

MULTIUSER DETECTION IN CDMA SYSTEM  
AND ITS SIMULATION IN MATLAB

A THESIS SUBMITTED TO  
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES  
OF  
ÇANKAYA UNIVERSITY

BY

BURHAN ÇAKIR

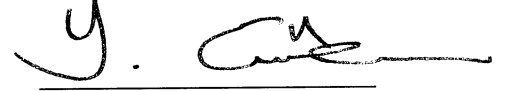
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR  
THE DEGREE OF MASTER OF SCIENCE  
IN  
ELECTRONIC AND COMMUNICATION

MAY 2007

Title of the Thesis: **Multiuser Detection In CDMA System And Its Simulation In Matlab.**

Submitted by **Burhan Çakır**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University



Prof. Dr. Yurdahan Güler  
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.



Prof. Dr. Yahya Kemal Baykal  
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

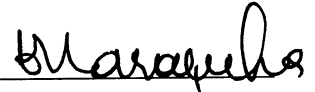


Asst. Prof. Dr. Halil Tanyer Eyyuboğlu  
Supervisor

**Examination Date : 31.05.2007**

**Examining Committee Members:**

Assoc. Prof. Dr. Ertuğrul Karaçuha (Telekomünikasyon. Kur.)




Asst. Prof. Dr. Halil Tanyer Eyyuboğlu (Çankaya University)



Prof. Dr. Yahya Kemal Baykal

(Çankaya University)



### Statement of Nonplagiarism Page

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Burhan akır

Signature : 

Date : 31.05.2007

## ABSTRACT

### MULTI-USER DETECTION IN CDMA SYSTEM AND ITS SIMULATION IN MATLAB

ÇAKIR, Burhan

M.S.c., Department of Electronics and Communication Engineering

Supervisor: Asst. Prof. Dr. Halil Tanyer EYYUBOĞLU

May 2007, 44 pages

In this thesis, we investigated performance of uncoded and with convolutional coded direct sequence code division multiple access (DS-SS) with decorrelating detection communication systems in additive white Gauss noise (AWGN) channel and Rician channel with Rician factors of 1, 10, 100.

Rician channel with a high Rician factor will gradually resemble an AWGN channel. But as the Rician factor is made smaller, the error performance degrades, finally approaching the Rayleigh limit.

Comparisons of uncoded and convolutional coded system show that: convolutional coded system has much better performance than the uncoded system. Two different convolutional code rates, which are  $1/2$ , and  $2/3$  are tested. It is observed that for the same bit error rate (BER) value, at the code rate  $1/2$  has a lower signal-noise ratio (SNR) value than the code rate  $2/3$ .

**Keywords:** Direct Sequences Code Division Multiple Access Communication System, Multiuser Detection, Decorrelating detection, Multiple Access Interference, AWGN Channel, Rician Channel, Gold Sequence, Convolutional Coding.

## ÖZ

### KOD PAYLAŞIMLI ÇOKLU ERİŞİM SİSTEMİNDE ÇOKLU KULLANICI ALGILAMASI ve MATLAB BENZETİMİ

ÇAKIR, Burhan

Yüksek Lisans, Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Tez Yöneticisi: Yrd. Doç. Dr. Halil Tanyer EYYUBOĞLU

Mayıs 2007, 44 sayfa

Bu tezde, kanal kodsuz ve evrişim kodlu ilintisiz alıcılı doğrudan dizilerle kod paylaşımli çoklu erişim (DS-CDMA) haberleşme sisteminin, toplanır beyaz Gauss gürültü (AWGN) kanalında ve Rician faktörü 1, 10 ve 100 olan Rician kanalında performansı incelenmiştir.

Yüksek Rician faktöründe, Rician kanalı AWGN kanalına benzemektedir. Düşük Rician faktöründe ise Rician kanalı Rayleigh kanalına yaklaşarak sistemin performansı azalmaktadır.

Kanal kodsuz ve evrişim kodlu sistemin mukayesesinde, evrişim kodlu sistem kodsuz sisteme göre oldukça iyi performans göstermektedir. 1/2 ve 2/3 evrişim kod oranlarının mukayesesinde ise, aynı bit hata oranı (BER), 2/3 kod oranına göre 1/2 kod oranında daha düşük işaret-gürültü oranı (SNR) değerinde elde edilmektedir.

**Anahtar Kelimeler:** Doğrudan Dizilerle Kod Paylaşımli Çoklu Erişim Haberleşme Sistemi, Çoklu Kullanıcı Algılaması, İlintisiz Alıcı, Çoklu Erişim Girişimi, Toplanır Beyaz Gauss Gürültülü Kanal, Rician Kanal, Evrişim Kodlaması.

## ACKNOWLEDGEMENTS

I would like to express my sincere appreciation to my advisor Asst. Prof. Dr. Halil Tanyer EYYUBOĞLU for his great input to this thesis, which would not have been completed without his patient guidance, continuous inspiration and in-depth expertise in the field of CDMA.

I would like to thank to Prof. Dr. Yahya Kemal BAYKAL and Assoc. Prof. Dr. Yusuf Ziya UMUL for their very instructive, useful discussions during the courses.

I also express my sincere appreciation to Assoc. Prof. Dr. Mustafa ALKAN and Assoc. Prof. Dr. Ertuğrul KARAÇUHA for their invaluable guidance and constant encouragement.

Finally, I would like to express my gratitude to my family. No fulfillments would have been possible without the ever-lasting love and support from my family.

## TABLE OF CONTENTS

STATEMENT OF NON PLAGIARISM.....	iii
ABSTRACT.....	iv
ÖZ.....	v
ACKNOWLEDGEMENTS.....	vi
TABLE OF CONTENTS.....	vii
LIST OF TABLES.....	ix
LIST OF FIGURES.....	x
LIST OF SYMBOLS.....	xi
LIST OF ABBREVIATIONS.....	xiii
CHAPTERS:	
1. INTRODUCTION.....	1
1.1 Background of the Study.....	1
1.2 Objectives of the Study.....	3
1.3 Organization of the Thesis.....	4
2. DS-CDMA COMMUNICATION SYSTEM.....	5
2.1 Transmitter.....	6
2.1.1 Channel Coding.....	6
2.1.2 Spreading.....	9
2.1.3 PN Sequence.....	11
2.1.4 Carrier Modulation.....	13
2.2 Channel.....	14
2.2.1 Rayleigh and Rician Fading Channel.....	16
2.3 Receiver.....	21
2.3.1 Carrier Demodulation.....	21
2.3.2 Despreading.....	22
2.3.3 Channel Decoding.....	23
2.4 CDMA Receivers.....	25

2.4.1 Conventional Single User Detector .....	26
2.4.2 Optimum Receiver .....	27
2.4.3 Suboptimum Linear Receiver.....	27
2.4.3.1 Minimum Mean-Square Error (MMSE) Receiver.....	27
2.4.3.2 Decorrelating Detector .....	27
2.5 Power Control in CDMA.....	28
3. SIMULATIONS RESULTS.....	31
3.1 Simulation Parameters.....	31
3.2 Performance Analysis of Uncoded System.....	33
3.3 Performance Analysis of 2/3 Convolutional Coded System .....	37
3.4 Performance Analysis of 1/2 Convolutional Coded System.....	40
4. CONCLUSIONS.....	43
REFERENCES.....	R1
APPENDICES	
APPENDIX A: Uncoded CDMA System with Decorrelator Detector in AWGN and Rician Channel Simulation Software.....	A1
APPENDIX B: 2/3 and 1/2 Convolutional Coded CDMA System with Decorrelator Detector in AWGN and Rician Channel Software.....	A4



## LIST OF TABLES

### TABLES

Table 3.1 Some selected SNR-BER values from Figure 3.1 .....	33
Table 3.2 Some selected SNR-BER values from Figure 3.2.....	34
Table 3.3 Some selected SNR-BER values from Figure 3.3.....	35
Table 3.4 Some selected SNR-BER values from Figure 3.4.....	36
Table 3.5 Some selected SNR-BER values from Figure 3.5.....	37
Table 3.6 Some selected SNR-BER values from Figure 3.6.....	38
Table 3.7 Some selected SNR-BER values from Figure 3.10.....	41

## LIST OF FIGURES

### FIGURES

Figure 2.1 A DS-CDMA communication system model.....	5
Figure 2.2 1/2 convolutional encoder with generator polynomial (111,101).....	6
Figure 2.3 State transition diagram of (2, 1, 3) convolutional encoder.....	7
Figure 2.4 Trellis representation for rate 1/2 convolutional encoder.....	8
Figure 2.5 Spreading.....	11
Figure 2.6 Generation of Gold sequence.....	13
Figure 2.7 Multipath propagation.....	15
Figure 2.8 Rayleigh probability density functions for various $\sigma$ .....	17
Figure 2.9 Rician probability density functions for various $\nu$ with $\sigma = 1$ .....	18
Figure 2.10 Rician probability density functions for various $\nu$ with $\sigma=0,25$ .....	19
Figure 2.11 Comparison of Rayleigh and Rician at $\nu / \sigma=0,5$ .....	20
Figure 2.12 Comparison of Rician, Gaussian and Rayleigh at $\nu / \sigma=1,5$ .....	20
Figure 2.13 Comparison of Gaussian and Rician at $\nu/\sigma=3,5$ .....	21
Figure 2.14 Despreading.....	22
Figure 2.15 Trellis diagram for Viterbi decoding.....	25
Figure 2.16 Decorrelating Detector.....	28
Figure 3.1 BER performance graph of uncoded system, $T=31, K=5$ .....	33
Figure 3.2 BER performance graph of uncoded system, $T=31, K=25$ .....	34
Figure 3.3 BER performance graph of uncoded system, $T=127, K=5$ .....	35
Figure 3.4 BER performance graph of uncoded system, $T=127, K=25$ .....	36
Figure 3.5 BER performance graph of 2/3 convolutional coded system, $T=31, K=5$ ....	37
Figure 3.6 BER performance graph of 2/3 convolutional coded system, $T=31, K=25$ ... 38	38
Figure 3.7 BER performance graph of 2/3 convolutional coded system, $T=127, K=5$ ... 39	39
Figure 3.8 BER performance graph of 2/3 convolutional coded system, $T=127, K=25$ .. 39	39
Figure 3.9 BER performance graph of 1/2 convolutional coded system, $T=31, K=5$ .... 40	40
Figure 3.10 BER performance graph of 1/2 convolutional coded system, $T=31, K=25$ .. 40	40
Figure 3.11 BER performance graph of 1/2 convolutional coded system, $T=127, K=5$ .. 42	42
Figure 3.12 BER performance graph of 1/2 convolutional coded system, $T=127, K=25$ . 42	42

## LIST OF SYMBOLS

$b_k$	Estimated signal for $k$ th user
$c(t)$	PN code sequence
$CR$	Code rate of convolutional coding
$E_b$	Information bit energy
$f_c$	Transmission carrier frequency
$u$	Generator polynomials of convolutional coding
$G_n(f)$	Noise power spectral density
$h_i$	Fading coefficient of the $i$ th path
$I_0(\cdot)$	Modified zero-order Bessel function of the first kind.
$k$	Number of input bits in convolutional coding
$K$	Number of user
$KK$	Rician factor
$L$	Constraint length of convolutional coding
$n$	Number of output bits in convolutional coding
$n(t)$	White Gaussian noise
$N_0$	Noise power
$N$	Number of information bits
$PG$	Processing Gain
$P$	Total received power
$P_o$	Specular received power
$P_f$	Diffuse fading received power
$P_e$	Bit error probability
$R_c(m)$	Crosscorrelation function
$R_c$	Chip rate
$R_b$	Information rate
$\sigma^2$	Variance
$r$	Amplitude of received signal $r(t)$

## CHAPTER 1

### INTRODUCTION

#### 1.1 Background of the Study

In a direct sequence spread spectrum code division multiple-access (DSSS-CDMA) wireless communication system, a channel with a given bandwidth is accessed by all the users simultaneously. The different mobile users are distinguished at the base station receiver by the unique spreading code assigned to the users to modulate the transmitted signals. A CDMA cellular communication system is inherently interference-limited. This is due to the difficulty of maintaining orthogonality on the reverse link between channels used by independent mobile stations, which transmit asynchronously. This form of interference limits the uplink capacity severely. In addition to multiple access interference (MAI), CDMA systems also suffer from multipath fading.

Different users in CDMA employ signals that have very small cross-correlation. Thus, correlators can extract individual signals from a mixture of signals even though they are transmitted simultaneously in the same frequency band. CDMA systems employ wideband signals with good cross-correlation properties, which mean the output of a filter matched to one user's signal is small when it receives a different user's signal as input. In direct-sequence spread-spectrum systems, a high-rate antipodal pseudorandom spreading sequence modulates the transmitted signal so that the bandwidth of the resulting signal is roughly equal to the rate of the spreading sequence. The cross-correlation of the signals is then largely determined by the cross-correlation properties of the spreading signals. Although CDMA signals overlap in both time and frequency domains, they can be separated, based on their spreading waveforms.

The purpose of the channel coding techniques is to detect and possibly correct errors that occur when messages are transmitted in a digital communication system. To accomplish this, the encoder transmits message signals in encoded format. The decoder uses this encoding format to detect and possibly correct whatever errors occurred during transmission [1],[2].

Convolution coding is employed to improve the error correcting capability and power efficiency of the system. CDMA systems exhibit their full potential, when combined with forward error correction (FEC) coding [3]. Collaborated with FEC coding, MUD can overcome its limitations in highly correlated multiuser systems. Therefore, in some practical systems, MUD is employed in conjunction with FEC coding to obtain greater capacity and throughput [4]. In this thesis, we used convolutional code at  $1/2$  and  $2/3$  code rates.

Multiuser detection techniques are widely used to combat the detrimental effects of multipath fading and multiple access interference (MAI), which are the major impairments in CDMA communication systems. Generally, there are three different methods in the detection of user's signal: the single user detection technique, the optimal detection technique and the suboptimal detection technique [5].

The conventional detector has a complexity that grows linearly with the number of users ( $K$ ), but its vulnerability to the near-far problem requires some type of power control. This detection method is optimum in the single-user case, or if all user PN code sequences are mutually orthogonal. The matched filter receiver works reasonably well in a multiuser environment if there are few users with low correlation sequences and the received powers from different users are nearly equal. However, in mobile wireless environments, the receiver suffers from deep fading and the near-far effect, where interfering users' powers are much greater than the desired user, which renders the conventional detector useless. In addition, orthogonality between the PN sequences can be destroyed by multipath signal propagation. Thus, using a conventional receiver, MAI is severe and the performance is very poor for uncoded wireless systems.

In the Decorrelating Detector, the crosscorrelation matrix is assumed to be known. The match filter outputs are multiplied by the inverse of the crosscorrelation matrix. This linear transformation eliminates the MAI and is said to be near far resistant. The computational complexity of the decorrelating detector is linear in  $K$ . The decorrelating receiver has the advantage that knowledge of the received amplitudes is not required. However, the inversion of the channel performed by the decorrelator enhances the background noise. In this thesis, decorrelating detector based on suboptimal linear multiuser detection technique is used.

Rayleigh and Rician fading channels are useful models of real-world phenomena in wireless communications. These phenomena include multipath scattering effects, time dispersion, and Doppler shifts that arise from relative motion between the transmitter and receiver. Typically, the fading process is characterized by a Rayleigh distribution for a nonline-of-sight path (NLOS) and a Rician distribution for a line-of-sight (LOS) path.

The performance of a detection technique, which is usually expressed in terms of bit error rate (BER), may be evaluated in three ways. The first one entails a mathematical formulation of the process, while the second course is based on computer simulation. The final method is the physical act of measuring BER out in the field. Here, it is the former two cases that are of concern to us. A mathematical formulation may be considered as an excellent choice in the sense that it will provide exact result almost instantly, and if no restrictions exist, it will be applicable throughout the entire range of signal to noise ratio (SNR) values [6].

## **1.2 Objectives of the Study**

Performance of uncoded and convolutional coded (code rates 1/2 and 2/3) DS-SS-CDMA communication system with decorrelating detection simulations were executed in two types of channels, the additive white Gaussian noise (AWGN) and the Rician channel with Rician factors ( $KK$ ) of 1,10, 100.

Simulations verify that when  $KK = 0$ , the channel is Rayleigh, and if  $KK$  is infinite, the channel is Gaussian. The fades have a high probability of being very deep when  $KK = 0$  to being very shallow when  $KK = 100$  (approaching Gaussian).

Comparison of uncoded and convolutional coded system simulations show that: convolutional coded system has much better performance than the uncoded system. Two different convolutional code rates, which are  $1/2$ , and  $2/3$  are tested. It is observed that for the same BER value, at the code rate  $1/2$  has a lower SNR value than the code rate  $2/3$ .

### **1.3 Organization of the Thesis**

This thesis comprises four chapters.

First chapter covers introduction, objectives of the study and organization of the thesis.

In chapter 2, background information on direct sequence spread spectrum code division multiple access (DS-CDMA) wireless communication systems, convolutional coding, a brief introduction of multiuser detection techniques, which are used in CDMA and Rayleigh and Rician channel are provided.

Chapter 3 discusses the calculation the probability of error via computer simulation of uncoded and convolutional coded DS-CDMA systems in the two types of channels, which are additive white Gaussian noise channel (AWGN), and Rician channel.

Chapter 4 assesses the simulations results.

## CHAPTER 2

### DS-CDMA COMMUNICATION SYSTEM

Information on DS-CDMA communication system is provided in this chapter. For more detailed and broader overview of CDMA spread spectrum communication systems, the readers are recommended to consult [8]-[28].

#### 2. DS-CDMA communication system

Code division multiple access (CDMA) is a modulation and multiple-access scheme based on spread-spectrum communication. In a CDMA system, each signal consists of a different pseudo random binary sequence, which is called the spreading code. In this scheme, multiple users share the same frequency band at the same time, by spreading the spectrum of their transmitted signals, so that each user's signal is pseudo-orthogonal to the signals of the other users. If CDMA is viewed in either the frequency or time domain, the multiple access signals overlap with each other. However, the use of statistically orthogonal spreading codes separates the various signals in the code space.

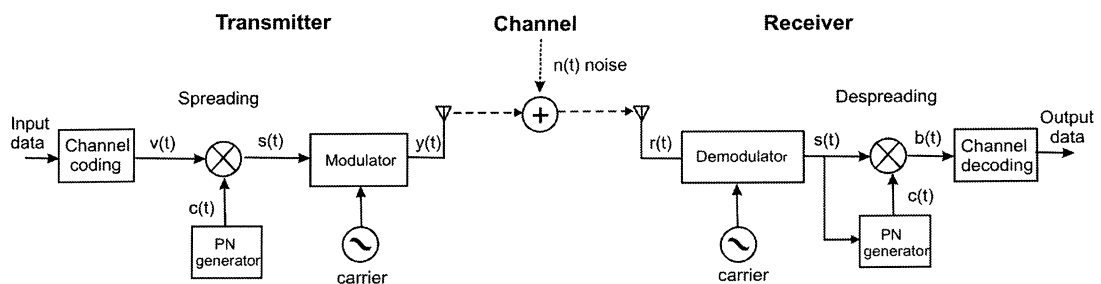


Figure 2.1 A DS-CDMA communication system model



The CDMA spread spectrum communication system depicted in Figure 2.1 consists of three major components: transmitter, channel and receiver.

## 2.1 Transmitter

Transmitter converts the electrical signal into a form that is suitable for transmission through the physical channel. The functional blocks in the transmitter include channel coding, spreading and carrier modulation [7].

### 2.1.1 Channel Coding

The purpose of the channel coding techniques is to detect and possibly correct errors that occur when messages are transmitted in a digital communication system.

In this thesis, we consider the use of convolutional codes with rates  $1/2$  and  $2/3$ . Convolutional codes are commonly specified by 3 parameters: the number of output bits  $n$ , the number of input bits  $k$  and the constraint length  $L$ . The constraint length of the code is one plus the number of past inputs affecting the current outputs.

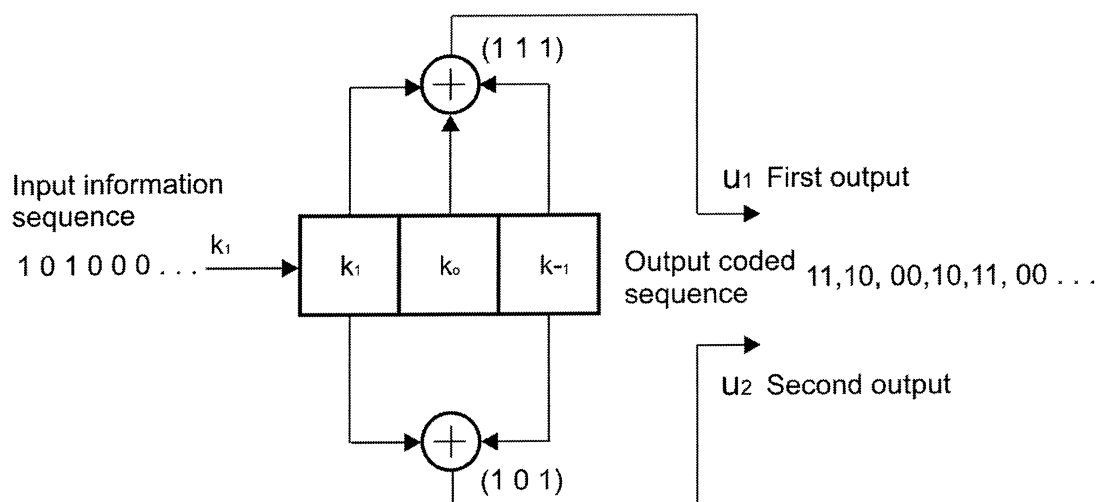


Figure 2.2 1/2 convolutional encoder with generator polynomial  $(111, 101)$

A convolutional encoder with parameters  $(n, k, L) = (2, 1, 3)$ , and code rate  $CR=k/n=1/2$ , is illustrated in Figure 2.2. It consists of a  $L = 3$  stage shift register that holds the information bits. The shift register stages are connected to modulo-2 adders. The connections are determined by the generator polynomial, which is (111,101) in this case. 111 and 101 binary numbers are equivalent to the octal numbers 6 and 7, respectively, so the generator polynomial matrix is [6 7].

The encoder operates on the incoming message sequence, one bit at a time. For each input bit  $k_l$ , the encoder outputs two coded bits  $u_1 = k_l \oplus k_0 \oplus k_{-1}$ ,  $u_2 = k_l \oplus k_{-1}$ . It is obvious that the outputs not only depend on the incoming information bit  $k_l$ , but also on the previous two information bits stored in the shift register. (The bits in the  $k_0$ , and  $k_{-1}$  stages of the shift register).

The encoder may be regarded as a finite state machine. The final two stages of the shift register hold past inputs and function as the memory of the machine. In this example, there are  $L-1 = 2$  memory stages and hence  $2^{L-1}$  possible states.

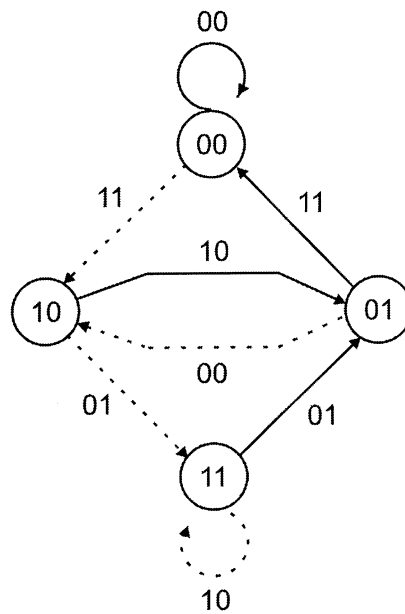


Figure 2.3 State transition diagram of (2, 1, 3) convolutional encoder

The convolutional code in Figure 2.2 can also be represented by a state transition diagram as shown in Figure 2.3. The state of the encoder is represented by the contents of the circle. The encoder begins in state 00. If the first encoder input bit is a 0, the encoder exits state 00 on the solid branch, outputs the two code symbols found on this branch 00, and returns to state 00. If the first input bit were a 1, the encoder would exit state 00 on the dashed branch, would output the two code symbols found on this branch 11, and would enter state 10. In general, encoder inputs equal to 0 cause the encoder to move along the solid branches to the next state and to output the codeword symbols corresponding to the branch label. Encoder inputs equal to 1 cause the encoder to move along the dashed branches to the next state again outputting two codeword symbols encountered along the branch. Convolutional coders can therefore be thought of as finite machines that change as a function of the input sequence. The branch labels in Figure 2.3 are calculated directly from the shift register presentation [1].

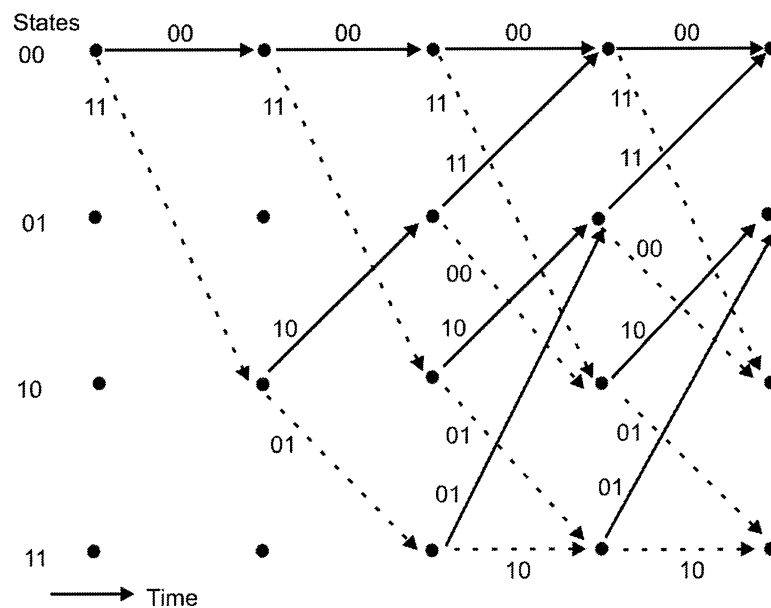


Figure 2.4 Trellis representations for rate 1/2 convolutional encoder

A convolutional encoder can also be represented by a trellis diagram as illustrated in Figure 2.4 for the code of Figure 2.2. The encoder represented by Figure 2.4 has four states, which are represented by the labels next to the points of the trellis and correspond also to the states transition diagram. The encoding operation start at state 00 on the far left of the trellis. If the first information bit is a 0, the encoder moves along the solid line out of state 00 arriving again at state 00. The encoder output is the symbol pair 00, which is the label on the trellis branch between the two states. If the first encoder input were a 1, the encoder would move along the dashed branch out of state 00 arriving at state 10. In this case, the encoder output is 11, which is the label on the branch connecting states 00 and 10. The second encoder input causes the encoder to move to the right one more branch and to output the associated branch label. This process of moving from left to right through the trellis and outputting the branch labels continues as long as desired. Input information bits equal to 0 cause the encoder to move along the solid branches to the next state while information 1s cause the encoder to move along the dashed branches. The code sequence generated is the sequence of branch labels encountered as the coder follows the path from left to right through the trellis.

The encoding procedure in a convolutional code is very simple. We assume that the encoder, before the first information bits enters it, is loaded with zeros (the all-zero state). The information bits enter the encoder  $k$  bits at a time and the corresponding  $n$  output bits are transmitted over the channel. This procedure is continued until the last group of  $k$  bits is loaded into the encoder and the corresponding  $n$  output bits are sent over the channel. We assume, for simplicity, that after the last set of  $k$  bits, another set of  $k(L-1)$  zeros enter the encoder and the corresponding  $n$  outputs are transmitted over the channel. This returns the encoder to the all-zero state and makes it ready for next transmission.

### 2.1.2 Spreading

Spreading and Despreading are the main features that distinguish the spread spectrum systems from general digital communication systems. The traditional

approach to digital communications is based on the idea of transmitting as much information as possible in as narrow a signal bandwidth as possible.

Spread spectrum is a technology used to combat the MAI and intentional jamming by spreading the transmitted signal over a wide frequency band, much wider than the minimum bandwidth required by the message signal.

In spread spectrum, the data is modulated by a spreading signal which uses more bandwidth than the data signal. Since multiplication in the time domain corresponds to convolution in the frequency domain, a narrow band signal multiplied by a wideband signal ends up being wideband. One way of doing this is to use a binary waveform as a spreading function, at a higher rate than the data signal.

Bits of the spreading signal are called *chips*.  $T_b$  represents the period of one data bit and  $T_c$  represents the period of one chip. The chip rate,  $1/T_c$ , is often used to characterize a spread spectrum transmission system. The *Processing Gain* or sometimes called the Spreading Factor is defined as the ratio of the information bit duration over the chip duration:

$$PG = SF = T_b / T_c \quad (2.1)$$

Hence, it represents the number of chips contained in one data bit. Higher Processing Gain ( $PG$ ) means more spreading. High  $PG$  also means that more codes can be allocated on the same frequency channel.

Figure 2.5 shows the basic elements of a direct sequence spread spectrum system (DSSS). The PN code generator output  $c(t)$  in the transmitter is a chip sequence of rate  $R_c=1/T_c$  which is typically much higher than the data rate  $R_b = 1/T_b$ . In short code CDMA systems,  $c(t)$  is periodic with the period equal to the symbol duration  $T_b$ . While in long-code CDMA systems,  $c(t)$  is different from symbol to symbol, its period is much longer than the symbol duration.

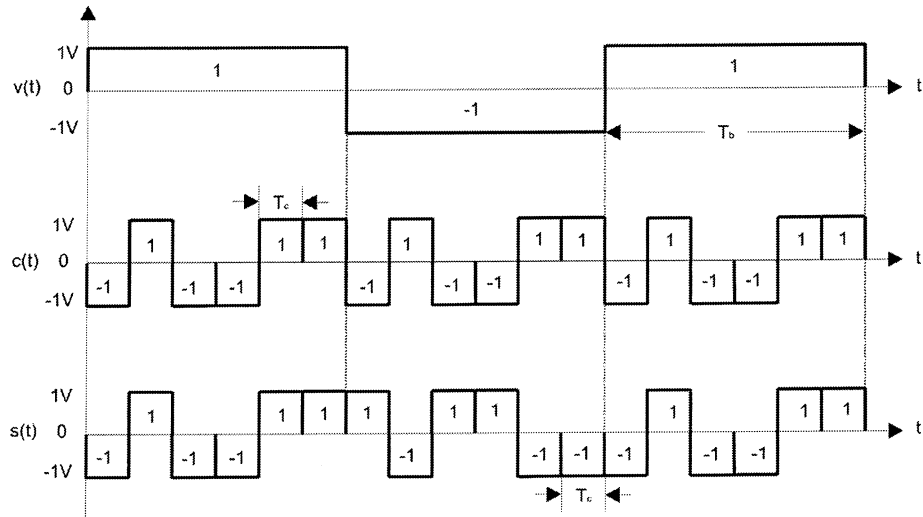


Figure 2.5 Spreading

In the transmitter binary data signal  $v(t)$  is directly multiplied with the PN spreading sequence  $c(t)$ , to produce the transmitted baseband signal  $s(t)$ . The product signal

$$s(t) = v(t) c(t) \quad (2.2)$$

is the signal after spreading. The effect of multiplication of  $v(t)$  with a PN sequence,  $c(t)$ , is to spread the baseband bandwidth  $W_b$  of  $v(t)$  to a baseband bandwidth of  $W_c$ .

### 2.1.3 PN Sequence

A pseudo-random or pseudo-noise (PN) sequence is a code sequence of 1's and 0's whose autocorrelation has the properties similar to those of white noise. By far the most widely known binary PN code sequences are the maximum-length shift register sequences. A maximum length shift register sequence, or  $m$ -sequence for short, has a length  $T = 2^m - 1$  bits and is generated by an  $m$ -state shift register with linear feedback. The sequence is periodic with period  $T$ . Each period has a sequence of  $2^{m-1} - 1$  ones and  $2^{m-1} - 1$  zeros [11],[12].

In DS spread spectrum applications, the binary sequence with elements  $\{0, 1\}$  is mapped into a corresponding binary sequence with elements  $\{-1, 1\}$ . We shall call the equivalent sequence  $\{c_n\}$  with elements  $\{-1, 1\}$  a bipolar sequence.

An important characteristic of a periodic PN sequence is its autocorrelation function, which is usually defined in terms of the bipolar sequences  $\{c_n\}$  as

$$R_c(m) = \sum_{n=1}^T c_n c_{n+m}, \quad 0 \leq m \leq T \quad (2.3)$$

where  $T$  is the period of the sequence. Since sequence is periodic with period  $T$ , the autocorrelation sequence  $\{R_c(m)\}$  is also periodic with period  $T$ .

Ideally, a PN sequence should have an autocorrelation function that has correlation properties similar to white noise. That is, the ideal autocorrelation sequence for  $\{c_n\}$  is  $R_c(0) = T$  and  $R_c(m) = 0$  for  $1 \leq m \leq T-1$ . In the case of  $m$ -sequence, the autocorrelation sequence is

$$R_c(m) = \begin{cases} T, & m = 0 \\ -1 & 1 \leq m \leq T-1 \end{cases} \quad (2.4)$$

For long  $m$ -sequence, the size of the off-peak values of  $R_c(m)$  relative to the peak value  $R_c(0)$ ; i.e., the ratio  $R_c(m) / R_c(0) = -1 / T$ , is small and, from a practical viewpoint, too small. Therefore,  $m$ -sequence is very close to the ideal PN sequence when viewed in terms of their autocorrelation function.

In some application, the crosscorrelation properties of PN sequences are as important as the autocorrelation properties. For example, In CDMA system each user is assigned a particular PN sequence. Ideally, the PN sequence among user should be mutually uncorrelated so that the level of interference experienced one by user from transmissions of other user adds on a power basis. However, the PN sequence used in practice by different user exhibit some correlation.

It is observed that the number of  $m$ -sequence of length  $T$  increases rapidly with  $m$ . It is also observed that, for most sequence, the peak magnitude  $R_{max}$  of the crosscorrelation function is a large percentage of the peak value of the autocorrelation function. Consequently,  $m$ -sequence are not suitable for CDMA communications system. Although it is possible to select a small subset of  $m$ -sequence that have relatively smaller crosscorrelation peak value than  $R_{max}$ , the number of sequences in the set is usually too small for CDMA application.

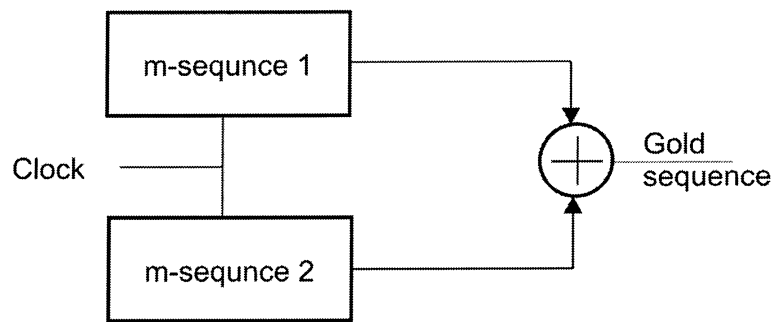


Figure 2.6 Generation of Gold sequence

Gold sequences have better periodic crosscorrelation properties than  $m$ -sequence. Gold sequences are constructed by taking a pair of specially selected  $m$ -sequence, called preferred  $m$ -sequence, and forming the modulo-2 sum of the two sequence, for each of  $T$  cyclically shifted versions of one sequence relative to the other sequence. Thus,  $T$  Gold sequence are generated as illustrated in Figure 2.6. For  $m$  odd, the maximum value of the crosscorrelation function between any pair of Gold sequences is  $R_{max} = \sqrt{2T}$ . For  $m$  even,  $R_{max} = \sqrt{T}$  [11],[12].

#### 2.1.4 Carrier Modulation

Carrier modulation is the process of shifting the frequency components of baseband pulse shapes to a suitable frequency band in order to efficiently pass the signals



through the channel. This process usually involves modulating the amplitude, frequency, and/or phase of a carrier.

The nature of the radio transmission necessitates the use of carrier modulation. For example, a 2400 symbols/sec pulse train is an audio-bandwidth signal and is ideal for transmission over a telephone line. But sending the same signal by radio would require an antenna hundreds of kilometers long. On the other hand, if the baseband pulses are first modulated on a higher frequency carrier, e.g., a 900 MHz carrier, the antenna length would be about a few centimeters [7]. Using only baseband transmission, frequency division multiplexing (FDM) could not be applied. Analog modulation shifts the baseband signals to different carrier frequencies. The higher the carrier frequency, the more bandwidth that is available for many baseband signals. Path loss, penetration of obstacles, reflection, scattering, and diffraction all these effects depend heavily on the wavelength of the signal. Depending on the application, the right carrier frequency with the desired characteristic has to be chosen: long waves for submarines, short waves for handheld devices, very short waves for directed microwave transmission. For these reasons, carrier or bandpass modulation is an essential step for most systems involving radio transmission [28].

The carrier modulation is accomplished by multiplying the spread baseband signal  $s(t)$  with a sinusoidal carrier of the form  $\cos(2 \pi f_c t)$ ,

$$y(t) = s(t) \cos(2 \pi f_c t) = v(t) c(t) \cos(2 \pi f_c t) \quad (2.5)$$

where  $f_c$  is the carrier frequency.

## 2.2 Channel

The channel is the physical medium used to convey the signal from the transmitter to the receiver. In radio transmission, channel is usually the free space. In this thesis, we emphasize on four types of channel effects on the transmitted signal: additive noise, channel propagation delay, fading, and multipath propagation.

Thermal noise is the predominant noise and unavoidable source for all communication systems. Its characteristics (additive, white and Gaussian, giving rise to the name AWGN) are most often used to model the noise in the detection process and in the design of receivers. An AWGN channel is assumed in our work, which means the noise is a Gaussian random process with power spectral density,

$$G_n(f) = N_0 / 2 \quad (2.6)$$

which is flat (constant) over the channel bandwidth.

Signals will experience propagation delays when arriving at the receiver end as long as the transmitter and receiver are not placed at the same location. The delay is directly proportional to the distance between the transmitter and receiver, thus vary with time in wireless communication systems due to the mobility of the mobile users.

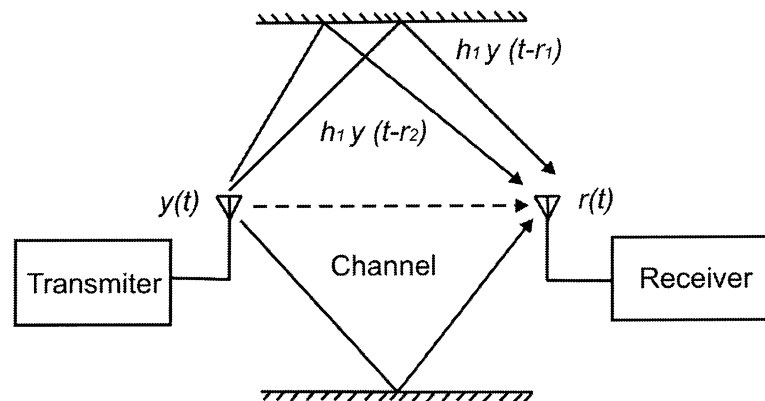


Figure 2.7 Multipath propagation

The presence of reacting objects and scatters in the channel will introduce multipath propagation. When a signal leaves the transmitting antenna, it can take a number of different paths with different delays to reach the receiver, as shown in Figure 2.7.

The difference in delay will result in different phases between received signals. If the phase difference approaches 180 degree, the signals will add destructively, resulting in a phenomenon called signal fading; while if the phase difference approaches 0, they will add constructively. If the duration of a modulated symbol is much greater than the time spread of the propagation path delays, e.g., in narrow-band transmission systems, the phase variations between different frequency components in the transmitted signal will be small and they will all undergo very similar amount of fading. This is so called flat fading. On the other hand, if the symbol duration is of the same order or even smaller than the multipath delay spread, e.g., in wide-band transmission systems, the frequency components in the transmitted signal will experience different phase shifts along different paths. The channel does not have a constant frequency response over the bandwidth of the transmitted signal; it creates frequency-selective fading on the received signal. When this occurs, the received signal includes multiple version of the transmitted waveform, which are attenuated (faded) in time. The channel can thus be modeled with several filter taps, which represent attenuation along each of the delay path [25]. Time dispersion of the transmitted symbols results in intersymbol interference (ISI). Let us consider a  $P$ -path model illustrated in Figure 2.7.

The received signal  $r(t)$  is given by a sum of delayed components

$$r(t) = \sum_{i=1}^P h_i y(t-\tau_i) \quad (2.7)$$

where  $h_i$ ,  $i = 1; 2; \dots; P$  is the fading coefficient of the  $i$ th path,  $y(t)$  is the transmitted waveform, and  $\tau_i$  is the propagation delay of the  $i$ th path [7].

### 2.2.1 Rayleigh and Rician Fading Channel

Rayleigh and Rician fading channels are useful models of real-world phenomena in wireless communications. These phenomena include multipath scattering effects,

time dispersion, and Doppler shifts that arise from relative motion between the transmitter and receiver.

Rayleigh fading is a statistical model for the effect of a propagation environment on a radio signal, such as that used by wireless devices. It assumes that the magnitude of a signal that has passed through such a transmission medium (also called a communications channel) will vary randomly, or fade, according to a Rayleigh distribution. It is a reasonable model for tropospheric and ionospheric signal propagation as well as the effect of heavily built-up urban environments on radio signals. Rayleigh fading is most applicable when there is no line of sight between the transmitter and receiver. If there is a line of sight, Rician fading is more applicable. Rayleigh fading occurs when there is no line of sight signal, and is sometimes considered as a special case of Rician fading [1],[5],[13].

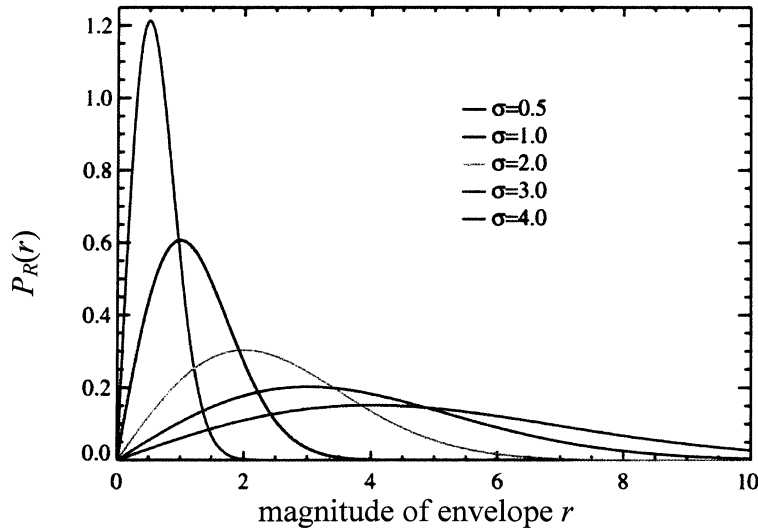


Figure 2.8 Rayleigh probability density functions for various  $\sigma$

The Rayleigh probability density function is given by

$$P_R(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) \quad r \geq 0 \quad (2.8)$$

where  $\sigma^2=P_f$  is the variance or diffuse fading received signal power and  $r$  is the amplitude of received signal,  $r(t)$ . Figure 2.8 illustrates Rayleigh probability density functions of the received amplitude,  $r(t)$  for various  $\sigma$ .

In Rician fading, the signal arrives at the receiver by two different paths, and at least one of the paths is changing (lengthening or shortening). Rician fading occurs when one of the paths, typically a line of sight signal, is much stronger than the others.

The Rician probability density function is given by

$$P_R(r) = \frac{r}{\sigma^2} \exp\left(-\frac{r^2 + v^2}{2\sigma^2}\right) I_0\left(\frac{rv}{\sigma^2}\right) \quad r \geq 0 \quad (2.9)$$

where the parameter  $v^2$  represent the power of received non fading signal component.  $I_0(.)$  is defined as the modified zero-order Bessel function of the first kind.

Figure 2.9 and Figure 2.10 illustrates Rician probability density functions of the received amplitude,  $r(t)$  for various  $v$  with  $\sigma=1$  and 0,25 respectively.

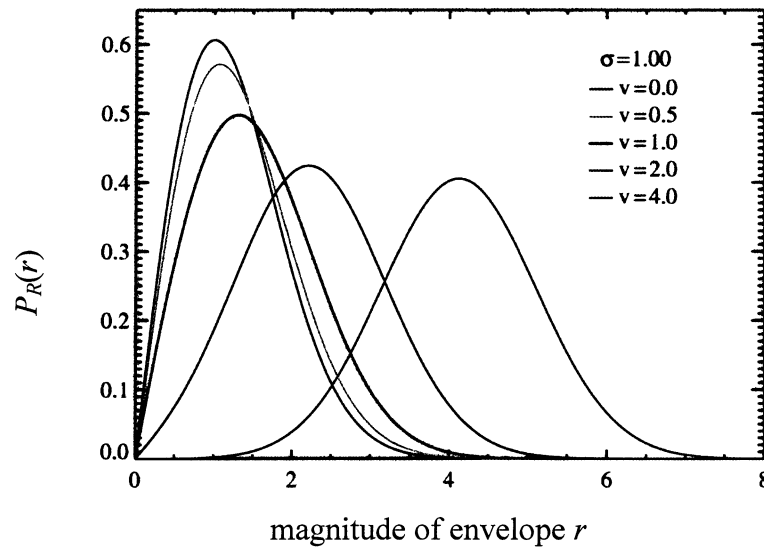


Figure 2.9 Rician probability density functions for various  $v$  with  $\sigma = 1$

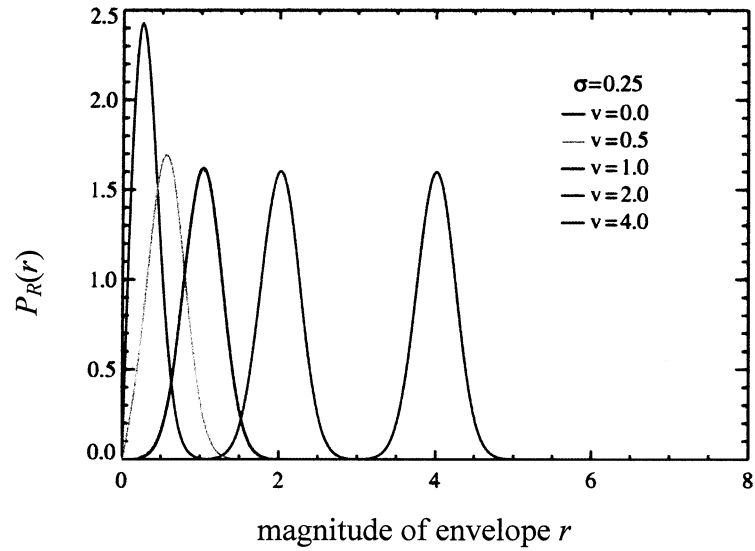


Figure 2.10 Rician probability density functions for various  $\nu$  with  $\sigma=0,25$

The total received power is

$$P = P_o + P_f \quad (2.10)$$

where  $P_o$  is the dominant specular received power component and  $P_f$  is the diffuse fading received power component [1].

Rician factor  $KK$  is the ratio of the power in the direct line-of-sight component to the local-mean scattered power.

Rician  $KK$ -factor is

$$KK = \nu^2 / \sigma^2 = P_o / P_f \quad (2.11)$$

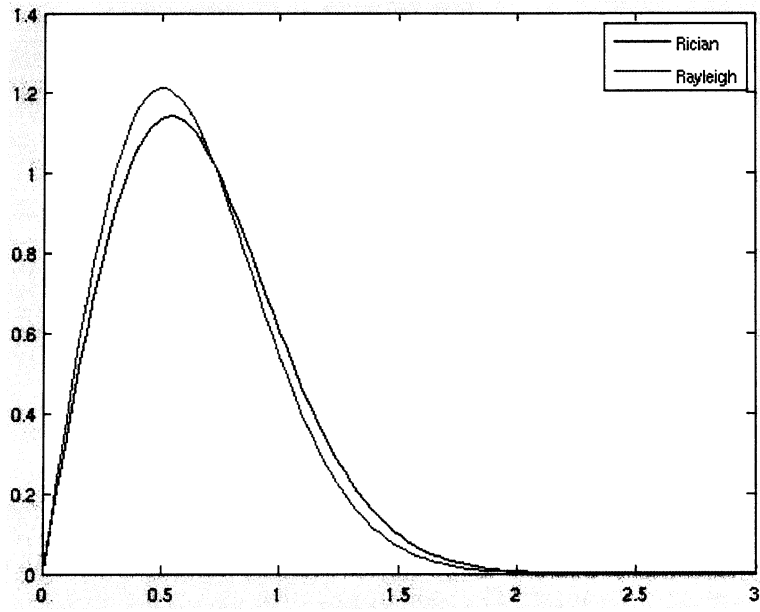


Figure 2.11 Comparison of Rayleigh and Rician at  $v / \sigma = 0,5$

When  $KK$  approaches zero, the Rician distribution reduces to a Rayleigh distribution. Figure 2.11 illustrates comparison of Rayleigh and Rician probability density function at  $v / \sigma = 0,5$ .

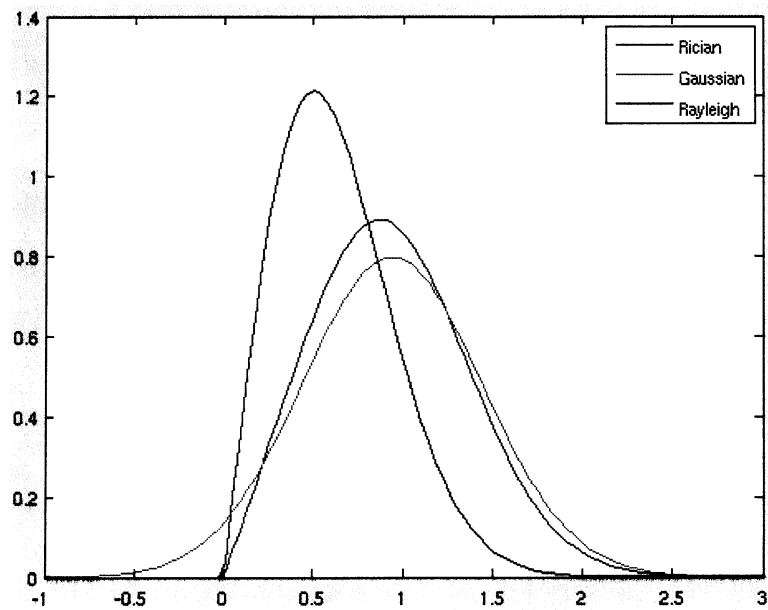


Figure 2.12 Comparison of Rician, Gaussian and Rayleigh at  $v / \sigma = 1,5$

Figure 2.12 illustrates comparison of Gaussian, Rician and Rayleigh probability density functions at  $v / \sigma = 1,5$ .

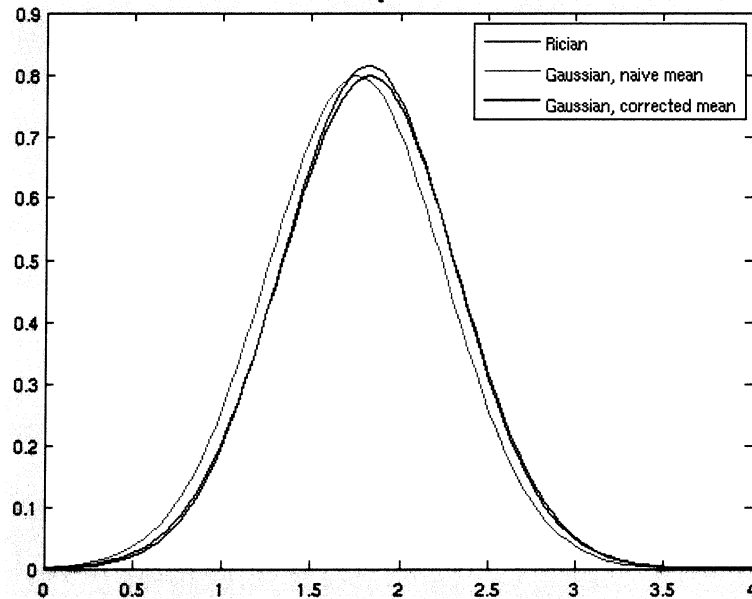


Figure 2.13 Comparison of Gaussian and Rician at  $v/\sigma=3,5$

At high SNR, Rician data is approximately Gaussian. Figure 2.13 illustrates comparison of Gaussian and Rician probability density functions at  $v / \sigma = 3,5$ .

## 2.3 Receiver

Receiver which recovers the message signal contained in the received signal. The functional blocks in the receiver include down mixing, despreading, demodulation, and decoding as depicted in Figure 2.1. They are more or less logically the inverse operations of the transmitter in reverse order [7].

### 2.3.1 Carrier Demodulation

Carrier demodulation is the inverse process of carrier modulation. It performs frequency translation to transform bandpass signals to baseband signals. The received signal,  $r(t)$  is demodulated by a carrier  $\cos(2 \pi f_c t)$



$$x_f(t) = r(t) \cos(2\pi f_c t) = b(t) c(t) \cos^2(2\pi f_c t) = 1/2 b(t) c(t) [1 + \cos(4\pi f_c t)] \quad (2.12)$$

The double frequency component in the above equation is filtered out by the LPF. In this way, the baseband signal

$$x(t) = b(t) c(t) \quad (2.13)$$

is extracted from the sinusoidal carrier.

### 2.3.2 Despreading

Despreading is the process to retrieve the information bearing signal  $v(t)$  from the scrambled signal

$$x(t) = b(t) c(t) \quad (2.14)$$

at the receiver. It is performed as illustrated in Figure 2.14.

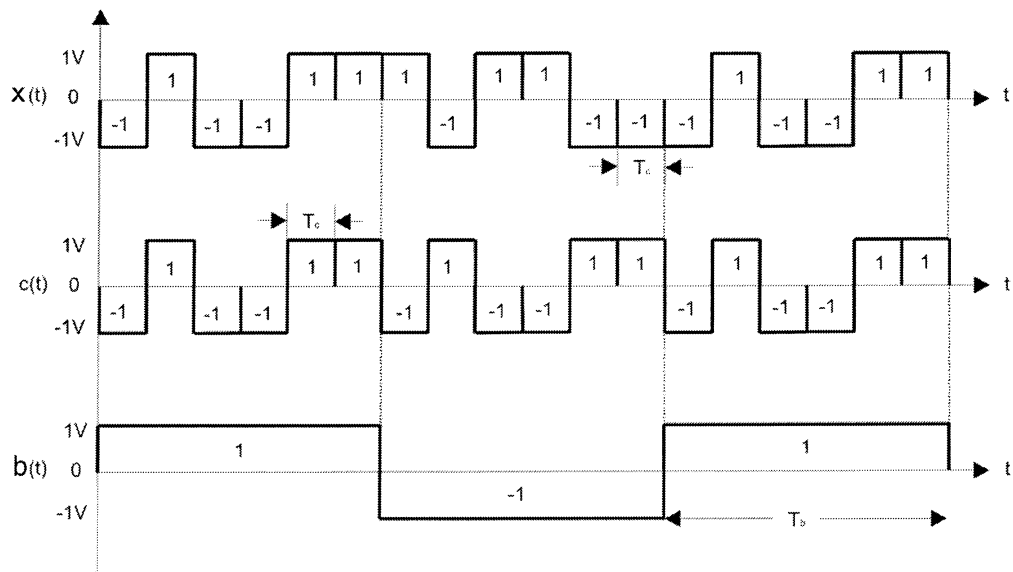


Figure 2.14 Despreading

The baseband signal  $x(t)$  is multiplied with a replica of the spreading signal  $c(t)$  generated by the PN code generator at the receiver, which is synchronized to the PN code in the received signal, such that

$$x(t) c(t) = b(t) c^2(t) = b(t) = v(t) \quad (2.15)$$

since  $c^2(t) = 1$  for all  $t$ .

In practice, the received signal should have the form:

$$r(t) = \sum_{i=1}^{L_c} y(t-\tau_i) h_i(t) + n(t) \quad (2.16)$$

where  $y(t)$  is the transmitted signal,  $\tau_i$  is the propagation delay of the  $i$ th path,  $h_i(t)$  is its corresponding fading process, and  $n(t)$  is the AWGN. In order to get a basic understanding of the principles of the spread spectrum communications, we only consider the ideal situation here and ignore the propagation delays, noise, fading and all other kinds of channel impairments.

### 2.3.3 Channel Decoding

Viterbi algorithm is the best-known implementation of the maximum likelihood decoding for convolutional codes. It can be illustrated with trellis diagram, which is a way to show the transition between various states as the time evolves. The Viterbi decoder finds a path through the trellis that is at minimum distance from a given sequence.

Decoding can be hard-decision decoding or soft-decision decoding, depending on whether Hamming distance or Euclidean distance is used as the metric to be minimized.

In soft decoding,  $r$ , the vector denoting the outputs of the matched filter is compared to the various signal points in the constellation of the coded modulation system, and the one closest to it in Euclidean is chosen.

In hard decision,  $r$ , is first turned into a binary sequence  $y$  by making decision on individual components of  $r$ , and then the code word, which is closest to  $y$  in the hamming distance, is chosen. It is seen that in both approach, a fundamental task is to find a path through the trellis that is at minimum distance from a given sequence [11].

In the Trellis diagram, the solid lines denote state transitions given input information bit 0, and the dashed lines denote state transitions given input information bit 1. The corresponding code bits are written underneath each transition line. The branches with letter 'D' are deleted after metric comparison. The number under each node is the accumulated metric value. The path drawn with red bold line is the final survivor's path. The last two 0s of the information bits are used to return the encoder to all-zero state, so we only considered the 0 inputs to the encoder in drawing the trellis in the last two stages.

Figure 2.15 shows an example of decoding process. Assume information bits 1 0 1 0 0 0 were convolutionally coded into sequence 11 10 00 10 11 00, which was transmitted over the channel. For simplicity, we demonstrate the hard-decision decoding, in which the Hamming distance is used for branch metric. After making hard decision on the received signal, the quantized received sequence becomes 11 11 00 10 10 10. The errors occur for the underlined bits due to the channel impairments (noise, fading, etc.). Starting from all-zero states 0, the decoder computes metrics (the Hamming distance) between all possible paths and the received sequence. The metrics are cumulative along the nodes of each path. When two branches enter the same node, the one with lower accumulated metric remains as a survivor and the other branch is eliminated. At the end, the path with minimum distance to the received sequence is selected to be the optimum trellis path. The information sequence will then be determined.

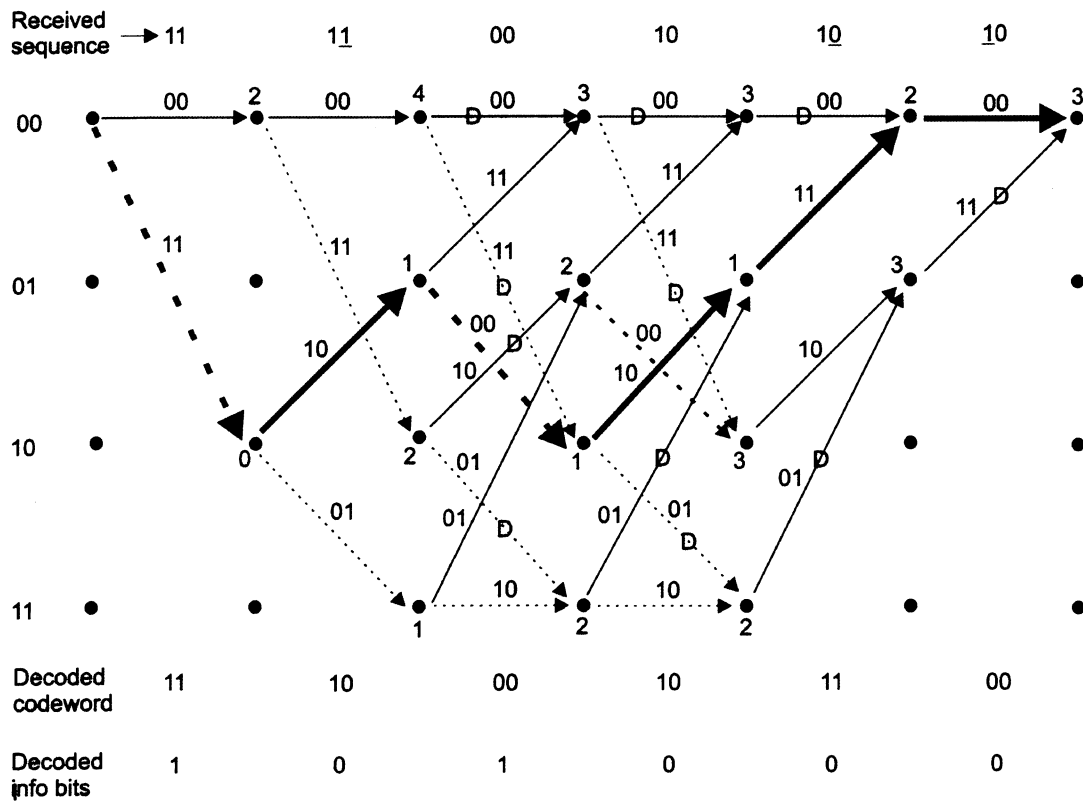


Figure 2.15 Trellis diagram for Viterbi decoding

In our example, the path drawn with bold red color lines through the trellis is selected. The codeword for this path is 11 10 00 10 11 00, which is at Hamming distance 3 from the received sequence. The corresponding information sequence is 1 0 1 0 0 0. Three bits errors in the codeword are corrected and the original information sequence is recovered.

When the channel propagation delay, MAI, noise, multipath, and fading are considered as in the real systems, the demodulation and decoding processes become much more sophisticated.

## 2.4 CDMA Receivers

A CDMA receiver separates the signals by means of a correlator that uses the particular binary sequence to despread the signal and collect the energy of the desired signal. Other users' signals, whose spreading codes do not match this sequence, are

not despread in bandwidth and, as a result, contribute only to the noise. These signals represent a self-interference generated by the system.

The output of the correlator is sent to a narrow-bandwidth filter. The filter allows all of the desired signal's energy to pass through, but reduces the interfering signal's energy by the ratio of the bandwidth before the correlator to the bandwidth after the correlator. This reduction greatly improves the signal-to-interference ratio of the desired signal.

The signal-to-noise ratio is determined by the ratio of the desired signal power to the sum of all of the other signal powers. It is enhanced by the processing gain or the ratio of spread bandwidth to baseband data rate.

The capacity of CDMA system is interference limited by multiple accesses interference (MAI) and multiple channel distortion. That is, when a new user or interferer for the other users enters into channel, the service quality of the other users will degrade.

Multiuser detection (MUD) seeks to overcome the inherent shortcomings of conventional CDMA receivers by providing near-far resistance for the receiver in the process of eliminating the MAI.

#### **2.4.1 Conventional Single User Detector**

In conventional single user detection, the receiver for each user consist of a demodulator that correlates (or match filters) the received signal with the signature sequence of the user and passes the correlator output to the detector, which makes a decision based on the single correlator output. Thus, convectional detector neglects the presence of the users of the channel or, equivalently, assumes that the aggregate noise plus interference is white and Gaussian.

If the signature sequences are orthogonal, the interference from the other users vanishes and the conventional single user detector is optimum. On the other hand, if one or more of the other signature sequences are not orthogonal to the user signature sequence, the interference from the other users can become excessive if the power levels of the signals (or the received signal energies) of one or more of the other users is sufficiently larger than the power level of the  $k$ th user. This situation is generally called the near- far problem in multiuser communications, and necessitates some type of power control for conventional detection [5].

#### **2.4.2 Optimum Receiver**

An optimum maximum likelihood (ML) MUD receiver was proposed by Verdú. This detector is based on the maximum likelihood sequence detection formulation. The complexity of the ML receiver is exponential with the number of users, rendering it impractical [5].

#### **2.4.3 Suboptimum Linear Receiver**

Suboptimum linear receivers such as the decorrelating and the minimum mean-squared error (MMSE) receivers have been proposed to trade off complexity and performance among the conventional and optimal receivers, however, they still require computationally intensive matrix inversion [5].

##### **2.4.3.1 Minimum Mean-Square Error (MMSE) Receiver**

Minimum mean-square error receiver is used to minimize mean square error between the transmitted signal and the output signal by using additional sequences to provide better performance [5].

### 2.4.3.2 Decorrelating Detector

The conventional detector has a complexity that grows linearly with the number of users, but its vulnerability to the near-far problem requires some type of power control. Decorrelating detector also has a linear computational complexity but does not exhibit the vulnerability to other user interference. In the decorrelating detector, the crosscorrelation matrix is assumed to be known. The match filter outputs are multiplied by the inverse of the crosscorrelation matrix. The disadvantage of this detector is that the noise energy is increased after the linear transformation, resulting performance degradation in the presence of noise. Figure 2.16 illustrates the receiver structure of decorrelating detector.

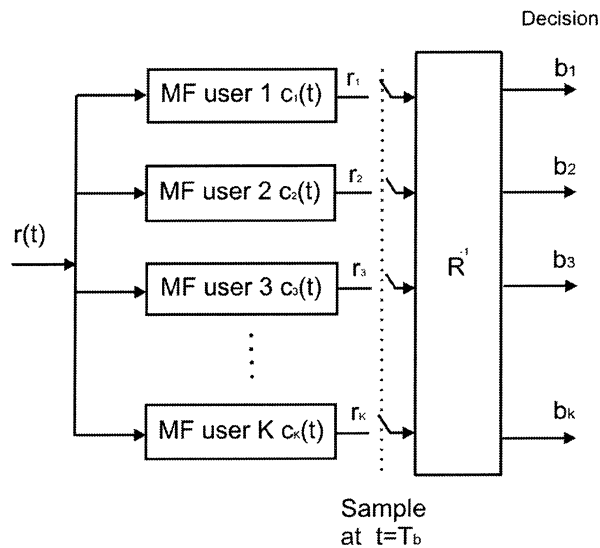


Figure 2.16 Decorrelating detector

## 2.5 Power Control in CDMA

Power control in general is all about controlling the transmitted power both in downlink and uplink direction for three reasons, which are near-far problem, system level interference leads to reduced system capacity and limited power source of user

equipment (UE). Main aim of the power control is to transmit the signal with lowest possible power level, which maintains the required signal quality.

The purpose of the power control is mainly to reduce the intra-cell interference. Because in the CDMA system the total bandwidth is shared simultaneously, other users can be experienced as a noise like interference. In case power control mechanism is missing, common sharing of bandwidth creates a severe problem referred to as near-far effect.

All signals should arrive at the base station's receiver with the same signal power. The mobile stations cannot transmit using fixed power levels, because the cells would be dominated by users closest to base station and faraway users could not get their signals heard in the base station. This phenomenon is called near-far effect. The main factors, that causes the near-far problem results from the path-loss variation of simultaneous users with different distances from the base station (BS), fading variation, and other signal-power variation of the users caused by the radio wave propagation.

In the uplink case, the optimal situation from the BS point of view is that the power representing one UE's signal is always equal when compared to the other UE signal regardless of their distance from the BS. In such cases SIR will be optimal and BS receiver is able to decode the maximum number of transmissions. However, in reality, the radio channel is extremely unstable and therefore the transmission power of UE should be controlled very accurately by utilizing efficient mechanisms.

In CDMA, power control is employed in both the uplink and the downlink. The main target of the uplink power control is to mitigate the near-far problem by making the transmission power level received from all terminals as good as possible at the home cell for the same QoS. The mobile stations far away from the base station should transmit with considerably higher power than the mobiles close to the base station. Therefore uplink power control is for fine tuning of terminal transmission power, resulting in the mitigation of the intra-cell interference and near-far effect.



The situation is different in downlink direction. The downlink signals transmitted by one base station are orthogonal. Signals that are mutually orthogonal do not interfere with each other. However, it is impossible to achieve full orthogonality in typical usage environments. Signal reflections cause non orthogonal interference even if one base station is considered. Moreover signals sent from other base stations are, of course, non orthogonal and thus they increase the interference level taking into account that in a CDMA system the neighbor cells use the same downlink frequency carrier. This calls for downlink power control. There are two basic types of power control, which are the open loop power control technique, and the closed-loop power control technique.

The open loop power control technique requires that the transmitting entity measures the channel interference and adjusts its transmission power accordingly. In this process, the UE estimates the transmission signal strength by measuring the received power level of the pilot signal from the BS in the downlink, and adjusts its transmission power level in a way that is inversely proportional to the power level of pilot signal. Consequently, the stronger the received pilot signal, the lower the UE transmitted power.

In the closed-loop power control technique, the quality measurements are done on the other end of the connection in the base station and the results are then sent back to mobile's transmitter so that it can adjust its transmitted power. This method gives much better results than the open loop method but it cannot react to quick changes in the channel conditions [29].

## CHAPTER 3

### SIMULATION RESULTS

#### 3.1 Simulation Parameters

This chapter discusses to calculate the probability of error via computer simulation of convolutional coded (code rates are 1/2 and 2/3) DS-CDMA with decorrelator detector communication system in the two types of channels, which are additive white Gaussian noise channel (AWGN), and Rician channel. A Monte Carlo (MC) method of simulation based on the computation of distance metrics is used. For generation and detection mechanisms, we benefit from the built-in MATLAB utilities.

- All simulations are executed in two types of channels, the additive white Gaussian noise (AWGN) channel and the Rician fading channel. For the simulations in the Rician fading channel, the channel is designed with a Rician parameters of  $KK=1, 10, 100$ . This parameter is defined as

$KK = \text{power of dominant path} / \text{power in scattered path}$

When  $KK=0$ , the channel is Rayleigh, and if  $KK$  is infinite, the channel is Gaussian.

- Bit error probability  $P_e$  (BER) is calculated as a function  $E_b/N_o$  (SNR) for the convolutional code rate (CR) of 1/2 and 2/3.

- For the decoding of convolutional code, hard decision Viterbi algorithm is used.

- To simplify simulations, we assumed that power control is optimum. All users supposed to arrive at the base station with the same power level.
- Synchronous system is assumed that the delays between users are zero.
- Two different lengths of Gold sequences which are  $T = 31$  and  $127$  are used.
- The numbers of simultaneous active users of  $K$  in the system are  $5$  and  $25$ .
- As the multiuser detection, decorrelator detector is chosen. In the decorrelator detector, the match filter outputs are multiplied by the inverse of the crosscorrelation matrix.
- It is assumed that modulation is binary phase shift keying (BPSK).
- The number of bits  $N$  that should be chosen depends on the expected level of errors, being inversely proportional to it. In order to obtain a probability of error,  $P_e$ , that will represent the practical situations accurately within a confidence interval of 90%, one needs to keep  $N$  at least at  $10/P_e$  or above. If for instance,  $P_e = 10^{-4}$ , then  $N = 10^5$  [6]. For this reason, to get  $P_e = 10^{-4}$ , each user transmits  $N = 100.000$  bits for each simulation.
- The parameter bit error rate is used to evaluate system performance. To obtain BER at each value of SNR for each simulation; transmitted bits are compared with the estimated bits by detector. If they are the same, there is no error. Otherwise, there are errors. Number of errors are summed and average  $P_e$  is obtained by dividing the error bits by  $K * N$ .

### 3.2 Performance Analysis of Uncoded System

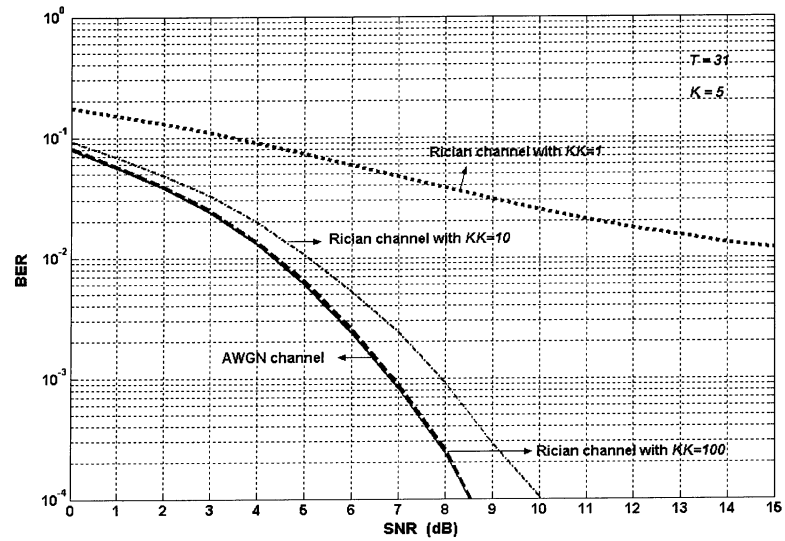


Figure 3.1 BER performance graph of uncoded system,  $T=31$ ,  $K=5$

The simulation shown in Figure 3.1 is performed for uncoded system with the length 31 of Gold code for 5 users in AWGN and Rician channel. Table 3.1 consisting of some selected SNR-BER values from Figure 3.1.

Table 3.1 Some selected SNR-BER values from Figure 3.1

BER	SNR (dB)-Signal to Noise Ratio			
	AWGN channel	Rician channel with $KK=100$	Rician channel with $KK=10$	Rician channel with $KK=1$
$10^{-2}$	4,5	4,53	5,1	>15
$10^{-3}$	6,82	6,87	7,9	>15
$10^{-4}$	8,5	8,6	10	>15

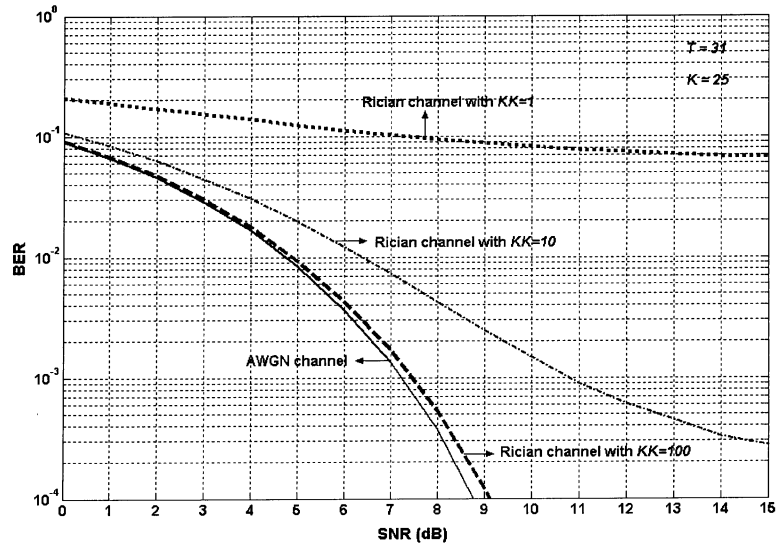


Figure 3.2 BER performance graph of uncoded system,  $T=31$ ,  $K=25$

The simulation shown in Figure 3.2 is performed for uncoded system with the length 31 of Gold code for 25 users in AWGN and Rician channel. Table 3.2 consisting of some selected SNR-BER values from Figure 3.2.

Table 3.2 Some selected SNR-BER values from Figure 3.2

BER	SNR (dB)-Signal to Noise Ratio			
	AWGN channel	Rician channel with $KK=100$	Rician channel with $KK=10$	Rician channel with $KK=1$
$10^{-2}$	4,7	4,8	6,4	>15
$10^{-3}$	7,3	7,5	10,8	>15
$10^{-4}$	8,8	9,1	>15	>15

If the PN sequences are perfectly orthogonal, there is no interference between the users. In practice, i.e. Gold sequences are not perfectly orthogonal hence; the crosscorrelation is not zero, which produces multiple access interference (MAI).

As seen from Table 3.1 and Table 3.2, it is clear that when the number of user increases, the performance of decorrelating detector becomes poor because of the MAI, although this degradation is minimal.

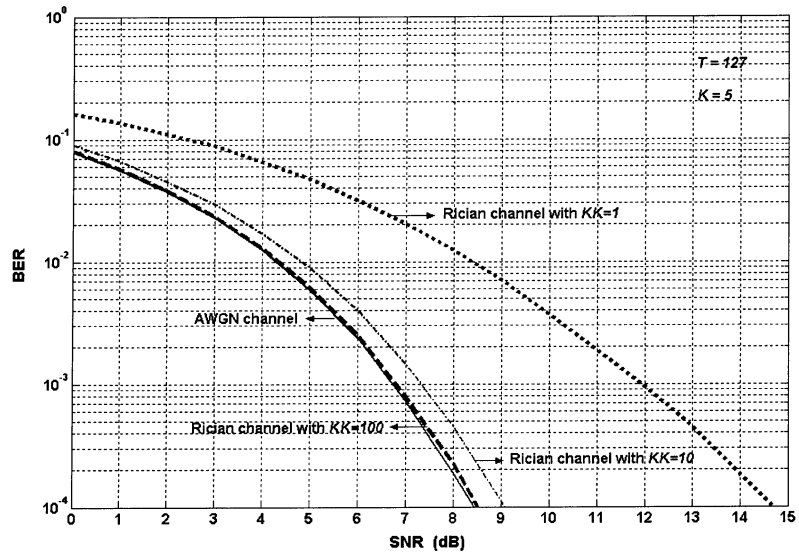


Figure 3.3 BER performance graph of uncoded system,  $T=127$ ,  $K=5$

The simulation shown in Figure 3.3 is performed for uncoded system with the length 127 of Gold sequence for 5 users in AWGN and Rician channel. Table 3.3 consisting of some selected SNR-BER values from Figure 3.3.

Table 3.3 Some selected SNR-BER values from Figure 3.3

BER	SNR (dB)-Signal to Noise Ratio			
	AWGN channel	Rician channel with $KK=100$	Rician channel with $KK=10$	Rician channel with $KK=1$
$10^{-2}$	4,3	4,33	4,8	8,4
$10^{-3}$	6,75	6,8	7,4	11,9
$10^{-4}$	8,3	8,4	9,1	14,8

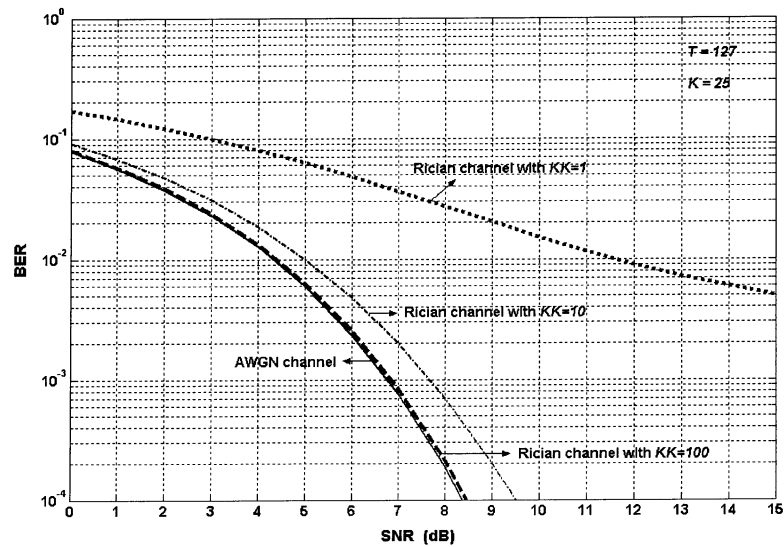


Figure 3.4 BER performance graph of uncoded system,  $T=127$ ,  $K=25$

The simulation shown in Figure 3.4 is performed for uncoded system with the length 127 of Gold code for 25 users in AWGN and Rician channel. Table 3.4 consisting of some selected SNR-BER values from Figure 3.4.

Table 3.4 Some selected SNR-BER values from Figure 3.4

BER	SNR (dB)-Signal to Noise Ratio			
	AWGN channel	Rician channel with $KK=100$	Rician channel with $KK=10$	Rician channel with $KK=1$
$10^{-2}$	4,4	4,4	5	11,5
$10^{-3}$	6,8	6,9	7,7	>15
$10^{-4}$	8,4	8,5	9,5	>15

When the spreading sequence length of Gold code increases, crosscorrelation of Gold sequences is lowered hence, performance is increased. Comparison of  $T=31$  and 127 length of Gold sequences simulations, as it is seen clearly from the all Figure, long code  $T=127$  has better performance than the lower code length  $T=31$ .

### 3.3 Performance Analysis of 2/3 Convolutional Coded System

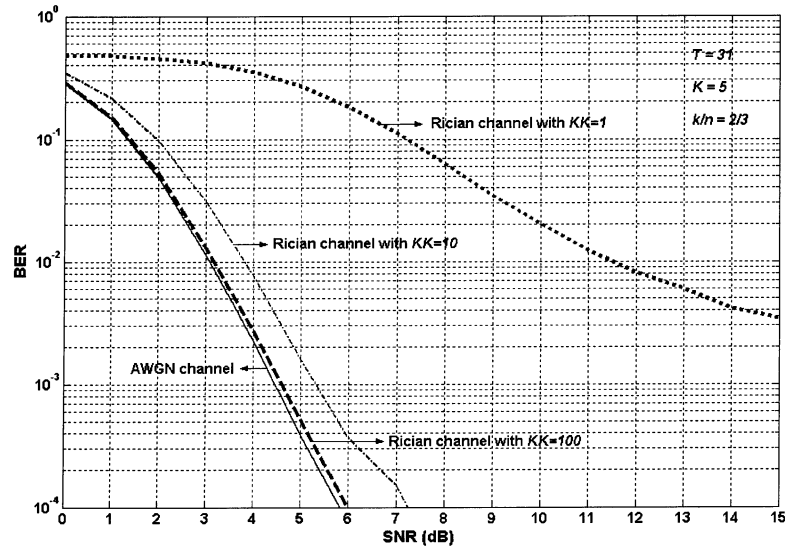


Figure 3.5 BER performance graph of 2/3 convolutional coded system,  $T=31$ ,  $K=5$

The simulation shown in Figure 3.5 is performed for 2/3 convolutional coded system with the length 31 of Gold code for 5 users in AWGN and Rician channel. Table 3.5 consisting of some selected SNR-BER values from Figure 3.5.

Table 3.5 Some selected SNR-BER values from Figure 3.5

BER	SNR (dB)-Signal to Noise Ratio			
	AWGN channel	Rician channel with $KK=100$	Rician channel with $KK=10$	Rician channel with $KK=1$
$10^{-2}$	3,15	3,25	3,8	11,5
$10^{-3}$	4,5	4,6	5,3	>15
$10^{-4}$	5,87	6	7,3	>15

It is seen from Figure 3.1 and Figure 3.5 that in AWGN channel, at an  $E_b/N_o$  (SNR) of approximately 2.3 dB indicating that, the communication system are better off not using at the code rate 2/3 for forward error correcting at low signal to noise ratios.



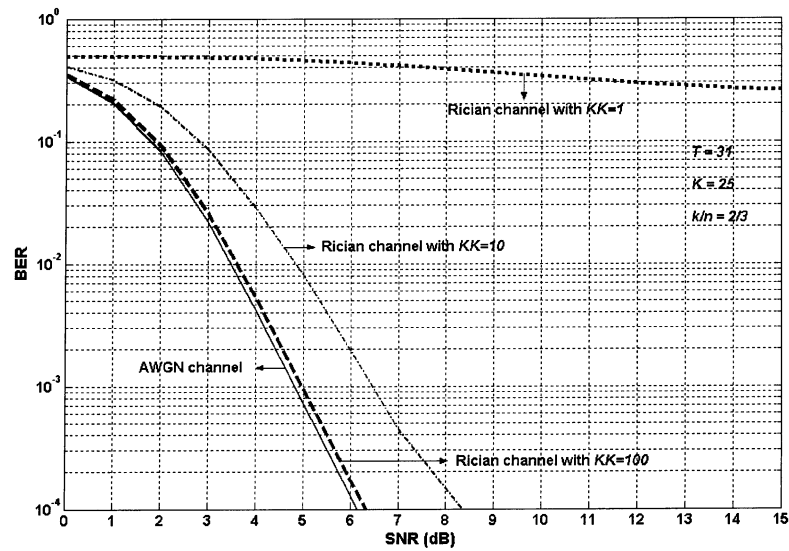


Figure 3.6 BER performance graph of 2/3 convolutional coded system,  $T=31$ ,  $K=25$

The simulation shown in Figure 3.6 is performed for 2/3 convolutional coded system with the length 31 of Gold code for 25 users in AWGN and Rician channel. Table 3.6 consisting of some selected SNR-BER values from Figure 3.6.

Table 3.6 Some selected SNR-BER values from Figure 3.6

BER	SNR (dB)-Signal to Noise Ratio			
	AWGN channel	Rician channel with $KK=100$	Rician channel with $KK=10$	Rician channel with $KK=1$
$10^{-2}$	3,5	3,6	4,8	>15
$10^{-3}$	4,8	5	6,5	>15
$10^{-4}$	6,2	6,3	8,4	>15

It is observed from the Figure 3.2 and Figure 3.6 that the coding gain is not constant so that coding gain changes with signal to noise ratio. At a bit error probability of  $10^{-3}$  in AWGN channel, coding gain is about 2,5 dB and at a bit error probability of  $10^{-4}$ , coding gain is about 2,6 dB.

The simulation shown in Figure 3.7 is performed for 2/3 convolutional coded system with the length 127 of Gold sequence for 5 users in AWGN and Rician channel.

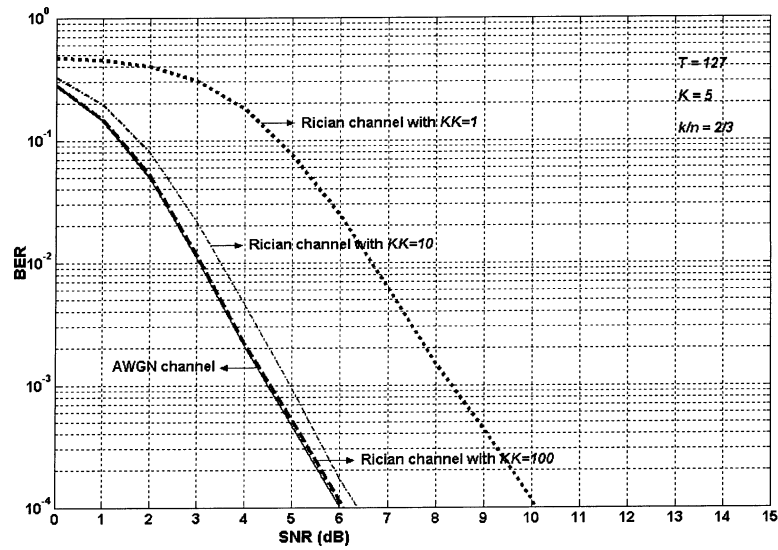


Figure 3.7 BER performance graph of 2/3 convolutional coded system,  $T=127$ ,  $K=5$

The simulation shown in Figure 3.8 is performed for 2/3 convolutional coded system with the length 127 of Gold sequence for 25 users in AWGN and Rician channel.

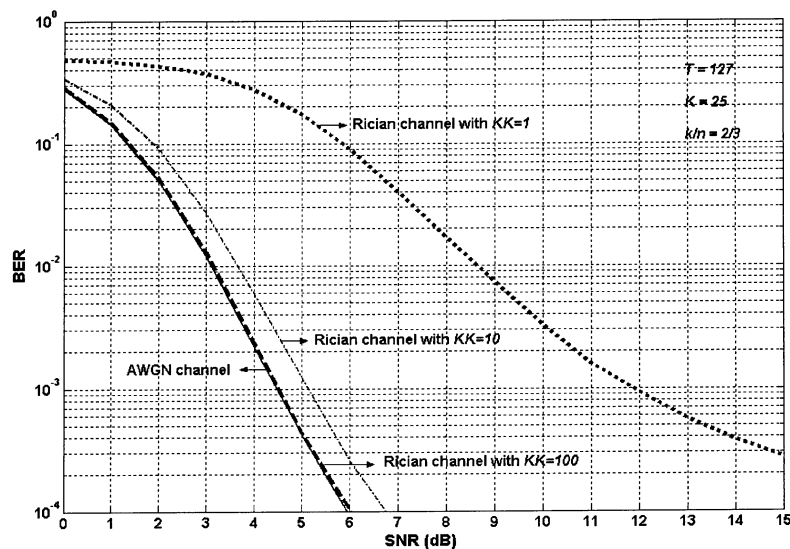


Figure 3.8 BER performance graph of 2/3 convolutional coded system,  $T=127$ ,  $K=25$

### 3.4 Performance Analysis of 1/2 Convolutional Coded System

The simulation shown in Figure 3.9 is performed for 1/2 convolutional coded system with the length 31 of Gold sequence for 5 users in AWGN and Rician channel.

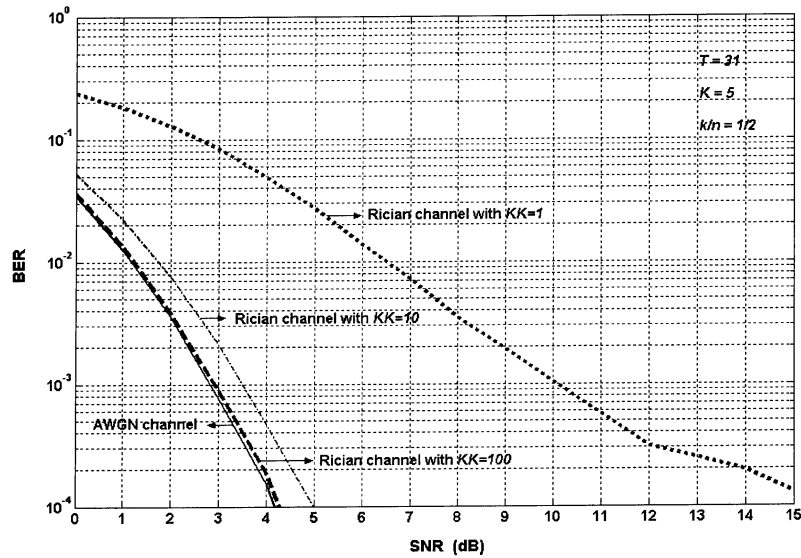


Figure 3.9 BER performance graph of 1/2 convolutional coded system,  $T=31, K=5$

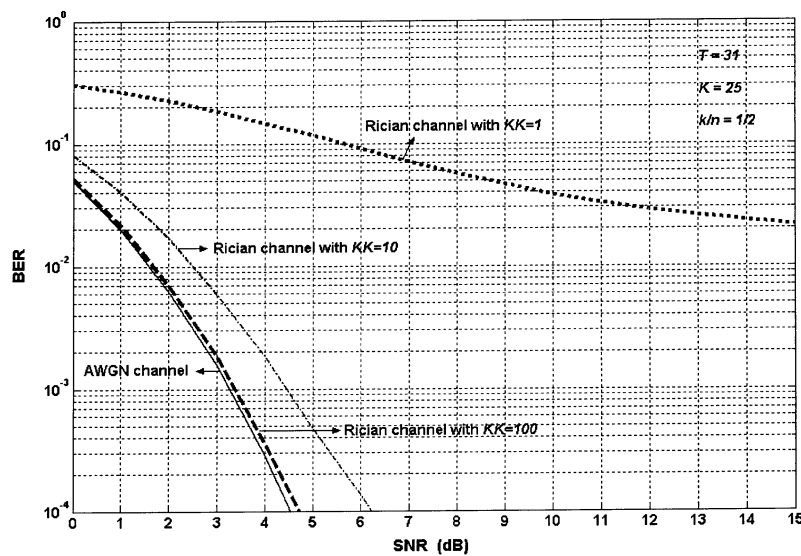


Figure 3.10 BER performance graph of 1/2 convolutional coded system,  $T=31, K=25$

The simulation shown in Figure 3.10 is performed for 1/2 convolutional coded system with the length 31 of Gold sequence for 25 users in AWGN and Rician channel. Table 3.7 consisting of some selected SNR-BER values from Figure 3.10.

Table 3.7 SNR-BER values from Figure 3.10

BER	SNR (dB)-Signal to Noise Ratio			
	AWGN channel	Rician channel with $KK=100$	Rician channel with $KK=10$	Rician channel with $KK=1$
$10^{-2}$	1,6	1,7	2,5	>15
$10^{-3}$	3,3	3,4	4,5	>15
$10^{-4}$	4,5	4,7	6,2	>15

It is observed from Figure 3.2 and Figure 3.10 that in AWGN channel at a bit error probability of  $10^{-3}$ , coding gain at 1/2 code rate is about 4 dB and at a bit error probability of  $10^{-4}$ , coding gain is about 4,3 dB. It is seen that if the code rate ( $CR$ ) is decrease, coding gain is increased.

A Rician channel with a high Rician factor ( $KK$ ) will gradually resemble an AWGN channel. However, as the Rician factor is smaller, it resembles the Rayleigh channel. It is seen from the all Figure that for the value of  $KK = 100$ , Rician channel behaves like AWGN channel and at the Rician factor  $KK=1$ , Rician channel behaves like a Rayleigh channel.

The simulation shown in Figure 3.11 is performed for 1/2 convolutional coded system with the length 127 of Gold sequence for 5 users in AWGN and Rician channel.

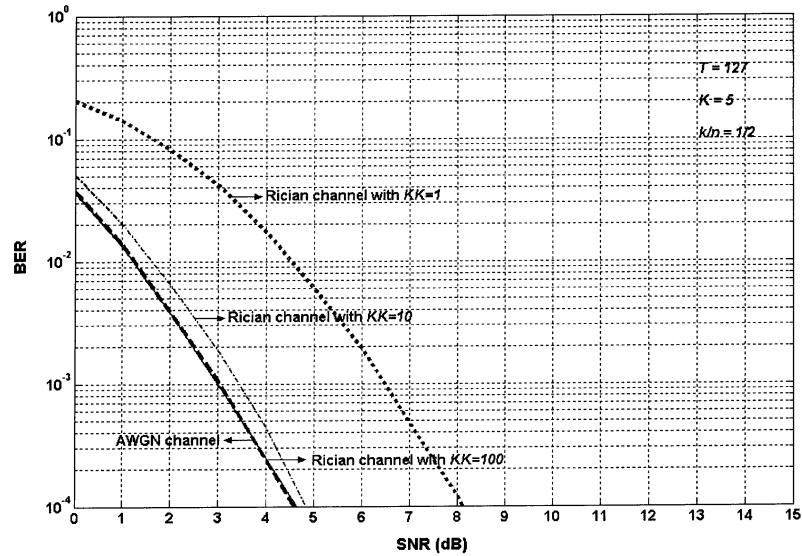


Figure 3.11 BER performance graph of 1/2 convolutional coded system,  $T=127, K=5$

The simulation shown in Figure 3.12 is performed for 1/2 convolutional coded system with the length 127 of Gold sequence for 25 users in AWGN and Rician channel.

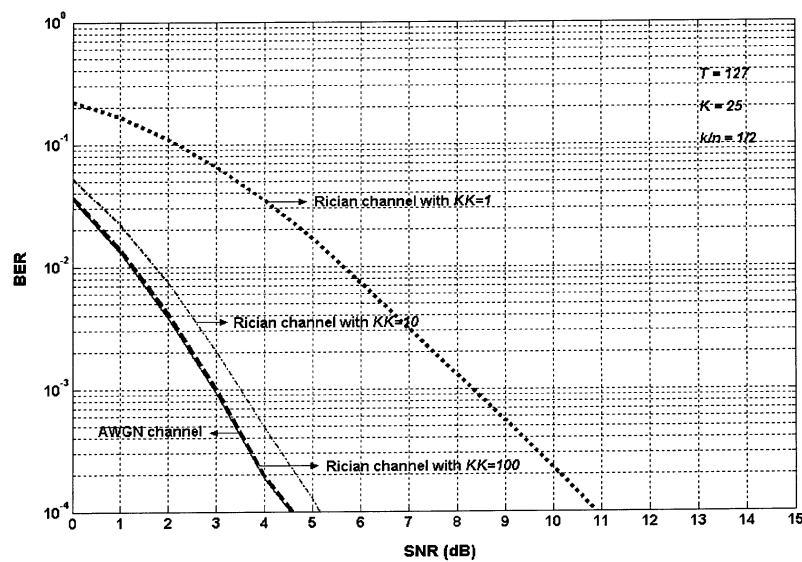


Figure 3.12 BER performance graph of 1/2 convolutional coded system,  $T=127, K=25$

## CHAPTER 4

### CONCLUSIONS

\* As depicted in all Figures, as the number of users increase from 5 to 25 in the communication system, the performance of decorrelator detector becomes poor. Decorrelating detection process suitably takes into account the interference from other users. When the user codes are orthogonal, there is no interference between the users after despreading. In practice, the codes are not perfectly orthogonal, hence the crosscorrelation between user codes introduce some performance degradation, which limits the maximum number of simultaneous users in the system.

\* Comparison of different length of Gold sequences simulations, where the length of  $T=31$  and 127 Gold sequence are used, show that the long sequence  $T=127$  has better performance than the lower sequence  $T=31$ . When the length of Gold sequence increases, crosscorrelation of the Gold sequences is lowered and more users can share the same channel simultaneously.

\* Rician channel behaves like AWGN channel at the limit of  $KK \rightarrow \infty$  and like a Rayleigh channel when  $KK \rightarrow 0$ . All simulations verify this theoretical expectation. This way numerically, Rician channel with a high  $KK=100$  factor gradually resembles an AWGN channel. But as the  $KK$  factor is made smaller,  $KK=1$ , Rician channel approaches the Rayleigh channel.

\* The purpose of using forward error correction is to enable improved communications efficiency in terms of the transmitter power necessary to achieve a required bit error probability. The coding gain of the system is the difference between the  $E_b/N_o$  (SNR) required to achieve a specified  $P_e$  (BER) without coding

and the  $E_b/N_o$  required to achieve the same  $P_e$  with coding. In simulations, 1/2 and 2/3 convolutional code rates are used for forward error correction. It is observed that firstly,  $P_e$  is typically a function of the coding gain. Secondly, if the code rate  $(CR)=k/n$  is decrease, coding gain is increased. Finally, communication system is better off not using forward error correction code at low signal to noise ratios for coding rate of 2/3.

## REFERENCES

- [1] **R. E. Ziemer and R. L. Peterson**, (2001) “*Introduction To Digital Communications*,” 2nd edition New Jersey: Prentice Hall.
- [2] **W. H. Tranter, K. S. Shanmugan, T. S. Rappaport and K. L. Kosbar**, (2004) “*Principles Of Communication System Simulation With Wireless Applications*,” New Jersey: Prentice Hall.
- [3] **J. Hagenauer**, (Sept 1996) “*Forward Error Correcting For CDMA Systems*,” *Proc. ISSSTA'96*, pp. 566-669.
- [4] **M. Moher**, (July 1998) “*An Iterative Multiuser Decoder For Near-Capacity Communications*,” *IEEE Transactions on Communications*, vol. 46, no. 7, pp. 870-880.
- [5] **J. G. Proakis**, (1995) “*Digital Communications*,” 3rd edition, New York, McGraw-Hill.
- [6] **H. T. Eyyuboğlu**, (2004) “*Multiuser Detection In CDMA And Its Simulation In Matlab Environmet*,” pp. 51-54, ICIT, İstanbul.
- [7] **Pei Xiao**, (2004) “*Iterative Detection, Decoding And Channel Parameter Estimation For Orthogonally Modulated DS-CDMA Systems*,” Doctorate Thesis, Göteborg.
- [8] **S. Verdu**, (Jan. 1986) “*Minimum Probability Of Error For Asynchronous Gaussian Multiple-Access Channels*,” *IEEE Trans. Inform. Theory* vol. IT 32, pp. 85-96.
- [9] **X. Wand and H. V. Poor**, (2004) “*Wireless Communication Systems Advanced Techniques For Signal Reception*”, New Jersey Prentice Hall, ch.1-6.



- [10] **S. G. Glisic**, (2002) “*Adaptive CDMA Theory And Practice*,” Chichester Wiley.
- [11] **J. G. Proakis and M. Salehi**, (2002) “*Communication Systems Engineering*,” 2nd ed., Prentice Hall.
- [12] **J. G. Proakis, M. Salehi and G. Bauch**, (2004) “*Contemporary Communication Systems Using MATLAB And Simulink*,” 2nd edition, Belmont, CA: Thomson.
- [13] **B. Sklar**, (1988) “*Digital Communications*,” Prentice-Hall.
- [14] **Glover and Grant**, (1997) “*Digital Communications*,” Prentice Hall.
- [15] **Viterbi**, (1995) “*CDMA Principles Of Spread Spectrum Communication*,” Addison-Wesley.
- [16] **R. E. Ziemer, W.H. Tranter**, (2002) “*Principles Of Communications*,” John Wiley & Sons.
- [17] **R. C. Dixon**, (1994) “*Spread Spectrum Systems With Commercial Applications*,” John Wiley & Sons, Inc.
- [18] **J. K. Holmes**, (1982) “*Coherent Spread Spectrum Systems*,” John Wiley & Sons.
- [19] **M. K. Simon**, (1994) “*Spread Spectrum Communications Handbook*,” Me Graw-Hill, Inc.
- [20] **Shimon Moshavi**, Bellcore, (October 1996) “*Multi-User Detection For DS-CDMA Communications*,” IEEE Communications Magazine, pp. 124-136.
- [21] **E. Dinan and B. Jabbari**, (October 1996) “*Spreading Codes For Direct Sequence CDMA And Wideband CDMA Cellular Networks*” IEEE Communications Magazine, pp. 124-136.

- [22] **S. Glisic, B. Vucetic**, (1997) “*Spread Spectrum CDMA Systems For Wireless Communications*,” Artech House.
- [23] **B. Andy**, (1999) “*Digital Communications*”, Addison-Wesley.
- [24] **P. M. Shankar**, (2001) “*Introduction To Wireless System*”, John Wiley & Sons.
- [25] **T. Rappaport**, (1996) “*Wireless Communications, Principles And Practice*,” Prentice-Hall.
- [26] **W. C. Jakes**, (1974) “*Microwave Mobile Communications*,” John Wiley & Sons, New York.
- [27] **S. Verdu**, (1998) “*Multiuser Detection*,” Cambridge Univ. Press.
- [28] **J. Schiller**, (2003) “*Mobil Communications*”, second edition, Addison-Wesley.
- [29] <http://myhsc.pbwiki.com/Power%20Control>, (01.05.2007)

## APPENDIX A

### UNCODED CDMA SYSTEM WITH DECORRELATOR DETECTOR IN AWGN AND RICIAN CHANNEL SIMULATION SOFTWARE

% Baseband Decorrelating Detector Simulation. The programme below tries to find the BER in CDMA environment with formulation based on Proakis 1995 Ch.15, Eq. (15-3-35). Note that phase coherent BPSK is used in the following, hence Rician channel function well at the limit of AWGN which is  $KK \gg 1$ , i.e.  $KK$  large, but not at Rayleigh channel limit, which is  $KK \ll 1$ , i.e.  $KK$  small, for Rayleigh channel implementation.

```
clear;clc;close all;tic
%%%%%%%%% Generating gold sequences %%%%%%%%%%%
M=2; K=5; alt=1; Rs=[]; gkt=[]; stm1=[1 0];
%T=31; genpo1=[5 2 0];genpo2=[5 4 3 2 0]; insta=[0 0 0 0 1];
T=63; genpo1=[6 1 0];genpo2=[6 5 2 1 0]; insta=[0 0 0 0 0 1];
%T=127; genpo1=[7 3 0];genpo2=[7 3 2 1 0]; insta=[0 0 0 0 0 0 1];
%T=511; genpo1=[9 4 0];genpo2=[9 6 4 3 0]; insta=[0 0 0 0 0 0 0 1];
seq_index=alt:K+alt-1;
set_param('goldgen/Gold Sequence Generator','genPoly1',genpo1)
set_param('goldgen/Gold Sequence Generator','genPoly2',genpo2)
set_param('goldgen/Gold Sequence Generator','iniState1',insta)
set_param('goldgen/Gold Sequence Generator','iniState2',insta)
for a=1:K
    set_param('goldgen/Gold Sequence Generator','index',num2str(seq_index(a)))
    sim('goldgen'); gkt=[gkt;pn(1:T)/sqrt(T)];
end
%%%%%%%%% Finding Crosscorrelation Coeffs.%%%%%%%%%%
Rs=gkt*gkt';
%%%%%%%%% Message Signal %%%%%%%%%%%
N=100000;
stpk=2*randint(K,N)-1;
ermat=zeros(K,16); ermat_ric1=zeros(K,16); ermat_ric2=zeros(K,16);
ermat_ric3=zeros(K,16);
for m=1:500
    stp=stpk(:,(m-1)*length(stpk)/500+1:m*length(stpk)/500);
    %%%%%%%%%% Spreading Operation %%%%%%%%%%%
    stm=[];
    for b=1:K
        stb=[];
        for c=1:length(stp)
```

```

stc=stp(b,c).*gkt(b,:);

stb=[stb stc];
end
stm=[stm;stb];
end
st=sum(stm,1);
stmT=stm(:,1:T);
k=[st',zeros(length(st'),1)];
st1=k(:,1)+j*k(:,2)+j*1e-10;
clear stb stc k;
%%%%%%%%% Passing via the channel %%%%%%%%%%
Fd=1; Fs=1; KK1='100'; KK2='10'; KK3='1';
Tss='N*1e-0'; Fddc='5e-2'; Fddy='5e-2'; gdB='0'; dVec='0';
set_param('Ricchannelc/Rician Fading
Channel','K',KK1,'Fd',Fddc,'gainVecdB',gdB,'delayVec',dVec);
sim('Ricchannelc');
stric1=[real(stc)];
set_param('Ricchannelc/Rician Fading Channel','K',KK2);
sim('Ricchannelc');
stric2=[real(stc)];
set_param('Ricchannelc/Rician Fading Channel','K',KK3);
sim('Ricchannelc');
stric3=[real(stc)];
clear st1 stc str Fd Fs Tss Fddc Fddy gdB dVec;
%%%%%%%%% Computation of Power, Energy Setting Noise Level %%%%%%%%%%
P_st=sum(stmT.^2,2)/T;
E_st=P_st*T;
SNR_db=0:15;
dblen=length(SNR_db);
set_ntp=10*log10(repmat(E_st(1,:),dblen,1))-SNR_db'-3.0103*ones(dblen,1);
clear P_st E_st stmT;
nt=[];
for s=1:dblen
nt1=wgn(1,length(st),set_ntp(s));
nt=[nt;nt1];
end
clear nt1;
%%%%%%%%% Received Signal %%%%%%%%%%
rt=repmat(st,dblen,1)+nt; clear st;
rtric1=repmat(stric1,dblen,1)+nt; clear stric1;
rtric2=repmat(stric2,dblen,1)+nt; clear stric2;
rtric3=repmat(stric3,dblen,1)+nt; clear stric3;
clear nt;
P_ec=[]; P_ecric1=[]; P_ecric2=[]; P_ecric3=[];
%%%%%%%%% Symbol by Symbol Detection %%%%%%%%%%
errmat=zeros(K,dblen);
for nos=1:length(stp)

```

```

rk=rt(:,(nos-1)*T+1:nos*T)*gkt';
bk_det=sign(inv(Rs)*rk');
[number,ratio,loc]=symerr(repmat(stp(:,nos),1,dblen) , bk_det);
errmat=errmat + loc;
end
eremat=eremat + errmat;
P_ec=sum(eremat)/(K*N);
errmat_ric1=zeros(K,dblen);
for nos=1:length(stp)
rkric1=rtric1(:,(nos-1)*T+1:nos*T)*gkt';
bk_detric1=sign(inv(Rs)*rkric1');
[number,ratio,loc]=symerr(repmat(stp(:,nos),1,dblen) , bk_detric1);
errmat_ric1=errmat_ric1 + loc;
end
eremat_ric1=eremat_ric1 + errmat_ric1;
P_ecric1=sum(eremat_ric1)/(K*N);
errmat_ric2=zeros(K,dblen);
for nos=1:length(stp)
rkric2=rtric2(:,(nos-1)*T+1:nos*T)*gkt';
bk_detric2=sign(inv(Rs)*rkric2');
[number,ratio,loc]=symerr(repmat(stp(:,nos),1,dblen) , bk_detric2);
errmat_ric2=errmat_ric2 + loc;
end
eremat_ric2=eremat_ric2 + errmat_ric2;
P_ecric2=sum(eremat_ric2)/(K*N);
errmat_ric3=zeros(K,dblen);
for nos=1:length(stp)
rkric3=rtric3(:,(nos-1)*T+1:nos*T)*gkt';
bk_detric3=sign(inv(Rs)*rkric3');
[number,ratio,loc]=symerr(repmat(stp(:,nos),1,dblen) , bk_detric3);
errmat_ric3=errmat_ric3 + loc;
end
eremat_ric3=eremat_ric3 + errmat_ric3;
P_ecric3=sum(eremat_ric3)/(K*N);
end
semilogy(SNR_db,P_ec,'-k','LineWidth',2); hold on;
semilogy(SNR_db,P_ecric1,'--k','LineWidth',3); hold on;
semilogy(SNR_db,P_ecric2,'-k','LineWidth',2); hold on;
semilogy(SNR_db,P_ecric3,':k','LineWidth',3); hold on;
set(gcf,'Color','w');set(gca,'FontSize',12); toc; grid on
xlabel('SNR (dB)','FontSize',14,'FontWeight','bold');
ylabel('BER','FontSize',14,'FontWeight','bold');
axis([min(SNR_db) max(SNR_db) min(10e-5) max(1)]);

```

## APPENDIX B

### 2/3 and 1/2 CONVOLUTIONAL CODED CDMA SYSTEM WITH DECORRELATOR DETECTOR IN AWGN AND RICIAN CHANNEL SIMULATION SOFTWARE

```
% The programme below tries to find the BER in CDMA environment 2/3 and 1/2
Convolutional code with formulation based on Proakis 1995 Ch.15., Eq. (15-3-35).
clear;clc;close all;tic
%%%%%%%%% Generating gold sequences %%%%%%%%%%
K=5; alt=1; Rs=[]; gkt=[]; stm1=[1 0]; tlen=10;
%T=31; genpo1='[5 2 0]';genpo2='[5 4 3 2 0]'; insta='[0 0 0 0 1]';
T=63; genpo1='[6 1 0]';genpo2='[6 5 2 1 0]'; insta='[0 0 0 0 0 1]';
%T=127; genpo1='[7 3 0]';genpo2='[7 3 2 1 0]'; insta='[0 0 0 0 0 0 1]';
%T=511; genpo1='[9 4 0]';genpo2='[9 6 4 3 0]'; insta='[0 0 0 0 0 0 0 0 1]';
seq_index=alt:K+alt-1;
set_param('goldgen/Gold Sequence Generator','genPoly1',genpo1)
set_param('goldgen/Gold Sequence Generator','genPoly2',genpo2)
set_param('goldgen/Gold Sequence Generator','iniState1',insta)
set_param('goldgen/Gold Sequence Generator','iniState2',insta)
for a=1:K
    set_param('goldgen/Gold Sequence Generator','index',num2str(seq_index(a)))
    sim('goldgen'); gkt=[gkt;pn(1:T)/sqrt(T)];
end
%%%%%%%%% Finding Crosscorrelation Coeffs.%%%%%%%%%
Rs=gkt*gkt';
%%%%%%%%% Message Signal %%%%%%%%%%
N=150000;
stpk=randint(K,N);
eremat=zeros(1,16); eremat_ric1=zeros(1,16); eremat_ric2=zeros(1,16);
eremat_ric3=zeros(1,16);
for m=1:500
    stp0=stpk(:,(m-1)*length(stpk)/500+1:m*length(stpk)/500);
    %%%%%%%%% Convolutional Encoding %%%%%%%%%
    stp0_conv=[];
    for b=1:K
        x_set=stp0(b,:);
        tre=poly2trellis(3,[5 7]); % Encoding trellis for a rate of 1/2
        %tre=poly2trellis([5 4],[23 35 0;0 5 13]);% Encoding trellis for a rate of 2/3
        x_set_tre=convenc(x_set,tre); % Convolutional encoding by a rate 1/2
```

```

stp0_conv=[stp0_conv; x_set_tre]; % +1, 0
end
stp=2*stp0_conv-1; % +1, -1
clear x_set_tre stp0_conv
%%%%%%%%%% Spreading Operation %%%%%%%%%%%
stm=[];
for c=1:K
    stc=[];
    for d=1:length(stp)
        std=stp(c,d).*gkt(c,:);
        stc=[stc std];
    end
stm=[stm;stc];
end
st=sum(stm,1);
stmT=stm(:,1:T);
k=[st',zeros(length(st'),1)];
st1=k(:,1)+j*k(:,2)+j*1e-10;
clear stc std stm k;
%%%%%%%%%% Passing via the channel %%%%%%%%%%%
Fd=1; Fs=1; KK1='100'; KK2='10'; KK3='1';
Tss='N*1e-0'; Fddc='5e-2'; Fddy='5e-2'; gdB='0'; dVec='0';
set_param('Ricchannelc/Rician Fading
Channel','K',KK1,'Fd',Fd,'Fddc','gainVecdB',gdB,'delayVec',dVec);
sim('Ricchannelc');
stric1=[real(stc)'];
set_param('Ricchannelc/Rician Fading Channel','K',KK2);
sim('Ricchannelc');
stric2=[real(stc)'];
set_param('Ricchannelc/Rician Fading Channel','K',KK3);
sim('Ricchannelc');
stric3=[real(stc)'];
clear st1 stc str Fd Fs Tss Fddc Fddy gdB dVec
%%%%%%%%%% Computation of Power, Energy Setting Noise Level %%%%%%%%%%%
P_st=sum(stmT.^2,2)/T;
E_st=P_st*T;
SNR_db=0:15;
dblen=length(SNR_db);
set_ntp=10*log10(repmat(E_st(1,:),dblen,1))-SNR_db'-3.0103*ones(dblen,1);
clear P_st E_st stmT
nt=[];
for s=1:dblen
nt1=wgn(1,length(st),set_ntp(s));
nt=[nt;nt1];
end
clear nt1
%%%%%%%%%% Received Signal %%%%%%%%%%%

```

```

rt=repmat(st,dblen,1)+nt; % signal with only noise added
rtric1=repmat(stric1,dblen,1)+nt;
rtric2=repmat(stric2,dblen,1)+nt;
rtric3=repmat(stric3,dblen,1)+nt;
clear nt st stric1 stric2 stric3
P_ec=[]; P_ecric1=[]; P_ecric2=[]; P_ecric3=[]; bk_det_arr=[];
bk_det_arr_ric1=[]; bk_det_arr_ric2=[]; bk_det_arr_ric3=[];
%%%%%%%%%% Symbol by Symbol Detection %%%%%%%%%%%
for nos=1:length(stp)
rk=rt(:,(nos-1)*T+1:nos*T)*gkt';
bk_det=sign(inv(Rs)*rk');
bk_det0=(bk_det + 1)/2;
bk_det_arr=cat(3, bk_det_arr, bk_det0);
end
[x,y,z]=size(bk_det_arr);
bk_det_x=[];
for ix=1:x
bk_det_y=[];
for iy=1:y
sig=bk_det_arr(ix,iy,:);
sig_a=sig(:,:);
bk_det_vitdec=vitdec(sig_a, tre, tblen,'trunc','hard'); % 1/2 Convolutional decoding
bk_det_y=[bk_det_y bk_det_vitdec];
end
bk_det_x=[bk_det_x; bk_det_y];
end
errmat=[];
for ns=1:dblen
bk_det_xk=bk_det_x(:,(ns-1)*length(stp0)+1:ns*length(stp0));
[number,ratio,loc]=symerr(stp0 , bk_det_xk);
loc_sum=sum(sum(loc));
errmat=[errmat, loc_sum];
end
eremat=eremat + errmat;
P_ec=eremat/(K*N)
clear rk rt bk_det bk_det0 bk_det_arr x y z bk_det_x bk_det_y
clear sig sig_a bk_det_vitdec bk_det_xk number ratio loc loc_sum
for nos=1:length(stp)
rkr1=rtric1(:,(nos-1)*T+1:nos*T)*gkt';
bk_detric1=sign(inv(Rs)*rkr1');
bk_det0ric1=(bk_detric1+1)/2;
bk_det_arr_ric1=cat(3, bk_det_arr_ric1, bk_det0ric1);
end
[x,y,z]=size(bk_det_arr_ric1);
bk_det_x_ric1=[];
for ix=1:x
bk_det_y_ric1=[];

```



```

for iy=1:y
sig_ri1=bk_det_arr_ri1(ix,iy,:);
sig_a_ri1=sig_ri1(:,:);
bk_det_vitdec_ri1=vitdec(sig_a_ri1,tre,tblen,'trunc','hard'); % Con. decoding
bk_det_y_ri1=[bk_det_y_ri1 bk_det_vitdec_ri1];
end
bk_det_x_ri1=[bk_det_x_ri1; bk_det_y_ri1];
end
errmat_ri1=[];
for ns=1:dblen
bk_det_xk_ri1=bk_det_x_ri1(:,(ns-1)*length(stp0)+1:ns*length(stp0));
[number_ri1,ratio_ri1,loc_ri1]=symerr(stp0, bk_det_xk_ri1);
loc_sum_ri1=sum(sum(loc_ri1));
errmat_ri1=[errmat_ri1, loc_sum_ri1];
end
ermat_ri1=ermat_ri1 + errmat_ri1;
P_ecric1=ermat_ri1/(K*N)
clear rkric1 rtric1 bk_detric1 bk_det0ric1 bk_det_arr_ri1 x y z bk_det_x_ri1
bk_det_y_ri1
clear sig_ri1 sig_a_ri1 bk_det_vitdec_ri1 bk_det_xk_ri1 number_ri1 ratio_ri1
loc_ri1 loc_sum_ri1
for nos=1:length(stp)
rkric2=rtric2(:,(nos-1)*T+1:nos*T)*gkt';
bk_detric2=sign(inv(Rs)*rkric2');
bk_det0ric2=(bk_detric2+1)/2;
bk_det_arr_ri2=cat(3, bk_det_arr_ri2, bk_det0ric2);
end
[x,y,z]=size(bk_det_arr_ri2);
bk_det_x_ri2=[];
for ix=1:x
bk_det_y_ri2=[];
for iy=1:y
sig_ri2=bk_det_arr_ri2(ix,iy,:);
sig_a_ri2=sig_ri2(:,:);
bk_det_vitdec_ri2=vitdec(sig_a_ri2,tre,tblen,'trunc','hard'); % Con. decoding
bk_det_y_ri2=[bk_det_y_ri2 bk_det_vitdec_ri2];
end
bk_det_x_ri2=[bk_det_x_ri2; bk_det_y_ri2];
end
errmat_ri2=[];
for ns=1:dblen
bk_det_xk_ri2=bk_det_x_ri2(:,(ns-1)*length(stp0)+1:ns*length(stp0));
[number_ri2,ratio_ri2,loc_ri2]=symerr(stp0, bk_det_xk_ri2);
loc_sum_ri2=sum(sum(loc_ri2));
errmat_ri2=[errmat_ri2, loc_sum_ri2];
end
ermat_ri2=ermat_ri2 + errmat_ri2;

```

```

P_ecric2=ermtat_ric2/(K*N)
clear rkric2 rtric2 bk_detric2 bk_det0ric2 bk_det_arr_ric2 x y z bk_det_x_ric2
bk_det_y_ric2
clear sig_ric2 sig_a_ric2 bk_det_vitdec_ric2 bk_det_xk_ric2 number_ric2 ratio_ric2
loc_ric2 loc_sum_ric2
for nos=1:length(stp)
rkric3=rtric3(:,(nos-1)*T+1:nos*T)*gkt';
bk_detric3=sign(inv(Rs)*rkric3');
bk_det0ric3=(bk_detric3+1)/2;
bk_det_arr_ric3=cat(3, bk_det_arr_ric3, bk_det0ric3);
end
[x,y,z]=size(bk_det_arr_ric3);
bk_det_x_ric3=[];
for ix=1:x
bk_det_y_ric3=[];
for iy=1:y
sig_ric3=bk_det_arr_ric3(ix,iy,:);
sig_a_ric3=sig_ric3(:,:);
bk_det_vitdec_ric3=vitdec(sig_a_ric3,tre,tblen,'trunc','hard'); % Con. decoding
bk_det_y_ric3=[bk_det_y_ric3 bk_det_vitdec_ric3];
end
bk_det_x_ric3=[bk_det_x_ric3; bk_det_y_ric3];
end
errmat_ric3=[];
for ns=1:dblen
bk_det_xk_ric3=bk_det_x_ric3(:,(ns-1)*length(stp0)+1:ns*length(stp0));
[number_ric3,ratio_ric3,loc_ric3]=symerr(stp0 , bk_det_xk_ric3);
loc_sum_ric3=sum(sum(loc_ric3));
errmat_ric3=[errmat_ric3, loc_sum_ric3];
end
ermtat_ric3=ermtat_ric3 + errmat_ric3;
P_ecric3=ermtat_ric3/(K*N)
clear rkric3 rtric3 bk_detric3 bk_det0ric3 bk_det_arr_ric3 x y z bk_det_x_ric3
bk_det_y_ric3
clear sig_ric3 sig_a_ric3 bk_det_vitdec_ric3 bk_det_xk_ric3 number_ric3 ratio_ric3
loc_ric3 loc_sum_ric3
end
semilogy(SNR_db,P_ec,'-k','LineWidth',2); hold on;
semilogy(SNR_db,P_ecric1,'--k','LineWidth',3); hold on;
semilogy(SNR_db,P_ecric2,'-k','LineWidth',2); hold on;
semilogy(SNR_db,P_ecric3,':k','LineWidth',3); hold on;
set(gcf,'Color','w');set(gca,'FontSize',12); toc; grid on
xlabel('SNR (dB)','FontSize',14,'FontWeight','bold');
ylabel(' BER','FontSize',14,'FontWeight','bold');
axis([min(SNR_db) max(SNR_db) min(10e-5) max(1)]);

```