_4$<br>$c_5\ c_5^*\ c_6\ c_6\ c_7\ c_7^*\ c_8\ c_8$ | 12 bits / 16 pulses $\Rightarrow$ 0.75 |
| 6th $\ c_1\ c_2\ c_3\ c_4\ c_5\ c_6\ c_7\ c_8$ | $c_1\ c_1\ c_1^*\ c_2\ c_2\ c_2^*\ c_3\ c_3\ c_3^*\ c_4\ c_4\ c_4^*$<br>$c_5\ c_5\ c_5^*\ c_6\ c_6\ c_6^*\ c_7\ c_7\ c_7^*\ c_8\ c_8\ c_8^*$ | 12 bits / 24 pulses $\Rightarrow$ 0.5 |

Table 4.7 RC - TCM schemes obtained from an 8 state MTCM with P = 4.
* indicates the symbol sent at the $i$th transmission.

In the above table, $c_1, c_2, \cdots c_8$ are the 8 symbols in each period of 4 intervals of the mother code shown in Fig. 4.12.

Applying the same method as we used in Section 4.2.1, we calculated the exact bit error probabilities of both the free Hamming distance error event and the free Euclidean distance error event for each of the 6 rate compatible trellis codes in this hybrid ARQ/FEC system. Based on these two bit error probabilities, lowerbounds on the overall bit error probabilities, $P_{b_i}$'s $i = 1$ to 6, can be obtained. Since P = 4 in system IV, we should consider 4 starting points for diverging paths in the decoding trellis when we calculate the bit error probabilities. In this example, only the first two codes have different sets of error events for the 4 starting points in the decoding

trellis. For the first two codes, the average of the bit error probabilities of the 4 free Hamming distance error events and the average of the bit error probabilities of the 4 free Euclidean distance error events were used to calculate $P_{b_1}$ and $P_{b_2}$. In Table 4.8, we list the distance parameters of the 6 RC - TCM schemes in Table 4.7, and 4 sets of distance parameters for each of the first two codes are included.

| | Along the free **Hd** path | | | Along the free **Ed** path | | |
|---|---|---|---|---|---|---|
| | $d_{fH}$ | $\prod_{k=1}^{d_{fH}} d_k^2$ | $d_E^2$ | $d_H$ | $\prod_{k=1}^{d_H} d_k^2$ | $d_{fE}^2$ |
| 1st | 3 | 8.0 | 8.0 | 3 | 1.172 | 4.586 |
| | 2 | 13.657 | 7.414 | 2 | 0.343 | 1.172 |
| | 2 | 13.657 | 7.414 | 2 | 0.343 | 1.172 |
| | 2 | 13.657 | 7.414 | 3 | 0.2 | 1.757 |
| 2nd | 3 | 8.0 | 8.0 | 4 | 0.686 | 5.172 |
| | 3 | 8.0 | 8.0 | 3 | 1.172 | 4.586 |
| | 2 | 13.657 | 7.414 | 2 | 0.343 | 1.172 |
| | 2 | 13.657 | 7.414 | 3 | 0.2 | 1.757 |
| 3rd | 3 | 8.0 | 8.0 | 4 | 0.686 | 5.172 |
| 4th | 5 | 109.255 | 15.414 | 6 | 0.235 | 6.343 |
| 5th | 8 | 64.0 | 16.0 | 8 | 0.471 | 10.343 |
| 6th | 9 | 512.0 | 24.0 | 12 | 0.323 | 15.515 |

Table 4.8 Distance parameters of the RC - TCM schemes in system IV.

After computing the $P_{b_i}$'s, for $i = 1$ to 6, the upperbound on the throughput efficiency of this system, as shown in Fig. 4.13, was obtained by substituting these probabilities into equations (4.5) and (4.4). Once again, the corresponding simulation result plotted in Fig. 4.13 agrees well with the analytical result.

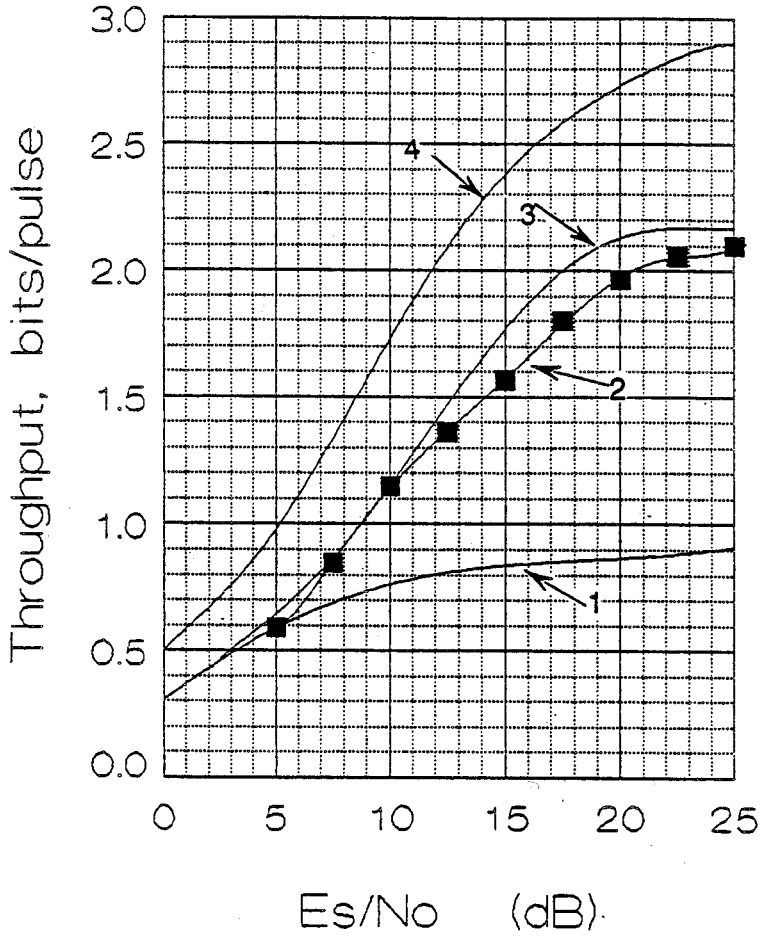



Fig. 4.13 Throughput of system IV, simulation result (curve 2); analytical result (curve 3); throughput of Hagenauer's system (curve 1); $R_0$ for 8PSK (curve 4).

As illustrated in Fig. 4.13, the maximum throughput of system IV reaches 2.1 bits/pulse, which is 2.3 times that of Hagenauer's system. Based on Fig. 4.13 and equation (4.24), we calculated the system - wide throughput gain of system IV over Hagenauer's system, when they are both used in the mobile radio environment of Fig. 4.6. A 74 % throughput gain is achieved by system IV over Hagenauer's system. The analytical result of the frame error rate of system IV, which is given in Fig. 4.14, shows the FER of this system is about 4 dB worse than that of Hagenauer's system at $10^{-5}$.

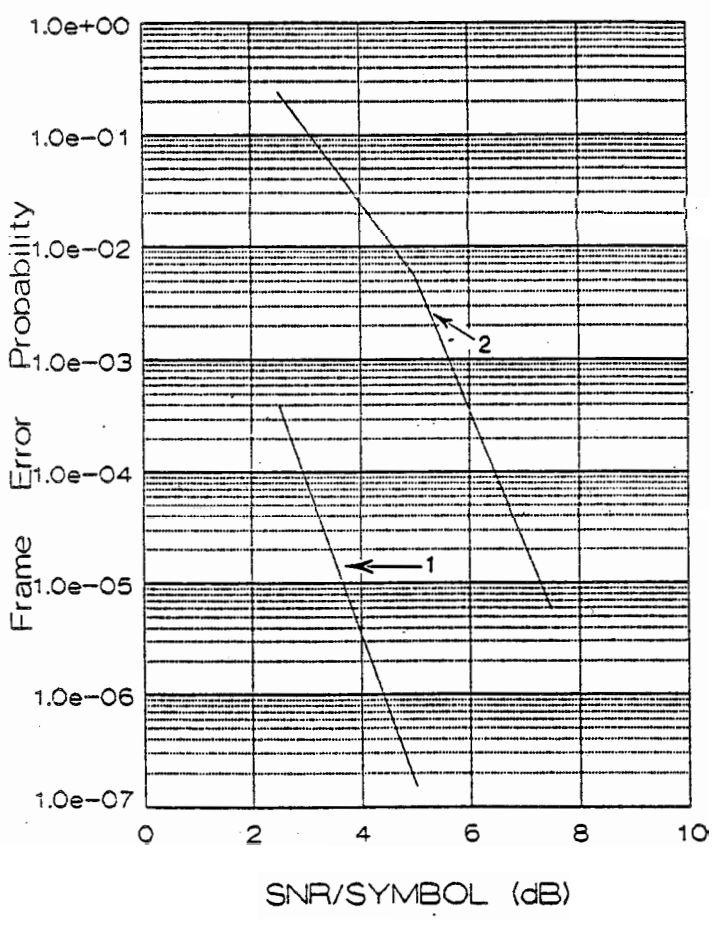



Fig. 4.14 Frame error rate of system IV (curve 2); Hagenauer's system (curve 1).

Recall that frame error rate means the probability that after the last decoding attempt, errors still exist in the data packet. It is determined by the error performance of each of the RC - TCM schemes in the system. As mentioned earlier, in Rayleigh fading channels, the error performance of a TCM system is primarily dominated by the free Hamming distance of the TCM scheme. One can observe from Table 4.2, 4.4, 4.6 and 4.8 that the free Hamming distances of the RC - TCM schemes in system IV are smaller than that of the RC - TCM schemes in any of the other three systems. Hence, system IV has the highest frame error rate.

The performances of systems I, II, III, and IV are summarized in the table:

| System | Normalized System - wide Throughput | $E_s/N_0$ Required to achieve a FER of $10^{-5}$ (dB) | $E_s/N_0$ Required to achieve a FER of $10^{-7}$ (dB) |
|---|---|---|---|
| system I | 1.51 | 5.1 | 6.1 |
| system II | 1.68 | 3.7 | 5.1 |
| system III | 1.81 | 2.9 | 3.8 |
| system IV | 1.74 | 7.3 | 8.4 |
| Hagenauer's | 1 | 3.6 | 5.1 |

Table 4.9 Performance comparison of system I, system II, system III, system IV and Hagenauer's system.

## 4.3 DECODING COMPLEXITY

In Table 4.9, we compared the throughput performances and the frame error rates of hybrid ARQ/FEC systems that use RC - TCM schemes and a Hagenauer's system

that uses RCPCC. But a fair comparison should also include the system decoding complexity. In this section, we analyze, calculate and compare the decoding complexities for all the hybrid ARQ/FEC systems we have discussed.

Following the system model described in section 4.1.1, a Viterbi decoder will be used to do the decoding, under the control of a puncturing and (or) repeating rule, for each of the decoding attempts. In our proposed hybrid ARQ/FEC systems ( as well as in Hagenauer's system ), at each decoding attempt, the decoder computes the metrics given in equation (4.2), and selects a codeword whose metric is the largest. Since the effective FEC at each decoding attempt has a different code rate, it in turn corresponds to a different decoding complexity, $X_i$. Here $X_i$ is defined as the number of metrics needed to be calculated for decoding 1 bit of information based on the $i$th FEC in the system, and

$$X_i = \frac{N_s \cdot 2^b \cdot \xi_i}{b \cdot P} \qquad i = 1, 2, \cdots, K, \tag{4.26}$$

where $K$ is the designed maximum number of transmissions for each message packet, $N_s$ is the number of encoder states, $b$ is the number of information bits to be encoded in each encoding interval, P is the puncturing and (or) repeating period, and $\xi_i$ is the number of coded modulation symbols per P intervals accumulated at the receiver for the $i$th decoding attempt. Then from Table 2.2, 4.1, 4.3, 4.5 and 4.7 we have for the different systems the following :

| System | $X_1$ | $X_2$ | $X_3$ | $X_4$ | $X_5$ | $X_6$ |
|---|---|---|---|---|---|---|
| Hagenauer's | 36 | 40 | 48 | 64 | 96 | |
| System I | 32 | 48 | 64 | 96 | 128 | 192 |
| System II | 16 | 24 | 32 | 48 | 64 | 96 |
| System III | 32 | 48 | 64 | 96 | 128 | 192 |
| System IV | 27 | 32 | 43 | 64 | 85 | 128 |

Table 4.10 Decoding complexities of each rate compatible FEC in Hagenauer's system, system I, system II, system III, and system IV.

A reasonable measure for the decoding complexity of a hybrid ARQ/FEC system, like those discussed in this thesis, is the average number of metrics ( see equation (4.2)) the decoder has to compute per information bit, $X_{av}$.

$$X_{av} = \sum_{i=1}^{K} X_i P_{C_i} \left( \prod_{j=0}^{i-1} P_{F_j} \right) + X_K \prod_{i=1}^{K} P_{F_i} \tag{4.25}$$

where $P_{F_i}$ is the probability that after the $i$th decoding attempt, errors are still present in the decoded word, i.e. a retransmission is needed. $P_{C_i} = 1 - P_{F_i}$ is the probability that the $i$th decoding attempt is successful, or no retransmissions are needed. Applying the results of $P_{F_i}$ ( equation (4.5) ) and $P_{C_i}$ ( equation (4.6) ) obtained in section 4.1.2, we calculated $X_{av}$ for 1) Hagenauer's system; 2) *system I* ( the 4 state MTCM based system ); 3) *system II* ( the 8 state Ungerboeck's code based system ); 4) *system III* ( the 16 state Ungerboeck's code based system ); 5) *system IV* ( the 8 state Divsalar and Simon's MTCM based system ). Because of the adaptability, the number of metrics needed to be calculated for successfully decoding 1 bit of information varies

according to the channel conditions as shown in Fig. 4.15.

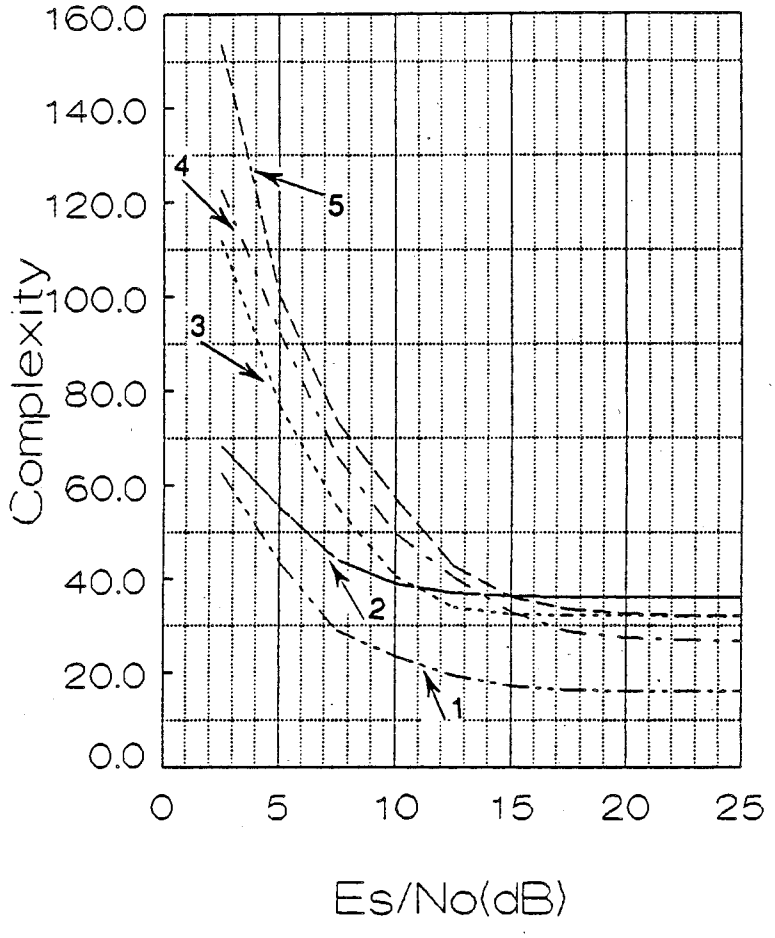



Fig. 4.15 Decoding complexities of Hagenauer's system (curve 2), system I (curve 5), system II (curve 1), system III (curve 3), system IV (curve 4).

Note that the analysis and the calculations in this section are under the conditions and the assumptions given in section 4.1 and 4.2.

Fig. 4.15 reveals that at SNR above 15 dB, all of our proposed systems have lower decoding complexities than that of Hagenauer's system. At relatively low SNR, system I, system III and system IV have higher decoding complexities compared with Hagenauer's system. Among all the systems, System II turns out to be the system with the lowest decoding complexity over the entire range of SNR.

Combining Fig. 4.15 and Table 4.9, we find that although system I and system IV have better throughput performances than Hagenauer's system, the expense for these gains are the increase in the frame error rates and the decoding complexities. System III has an 81 % throughput gain over Hagenauer's system, and a lower frame error rate, but a higher decoding complexity (at SNR less than 10 dB) than that of Hagenauer's system. System II can achieve 68 % throughput gain over Hagenauer's system without increasing the frame error rate, and its decoding complexity is much lower than that of Hagenauer's system. Therefore, without increasing the decoding complexity and the frame error rate, and without any bandwidth expansion, one can still achieve a much higher throughput efficiency by using our proposed system than employing Hagenauer's system.

# CHAPTER FIVE

# CONCLUSIONS

## 5.1 CONCLUSIONS

In this thesis, we have presented a novel technique that integrates error control protocol design and trellis coded modulation design into one single process. The concept of generating a sequence of rate compatible trellis coded modulation (RC - TCM) by puncturing and / or selectively repeating the coded symbols of a mother code periodically is introduced. Because of the code rate compatibility, only one trellis encoder and one decoder, employing the Viterbi algorithm, are needed in each of our proposed systems. Only the control rule, i.e. the puncturing or repeating pattern is changed for different codes in the sequence.

We found that a class of fairly good RC - TCM schemes can be obtained by periodically selective repeating an Ungerboeck's code, such as the 8 state and the 16 state, rate 2 bits/pulse 8PSK trellis codes.

By using RC - TCM schemes, a generalized type II hybrid ARQ/FEC system can provide unequal error protection over a wide range of SNR, and a maximum throughput efficiency of several bits per channel symbol. This is in contrast to the maximum throughput of 1 bit per channel symbol found in most conventional error control systems. For mobile radio applications, the analytical and computer simulation results show that our proposed systems achieve up to 81 % overall system throughput gain

over Hagenauer's system. At least a 68 % throughput gain can be achieved by one of our proposed systems over Hagenauer's system without any increase in the decoding complexity and the frame error rate. It should be pointed out that Hagenauer's system achieves the best throughput performance amongst all existing systems reported in the literature.

## 5.2 SUGGESTIONS FOR FURTHER RESEARCH

As mentioned earlier, we generated RC - TCM schemes based on the assumption that good code generates good codes, and by picking heuristically a puncturing/repeating rule. However, the true optimum puncturing and repeating rule of a system using RC - TCM schemes can only be obtained by exhaustive computer search. This should be carried out in future in order to obtain better RC - TCM schemes.

Within the scope of this study, only coded 8PSK modulation is used in all the RC - TCM schemes. It would be interesting to explore more bandwidth / power efficient modulation schemes such as trellis coded Quadrature Amplitude Modulations (QAM), trellis coded Continuous Phase Modulations (CPM), and higher level coded PSK modulations.

Perfect coherent detection and ideal interleaving are assumed in this thesis. To examine the cases without these assumptions makes another topic for future studies.

It would be beneficial to apply the RC - TCM technique proposed in this thesis in the design of combined source and channel coding systems for voice transmission [41] and image transmission [42].

# REFERENCES


[1] S. Lin and P. S. Yu, "A Hybrid ARQ/FEC Scheme with Parity Retransmission for Error Control of Satellite Channels", *IEEE Trans. Commun., vol.* COM - 30, pp. 1701 - 1719, July 1982.

[2] J. Wolf, A. Michelson, and A. Levesque, "On the Probability of Undetected Error for Linear Block Codes", *IEEE Trans. Commun., vol.* COM - 30, pp. 317 - 324, Feb. 1982.

[3] H. C. A. Van Duuren, "Printing Telegragh System", U.S. Patent 313 980, Mar. 1943.

[4] S. Lin and D. Costello, *Error Control Coding: Fundamentals and Applications*, Prentice Hall, Englewood Cliffs, NJ, 1983.

[5] D. Mandelbaum, "Adaptive - Feedback Coding Scheme Using Incremental Redundancy", *IEEE Trans. Infor. Theory, vol.* IT - 20, pp. 388 - 389, May 1974.

[6] A. Sastry, "performance of Hybrid Error Control Schemes on Satellite Channels", *IEEE Trans. Commun., vol.* COM - 24, pp. 385 - 393, Apr. 1976.

[7] A. Drukarev and D. Costello, Jr., "Hybrid ARQ/FEC Error Control Using Sequential Decoding", *IEEE Trans. Infor. Theory, vol.* IT - 29, pp. 521 - 535, July 1983.

[8] A. Drukarev and D. Costello, Jr., "A Comparison of Block and Convolutional Codes in ARQ Error Control Schemes", *IEEE Trans. Commun., vol.* COM - 30, pp. 2449 - 2455, Nov. 1982.

[9] C. Leung and A. Lam, "Forward Error Correction for an ARQ Scheme", *IEEE Trans. Commun., vol.* COM - 29, pp. 1514 - 1519, Oct. 1981.

[10] J. Metzner, "Improvement in Block - Retranmission Schemes", *IEEE Trans. Commun., vol.* COM - 27, pp. 525 - 532, Feb. 1979.

[11] R. Comroe and D. Costello, Jr., "ARQ Schemes for Data Transmission in Mobile Radio System", *IEEE J. Selected Areas of Commun., vol. SAC - 2*, pp. 472 - 481, July 1984.

[12] J. B. Cain, G. C. Clark and J. M. Geist, "Punctured Convolutional Codes of Rate (n-1)/n and Simplified Maximum Likelihood Decoding", *IEEE Trans. Infor. Theory, vol.* IT - 25, pp. 97 - 100, Jan. 1979.

[13] J. Hagenauer, "Rate Compatible Punctured Convolutional Codes", Proc. ICC 87, Seattle, Washington, pp. 29.1.1 - 5, June 1987.

[14] J. Hagenauer, "Rate - Compatible Punctured Convolutional Codes (RCPC Codes) and Their Applications", *IEEE Trans. Commun., vol.* COM - 36, pp. 389 - 400, Apr. 1988.

[15] H. Krishna and S. Morgera, "A New Error Control Scheme for Hybrid ARQ Systems", *IEEE Trans. Commun., vol.* COM - 35, pp. 981 - 990, Oct. 1987.

[16] S. D. Morgera and V. K. Oduol, "Soft - Decision Decoding Applied to the Generalized Type-II Hybrid ARQ Scheme", *IEEE Trans. Commun., vol.* COM - 37, pp. 393 - 396, Apr. 1989.

[17] A. J. Viterbi and J. K. Omura, *Principles of Digital Communication and Coding*, New York: McGraw Hill, 1979.

[18] J. L. Massey, "Coding and Modulation in Digital Communications", *Proc. 1974 Int. Zurich Seminar on Digital Commun.*, Zurich, Switzerland, pp. E2(1) - (4), Mar. 1974.

[19] J. Hagenauer, "Viterbi Decoding of Convolutional Codes for Fading and Burst Channels", *Proc. 1980 Int. Zurich Sem.*, pp. G2.1 - G2.7 Mar. 1980.

[20] P. F. Turney, "An Improved Stop - and - Wait ARQ Logic for Data Transmission in mobile radio systems", *IEEE Trans. Commun. vol.* COM - 29, pp. 68 - 71, Jan. 1981.

[21] D. Chase, "Code Combining -- A Maximum - Likelihood Decoding Approach for Combining an Arbitrary Number of Noisy Packets", *IEEE Trans. Commun., vol.* COM - 33, PP. 385 - 393, May 1985.

[22] J. Cavers and P. Ho, "Analysis of Error Performance of Trellis - Coded Modulations in Rayleigh Fading Channels", To appear on *IEEE Trans. Commun.*

[23] G. Ungerboeck, "Channel Coding with Multilevel / Phase signals", *IEEE Trans. Infor. Theory, vol. IT - 28*, pp. 55 - 67, Jan. 1982.

[24] G. Ungerboeck, "Trellis - Coded Modulation with Redundant Signal Sets; Part I: Introduction, Part II: State of Art", *IEEE Commun. Mag., vol.* COM - 25, pp. 5 - 21, Feb. 1987.

[25] D. Divsalar and M. K. Simon, "The Design of Trellis Coded MPSK for Fading Channels: Performance Criteria", *IEEE Trans. Commun., vol.* COM - 36, pp. 1004 - 1012, Sept. 1988.

[26] D. Divsalar and M. K. Simon, "The Design of Trellis Coded MPSK for Fading Channels: Set Partitioning for Optimum Code Design", *IEEE Trans. Commun., vol.* COM - 36, pp. 1013 - 1021, Sept. 1988.

[27] D. Divsalar and M. K. Simon, "Trellis Coded Modulation for 4800 to 9600 bps Transmission Over a Fading satellite Channel", *IEEE J. Selet. Areas Commun., vol.* SAC - 5, pp. 162 - 175, Feb. 1987.

[28] P. J. Mclane, P. H. Wittke, P. Ho, and C. Loo, "PSK and DPSK Trellis Codes for Fast Fading, Shadowed Mobile Satellite Communication Channels", *IEEE Trans. Commun., vol.* COM - 36, pp. 1242 - 1246, Nov. 1988

[29] D. Divsalar, M. K. Simon and T. Jedrey, "Trellis Coding Techniques for Mobile Communications", *Proc. IEEE Military Commun. Conf.*, pp. 35.3.1 - 7, 1988.

[30] Y. Wu and P. Ho, "Multiple Trellis Codes for Mobile Fading Channels", *Proc. Canadian conf. on Electrical and Computer Engineering*, pp. 417 - 420, Vancouver, Nov. 1987.

[31] Y. Wu and P. Ho, "The Performance of Trellis - Coded DPSK for Rayleigh Fading Channels", *Proc. IEEE Pacific Rim Conf. on Commun.*, pp. 400 - 403, Victoria, June 1989.

[32] E. Zehavi and J. K. Wolf, "On the Performance Evaluation of Trellis Codes", *IEEE Trans. Infor. Theory, vol.* IT - 33, pp. 196 - 202, Mar. 1987.

[33] J. G. Proakis, *Digital Communications*, McGraw Hill, 1983.

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

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

[36] G. Benelli, "An ARQ Scheme with Memory and Integated Modulation", *IEEE Trans. Commun., vol.* COM - 35, pp. 689 - 697, July 1987.

[37] T. Aulin, N. Rydbeck, and C. E. Sundberg, "Continuous Phase Modulation; Part I: Full Response Signaling, Part II: Partial Response Signaling", *IEEE Trans. Commun., vol.* COM - 29, pp. 196 - 255, Mar. 1981.

[38] R. J. Benice and A. H. Frey, Jr, "An Analysis of Retransmission Systems", *IEEE Trans. Commun. Technol., vol.* COM - 12, pp. 135 - 145, Dec. 1964.

[39] M. Moeneclaey, H. Bruneel, I. Bruyland, and D. - Y. Chung, "Throughput Optimization for a Generalized Stop - And - Wait ARQ Scheme", *IEEE Trans. Commun., vol.* COM - 34, pp. 205 - 207, Feb. 1986.

[40] A. R. K. Sastry and L. N. Kanal, "Hybrid Error Control Using Restransmission and Generalized Burst - Trapping Codes", *IEEE Trans. Commun., vol.* COM - 24, Apr. 1976.

[41] D. J. Goodman and C. E. Sundberg, "Combined Source and Channel Coding for Variable - Bit - Rate Speech Transmission", *Bell syst. Tech. J., vol. 62*, No. 7, PP. 2017 - 2036, Sept. 1983.

[42] D. G. Daut and J. W. Modestino, "Two Dimensional DPCM Image Transmission Over Fading Channels", *IEEE Trans. Commun., vol. COM - 31*, PP. 315 - 328, Mar. 1983.

[43] J. K. Cavers, "Phase Locked Transparent Tone - in - Band - An Analysis", *Proc. Canadian conf. on Electrical and Computer Engineering*, pp. 267 - 270, Vancouver, Nov. 1987.