INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

CONVOLUTIONAL RING CODES FOR FADING CHANNELS

by

RONALD W. KERR
M.Sc.(Eng.), Queen's University at Kingston, 1989
B.Sc.(Eng.), Queen's University at Kingston, 1987

A Dissertation Submitted in Partial Fulfillment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical and Computer Engineering

We accept this dissertation as conforming
to the required standard

Dr. V. K. Bhargava, Supervisor, Dept. of Elect. & Comp. Eng.

Dr. Q. Wang, Member, Dept. of Elect. & Comp. Eng.

Dr. P. Agathoklis, Member, Dept. of Elect. & Comp. Eng.

Dr. N. Horspool, Outside Member, Dept. of Computer Science

Dr. T. A. Gulliver, External Examiner
University of Canterbury

Supervisor: Dr. V.K. Bhargava

ABSTRACT

Rate 1/2 systematic recursive convolutional codes over integer rings modulo-q are investigated for their performance. The investigation examines the performance in severe fading and additive white Gaussian noise for codes with various constraint lengths. The arithmetic for the codes is modulo-q, where the value of q is within the range of 2 to 16. An exhaustive search is carried out for codes with short constraint lengths. A reduced search is developed for larger constraint lengths which restricts the tap polynomials to irreducible polynomials over $\mathbb{Z}_q$. The irreducible polynomials are generated and the ones not found in the literature are presented in tables. The search algorithms are outlined and the results for the codes are tabulated.

The performance of selected codes are verified by Monte-Carlo simulation techniques. Several codes have better performance than comparable codes presented in the literature for the Rayleigh fading channel. In sme of cases, the codes found have better performance on the AWGN channel than the best known ring codes.

The characteristics of rotationally invariant (RI) ring codes presented in the literature are used in an exhaustive search for codes over $\mathbb{Z}_q$ which are invariant to phase shifts of $2\pi/q$. Tables of RI codes optimized for the Rayleigh fading channel are presented along with codes which are optimized for the AWGN channel.

Examiners:

Dr. V. K. Bhargava, Supervisor, Dept. of Elect. & Comp. Eng.

Dr. Q. Wang, Member, Dept. of Elect. & Comp. Eng.

Dr. P. Agathoklis, Member, Dept. of Elect. & Comp. Eng.

Dr. N. Horspool, Outside Member, Dept. of Computer Science

Dr. T. A. Gulliver, External Examiner
University of Canterbury

Table of Contents

List of Figures

List of Tables

Notation

d^2_{prod}	squared product distance.
d^2_{free}	minimum squared free distance.
n_{free}	number of paths with distance d^2_{free}
n_{prod}	number of paths with distance d^2_{prod}
n_e	average number of errors paths with d^2_{free}
n_p	average number of errors paths with d^2_{prod}
l_{eff}	effective length of a code
R_c	code rate
$\prod$	product
$\sum$	sum
$\mathbb{Z}_q$	ring of integers modulo-q
$\dot{+}$	addition modulo-q
$\dot{-}$	subtraction modulo-q
$\dot{\times}$	multiplication modulo-q
g_∞	asymtotic coding gain
$\neq$	not equal to
$\|(\bullet)\|$	Euclidean magnitude
$\gcd(x, y)$	greatest common divisor of x and y
$\mathrm{lcm}(x, y)$	least common multiple of x and y
$\deg(f)$	degree of $f(x)$
$f(D)$	polynomial in D $f_s D^s + f_{s-1} D^{s-1} + \ldots + f_0$
$\forall$	for all
$\in$	element of
E_b	Energy per bit
E_s	Energy per channel symbol
ρ	fading amplitude of the channel

Acknowledgement

I would like to thank the following people:

- My supervisor Dr. V.K. Bhargava, for the research infrastructure, funding, friendship, encouragement and guidance.
- Dr. T.A. Gulliver for his advice and many valuable discussions throughout my program.
- Dr. Ivan Fair for many things, but especially for his question about non-field linear feedback shift registers.
- Dr. Q. Wang for his many helpful discussions.
- the students in the telecommunication lab for their friendship and valuable discussions. I would especially like to thank Joe Mueller for allowing me to use and modify some of his channel model software for my work and Roman Pichna for writing miscellaneous scripts that made my search easier.
- Dr. Ian Blake and Thomas Mittelholzer for providing me with copies of some of their papers and manuscripts that were difficult to obtain through other sources.
- most of all Yoko for her patience, encouragement and *aisaibento*.

Dedication

To Yoko

Chapter 1

Introduction

In 1948, Claude Shannon published "A Mathematical Theory of Communications" [1] in which he demonstrated that reliable communication could be achieved over a noisy channel. Proper encoding of the information and a transmission rate less than the channel capacity are required to achieve reliable communication of information. Absent from his work was a method for constructing good codes. This started an area of research to find good codes and methods to construct these codes. The problem is to find an encoding/decoding strategy which adds the minimum amount of redundancy, and can meet the error performance and delay requirements of the system.

Traditionally, coding theorists have developed Forward Error Correcting (FEC) codes with a lot of structure, which lends itself to efficient decoding strategies. In many cases, the arithmetic for these codes is over an algebraic structure known as a Galois field [50]. A Galois field is the basis for several well-known codes such as the binary BCH codes [38] and Reed-Solomon codes [36]. In these cases, a number of information symbols enter the encoder and redundant symbols are added to form a codeword. The codeword is transmitted and the receiver decodes the codeword independent of previous or future codewords. This type of coding is called block coding.

Another type of coding is convolutional coding. This type uses a finite state machine which adds memory into the information sequence to form a coded output sequence. Each state has a defined output for a specific input symbol. The current state is dependent on all previous inputs and the starting state of the encoder. The coded output sequence is transmitted and the receiver observes the noisy coded sequence and uses the input-output relationship of the encoder to estimate the infor-

mation sequence.

In both cases, the distance spectrum between codewords for block codes, or coded sequences for convolutional codes, will define the performance of the code. Here, distance spectrum means the set of all possible distances between codewords or sequences. The choice of the distance metric depends on the channel that the system is transmitting on and the code used. For block codes, a common distance metric is the Hamming distance between codewords. For convolutional codes operating over the additive white Gaussian noise (AWGN) channel and utilizing a soft-decision decoder, the metric is the Euclidean distance between the code sequences. If a hard-decision decoder is used then the Hamming distance is used. The symbol or bit error performance of the codes at high signal-to-noise ratios (SNR) is determined mainly by the minimum distance. However, at lower SNR other terms in the distance spectrum will contribute to the error rate of the system.

On a mobile radio channel, the user's signal is affected by multi-path fading where the received signal power fluctuates due to additive and destructive interference from multiple delayed copies of the transmitted signal. In severe fading, it has been shown for trellis coded modulation (TCM) that the product distance is the correct metric to maximize along with the symbol distance [20]. In TCM systems, the signal set is expanded. The signal set expansion provides the redundancy necessary for coding [17]. The symbol distance is the number of symbols along an error path which differ from the correct symbols.

Ring codes utilize arithmetic over an integer ring. This can be applied to both block and convolutional codes. This research seeks to find good convolutional ring codes which can operate in the fading environments of both the mobile satellite and mobile terrestrial channels. The terrestrial cellular channel is subjected to Rayleigh fading since no direct path between the mobile and the base station usually exists. The mobile satellite channel is subjected to shadowing and fading [55]. For Low Earth Orbit (LEO) and Medium Earth Orbit (MEO) satellites, the channel statistics can change rapidly as the satellite moves with respect to the user [52]. As the channel can change quickly, it is necessary for the codes to perform well in severe fading as well as on the AWGN channel. To satisfy these requirements, we first maximize the effective length and the squared product distances of the codes and then from this

set of codes we find the codes with maximum squared Euclidean distance.

1.1 Related Results

Boztaş et al. [40] introduced two families of 4-phase spreading sequences. (One of the families had first been discovered by P. Solé [41] unbeknownst to Boztaş *et al.*). These families have lower cross-correlation value than binary sequences. This fact is interesting because: the sequences were designed using rings rather than Galois Fields, and they are formed with a linear feedback shift register using modulo-4 arithmetic [43]. The sequences prior to mapping onto a QPSK modulation forms a linear cyclic quaternary code. As the sequences have low correlation values they also have large minimum Euclidean distance indicating a potential for use in error-correction coding [45].

Hammons et al. [44] showed that several non-linear cyclic binary codes, such as Nordstrom-Robinson, Kerdock, and Preparata codes, are linear in $\mathbb{Z}_4$ (the integers mod4). The modification from the classical theory is to view the codes as ideals in polynomial rings over a ring of integers modulo 4 rather than over finite fields. This is of interest as linear codes are easier to decode than non-linear codes.

Given Hammons et al. [44] results in cyclic coding, and working in rings, the question naturally arises if good codes can be found by working in integer rings modulo-q for an arbitrary q and mapping the q-ary symbols naturally onto a q-PSK modulation set. There are several benefits in using these codes. One benefit is that no special circuitry is required in modern computers and digital signal processors as they are well suited to modulo-q arithmetic. Also, a ring code requires no set partitioning as does trellis coded modulation [3]. Ring codes can be made systematic, i.e., separating the information symbols from the parity symbols which is impossible when using trellis coded modulation [17].

Although this dissertation is concerned with convolutional codes, there are several papers of interest for block ring codes. Shankar [46] presented BCH codes over arbitrary integer rings. In papers by Blake [51, 13] ring analogs of Hamming, Reed-Solomon and BCH codes were presented for certain rings. He suggested that a q-ary communication channel might be better for computers to communicate than a binary

one. When computers are capable of q-ary logic, utilizing a q-ary code would seem to be a natural choice. A good introduction on block coded modulation over integer rings is presented in Baldini and Farrell [2]. The rings presented were $\mathbb{Z}_4$, $\mathbb{Z}_8$ and $\mathbb{Z}_{16}$. In Chen [28], 6 PSK ring codes are made up from a linear binary and a linear ternary code.

For convolutional codes, there have been many recent papers using convolutional coding over integer rings. Ring coded Continuous Phase Modulation (CPM) was considered in Rimoldi and Li [14] and Yang and Taylor [15]. They both used an encoder and a continuous phase modulator with the same arithmetic, thus eliminating a binary to M-ary mapping. Yang and Taylor worked with Continuous Phase Frequency Shift Keying (CPFSK) and obtained significant coding gains over previous work in coded CPFSK. Rimoldi and Li compared their results with binary codes with the same complexity and found that using codes over rings can improve the performance of many coded CPM systems [14]. They also found it was beneficial to feedback the state information from the modulator to the encoder [14]. By utilizing this feedback, they were able to achieve a small coding gain over a non-feedback coded system. Karam et al. [16] used trellis coded CPFSK over rings for quaternary and octary modulations and showed that ring-coded CPFSK techniques outperform previously known coding approaches.

In Baldini and Farrel [3] coded modulation using convolutional codes was presented. They searched for systematic rate 1/2, 2/3 and 3/4 codes for 4, 8, and 16 PSK, respectively. Massey et al. [8] also considered systematic convolutional codes over $\mathbb{Z}_q$. In fact both of these papers choose q to be a power of 2. The choice of a power of 2 allows for an easy mapping from a binary information source onto a ring (i.e., an integer number of bits define the ring symbol). Massey et al. presented [8] convolutional codes for $\mathbb{Z}_8$ and $\mathbb{Z}_{16}$.

Baldini and Farrel [3] and Mittelholzer [6], searched for rotationally invariant codes. These are beneficial as no absolute phase reference is necessary at the receiver allowing for a less complex receiver and a large number of errors are avoided if the channel has slow phase rotations.

Baldini and Farrel in [2, 3] worked on coded modulation using ring codes. Trellis Coded Modulation (TCM), proposed by Ungerboeck [17], showed that the system

could achieve a significant coding gain by expanding the number of modulation signals and designing the code considering the actual channel and modulation to be used. The development of TCM in [17, 18, 19] considered the Gaussian channel. Divsalar and Simon [20, 21] showed that the performance criteria for the code design are different on fading channels. In fading channels, the product trellis and the minimum effective length [20] dominate the asymptotic performance of the code. Using these criteria, TCM codes for fading channels have been found or constructed. A good overview of the work done on coded modulation for fading channels can be found in Jamali and Le-Ngoc [32].

1.2 Contributions

In this dissertation, we present the results of an exhaustive and reduced search for ring codes suitable for severe fading environments. The codes and their characteristics are presented.

Based on previous work from Baldini and Farrell [3] and works from Massey and Mittelholzer et al. [5]-[11], for good codes on the AWGN channel, we search for systematic recursive convolutional codes which perform well in a fading environment. The codes with maximum effective length and squared product distance to achieve good performance in Rayleigh fading are found. From this set, we then search for codes with maximum squared Euclidean distance, as this affects the performance on the AWGN channel and is a factor at low signal-to-noise ratios.

Also, codes optimized for the AWGN channel are included for means of comparison with the performance of the codes designed for fading. The AWGN codes perform well on the AWGN channel and in some cases better than the fading codes at slow signal-to-noise ratio.

An exhaustive search is carried out over codes with short constraint length. A reduced search is then carried out over longer constraint lengths. Irreducible polynomials over $\mathbb{Z}_q$ where q is a non-prime number were found and used in the reduced search for codes suitable for fading.

The polynomials may be useful for the development of cyclic codes over $\mathbb{Z}_q$ and are presented in Appendix A.

1.3 Thesis Outline

In Chapter 2, fundamentals of rings and an introduction to the digital communication system are presented.

Chapter 3 presents the exhaustive search algorithm and tables of the codes found and their characteristics. A reduced search is necessary as the time required for an exhaustive search becomes quickly impractical for longer constraint lengths and larger values of q. The development of the reduced search is described and the codes found are presented. The exhaustive and reduced search algorithms were designed to search for codes suitable for fading environments; however, they were also designed to find codes which were optimal on the AWGN channel. These codes are also presented in Chapter 3.

In Chapter 4, the simulation model for the system is presented. The coded system is simulated to verify the performance on the Rayleigh fading channel. The performance of the codes found here are compared with known codes from the literature. Also, we compare the performance difference between the codes designed for the fading channel and codes designed for the AWGN channel.

In Chapter 5, the results of a search for rotationally invariant codes are presented. The codes are compared with codes from literature as well as codes found in Chapter 3.

Chapter 6 contains concluding remarks and suggestions for future investigations based on the results presented in this dissertation.

Chapter 2

Fundamentals

2.1 Introduction

This chapter will introduce convolutional ring coding, the system diagram of the digital communication system and some of the distance measures. In Section 2.2, we introduce some basic definitions and properties of rings which will be used in later chapters. In Section 2.3, a system description and an overview of how the ring codes will fit into the system is given. In Section 2.4 we introduce convolutional ring codes for MPSK modulation. The distance measures which define the performance over Gaussian and Fading channels are presented in Section 2.6.

2.2 Rings

A *ring* is an algebraic structure consisting of a set of elements R and two binary operations: addition and multiplication, such that for all elements a, b and c in R

- addition is associative, i.e. $(a+b)+c = a+(b+c)$.
- addition is communitative, i.e. $a+b = b+a$.
- there is an element r_0 in R, called the additive identity, with the property that $r_0 + a = a + r_0 = a$ for all a in R.
- each element a in R has an additive inverse, $-a$, such that $a+(-a) = -a+a = r_0$.
- multiplication is associative.
- multiplication is distributive over addition, i.e. $a(b+c) = ab+ac$ and $(a+b)c = ac+bc$.

Addition of an additive inverse is called subtraction.

An element, a, in an arbitrary ring R, is called a *unit* if there exists another element b, such that $ab = ba = 1$. The element b is called the multiplicative inverse of a and can be written a^{-1}. An element $a \in R$ is called a *zero divisor* if there exists another non-zero element $b \neq 0$ such that $ab = 0$ or $ba = 0$ [34].

In $\mathbb{Z}_q$, an element is a *unit* only if it is relatively prime to the modulus q. When q is a prime number, all elements are units and have multiplicative inverses and therefore division is defined. Then $\mathbb{Z}_q$ is a *field* [34].

As an example of rings , consider the ring of integers $\mathbb{Z}$. For any two elements: a and b, we see that $ab, a + b \in \mathbb{Z}$. Now let us denote $\mathbb{Z}_q$ as the *integer residue ring*. It consists of the set $\{0, 1, \ldots, q-1\}$ and all addition and multiplication is carried out modulo-q. Any integer c can be represented by $r + tq$, and as the arithmetic is modulo-q, it is said that c is *congruent* to r, or

$$c \equiv r \bmod q. \tag{2.1}$$

We now consider the ring of polynomials denoted as $R[x]$ with coefficients in R. R is a finite commutative ring with multiplicative identity 1. The *leading coefficient (trailing coefficient)* of a non-zero polynomial is the coefficient of the largest (smallest) power of x whose coefficient is non-zero. If a(x) and b(x) are polynomials and if the leading coefficient of b(x) is a unit in R, then there exist unique polynomials q(x) and r(x) such that a(x)=q(x)b(x)+r(x) and deg[$r(x)$] < deg[$b(x)$] [9].

For the ratio of polynomials $a(x)/b(x)$ to be a rational function, the trailing coefficient of $b(x)$ must be a unit in R [9].

2.3 System Description

The system block diagram is shown in Fig. 2.1. The input to the convolutional encoder is a sequence of q-ary symbols. The encoder convolutionally encodes at a rate of k/n where k and n are the number of input and output symbols, respectively, per encoding interval. The n encoded symbols are mapped onto an MPSK signal set. Here, there is a natural labelling of the signal points of MPSK by elements of the ring, $0, 1, 2, \ldots, q-1$. In other words, the q-ary symbol l corresponds to the MPSK

signal, s_l by

$$s_l = \exp\left(\frac{i2\pi l}{q}\right). \tag{2.2}$$

where $i = \sqrt{-1}$. Note, that q must be equal to the number of signal points in the constellation (i.e., $q = M$). The sequence of coded symbols can be represented by

$$x = x_{01}x_{02}\ldots x_{0n}.x_{11}x_{12}\ldots x_{1n}\ldots x_{N1}x_{N2}\ldots x_{Nn}\ldots \quad . \tag{2.3}$$

The encoded sequence x is then interleaved and transmitted over a channel. In the channel, the sequence is subjected to fading and AWGN. The receiver deinterleaves the encoded symbol sequence and decodes the information by applying a soft-decision Viterbi decoding algorithm [35].

The Viterbi algorithm combined with convolutional codes works well only when the channel errors are independent. However, on a mobile fading channel, amplitude fades will produce bursts of channel errors. Interleaving is used to scramble the order of the coded sequence before transmitting it over the channel. After deinterleaving (descrambling) a burst of channel errors is broken into several smaller bursts or ideally independent errors. There are several methods of interleaving: block, convolutional and pseudo-random [38, 39]. Throughout this document, we will consider the case that the interleaving is ideal and that the channel is memoryless. This results in the fading amplitudes being independent between symbol intervals. We assume that the fading is slow enough to be considered constant over one channel symbol interval.

2.4 Convolutional coding on $\mathbb{Z}_q$

This section introduces the rate 1/2 convolutional encoder over $\mathbb{Z}_q$, that is studied in this dissertation.

Following the nomenclature presented in [3], the codes are defined as G(D) where,

$$G(D) = [\ 1 \quad g(D)/f(D)\ ]. \tag{2.4}$$

Here $g(D) = g_s D^s + g_{s-1} D^{s-1} + \ldots g_1 D + g_0$ and $f(D) = f_s D^s + f_{s-1} D^{s-1} + \ldots f_1 D + f_0$ define the feedforward and feedback taps as shown in Figure 2.2. The boxes are delay elements that delay a q-ary symbol by one clock cycle. The circles with a coefficient inside denote modulo-q multiplication and the $\dot{-}$ elements denote modulo-q

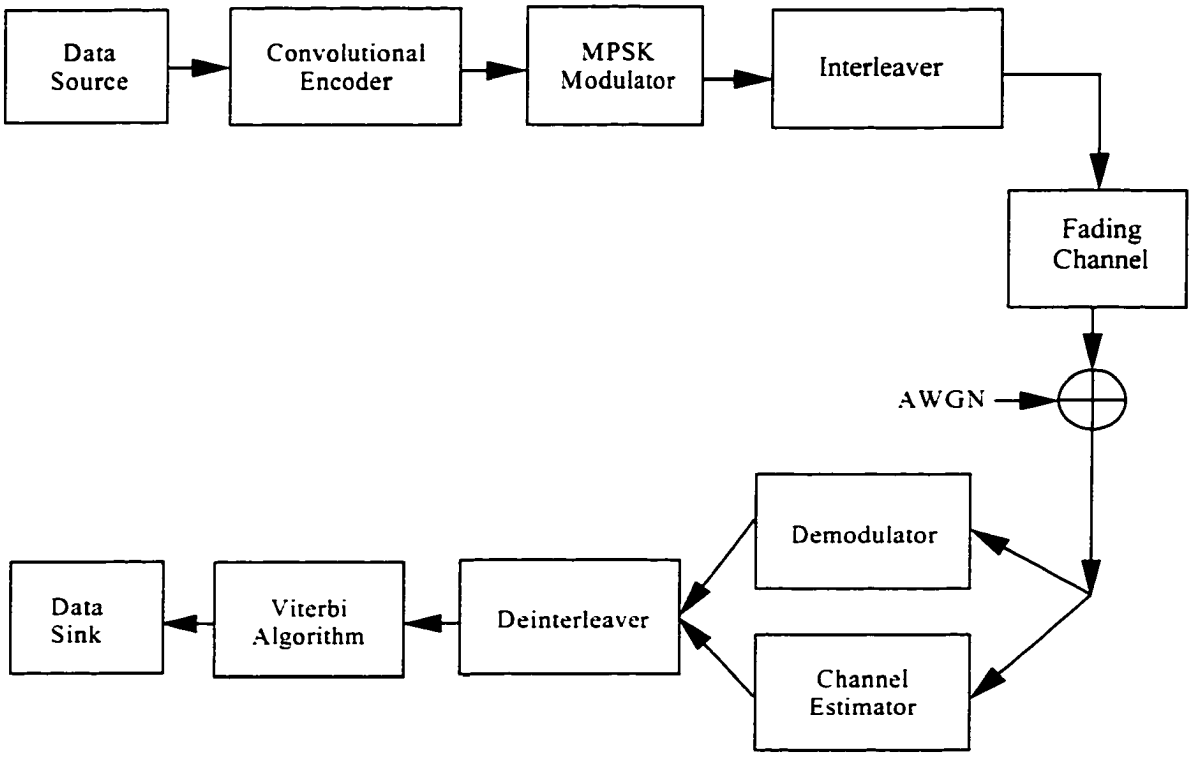


Figure 2.1. *System Model*

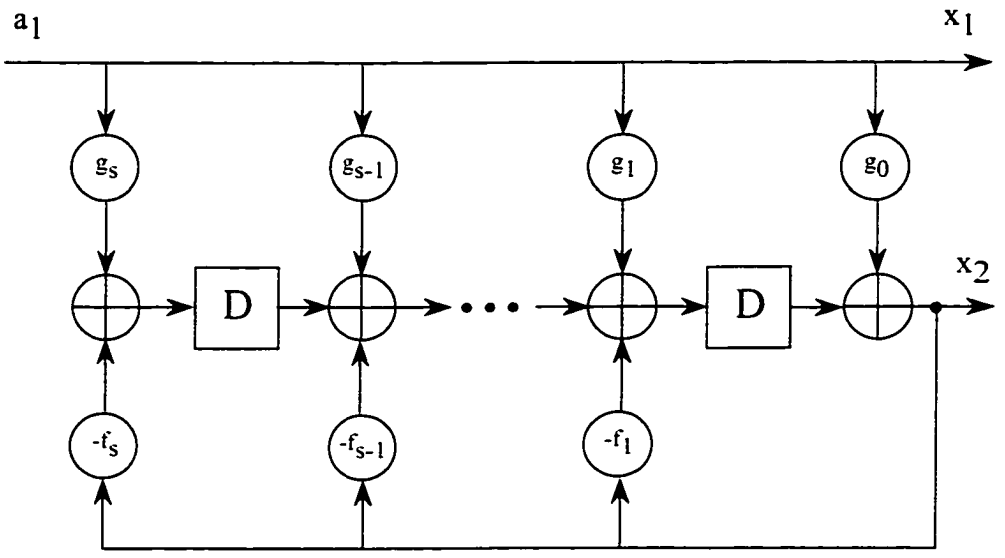


Figure 2.2. *Block Diagram of the encoder defined by G(D) of Eq. 2.4*

addition. The codes will be defined throughout this dissertation by the tap polynomial coefficients from G(D). (i.e., $g_s g_{s-1} \ldots g_0 / f_s f_{s-1} \ldots f_0$).

As an example, consider the code defined by $g(D) = D + 1$ and $f(D) = 2D + 1$ over $\mathbb{Z}_3$ (11/21 code). The generator matrix is defined as:

$$G(D) = [\ 1 \quad \tfrac{1D+1}{2D+1}\ ]. \tag{2.5}$$

and the encoder is shown in Figure 2.3. The input, output and state transition information for this encoder are presented in Table 2.1.

The codes examined here are *systematic*, which means that the information symbol appears unaltered in the output of the encoder. Systematic codes are considered to eliminate the possibility of finding a *catastrophic* code. A *catastrophic* code has error paths that are a finite distance from the correct path but have an infinite number of errors. With this characteristic a small amount of noise can cause the decoder to lose the correct path and never recover and thus, continue to output incorrect symbols.

2.5 MPSK signal set

The MPSK signal constellation is shown in Fig. 2.4. A benefit in the code design using this modulation scheme is the convenient mapping between the squared Euclidean

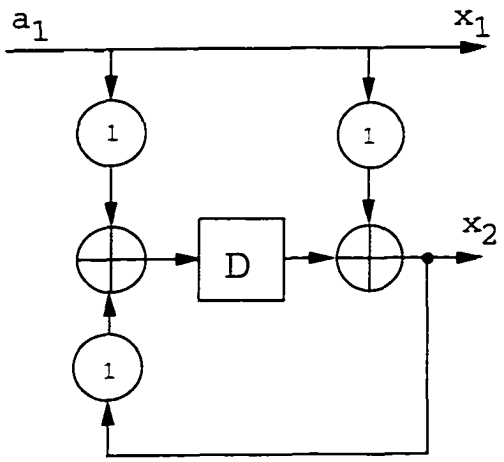


Figure 2.3. *Example of the 11/21 encoder over* $\mathbb{Z}_3$

Table 2.1. *State description of the 11/21 encoder over* $\mathbb{Z}_3$

Input	$State_t$	$State_{t+1}$	Output(x_1, x_2)
0	0	0	0 0
1	0	1	1 1
2	0	2	2 2
0	1	1	0 1
1	1	2	1 2
2	1	0	2 0
0	2	2	0 2
1	2	0	1 0
2	2	1	2 1

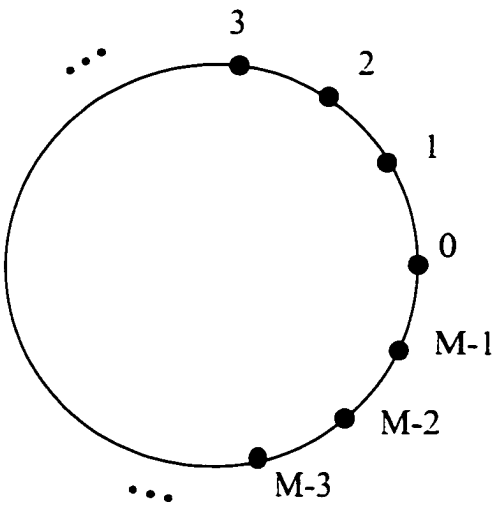


Figure 2.4. *MPSK Constellation*

distance and the modulo-q difference of their squared Euclidean weight [6]. Consider two q-ary symbols a and b, which are mapped onto an MPSK signal set by Equation 2.2. The squared Euclidean distance between the two signal points is given by $d^2(a,b)$ where

$$\begin{aligned} d_e^2(a,b) &= \left|\exp(\frac{2\pi ia}{q}) - \exp(\frac{2\pi ib}{q})\right|^2 \\ &= \left|1 - \exp(\frac{-2\pi ia}{q})\exp(\frac{2\pi ib}{q})\right|^2 \\ &= \left|1 - \exp(\frac{2\pi(b \dot{-} a)}{q})\right|^2 \\ &= d_e^2(0, b \dot{-} a) \\ &= w_E^2(b \dot{-} a). \end{aligned} \tag{2.6}$$

Here, $w_E^2(\cdot)$ is the squared Euclidean weight of a symbol.

2.6 Performance Estimates

In this section, we use performance estimates from the literature to introduce the parameters that affect a code's performance on the AWGN and fading channels. In Section 2.6.1, the uncoded performance of the modulation is given for the AWGN and Rayleigh fading channel. These formulas will be used to determine the coding gain achieved by the coded system over the uncoded system. In Section 2.6.2, squared free distance d^2_{free}, and the asymptotic gain of the coded system over an uncoded system, are introduced. In Section 2.6.3, the development of the performance estimate of the coded system on the Rician fading channel is shown. The definition of the parameters which affect the performance on the fading channel are presented.

2.6.1 Uncoded Performance

The symbol error rate of MPSK on the AWGN channel with a amplitude gain of ρ is approximated for high signal-to-noise ratios by [32]

$$P_s \approx 2Q\left[\rho\sqrt{\frac{2E_s}{N_0}}\sin(\frac{\pi}{M})\right] \tag{2.7}$$

where the energy per symbol is $E_s = \log_2 M R_c E_b$. E_b is the energy per bit, M is the number of modulation signals in an M-ary PSK modulation. R_c is the rate of the encoder used, and

$$Q(x) = \frac{1}{2\pi}\int_x^\infty \exp\left(\frac{t^2}{2}\right)dt. \tag{2.8}$$

Equation 2.7 can be written in terms of E_b/N_0 as

$$P_{sM}(\rho) \approx 2Q\left[\rho\sqrt{\frac{2m R_c \gamma_b}{N_0}}\sin(\frac{\pi}{M})\right]. \tag{2.9}$$

where $m = \log_2 M$ and $\gamma_b = E_b/N_0$. In Rayleigh fading, the uncoded symbol error rate can be calculated by averaging the conditional probability with respect to the fading amplitude ρ [32] over the Rayleigh density function.

The density function of Rayleigh fading is

$$p_\rho(\rho) = \frac{\rho}{\sigma_\rho^2}\exp\left(-\frac{\rho^2}{2\sigma_\rho^2}\right) \tag{2.10}$$

here, for the normalized case we set $E[\rho^2] = 1$, which results in $\sigma_\rho^2 = 1/2$.

After averaging Equation 2.9 over the normalized Rayleigh fading, the symbol error rate is given by [32].

$$P_{s,M} = 1 - \sqrt{\frac{m R_c \gamma_b \sin^2(\pi/M)}{1 + m R_c \gamma_b \sin^2(\pi/M)}}. \tag{2.11}$$

This equation was used to calculate the uncoded performance for MPSK modulation to obtain the coding gain of the coded system. The coding gain is the the difference in E_b/N_0, between the coded and uncoded (or reference) systems, required to achieve a given symbol error rate (SER) or bit error rate (BER).

2.6.2 Performance in AWGN

An error event occurs when the transmitter sends the symbol sequence, x and the receiver decides $\hat{x}$ where $x \neq \hat{x}$. The minimum error event is defined as the error event with the smallest distance. At high SNR, the performance of the code can be approximated by the distance of the minimum error event. In AWGN, the appropriate distance measure to maximize is the Euclidean distance. The squared Euclidean distance of the minimum error event is defined as

$$d_{free}^2 = \sum_{n \in \eta} |x_n - \hat{x}_n|^2 \tag{2.12}$$

where the set η is the set of all n along the minimum error path such that $x_n \neq \hat{x}_n$.

The asymptotic gain in dB on an AWGN channel is given by [3]

$$g_\infty = 10 \log_{10} \left[\frac{\log M_c}{\log M_u} R_c \frac{d_{free,c}^2}{d_{free,u}^2} \right] \tag{2.13}$$

where R_c is the coding rate, M_c, M_u and $d_{free,c}^2$, $d_{free,u}^2$ are the number of modulation signals and the d_{free}^2 of the coded and reference system, respectively.

2.6.3 Performance Estimate for the Rician Channel

Maximizing d_{free}^2 is effective on the AWGN channel. However, in fading, the product distance and the effective length of the code (also known as the minimum symbol

distance) are important parameters [20]. Here, we include sections of the development in [20], to develop the code search criteria for codes over fading channels.

Consider the M-ary convolutional encoder shown in Fig. 2.1. The signal received at the decoder is

$$r_i = \rho_i s_i + \eta_i. \tag{2.14}$$

where s_i is the MPSK signal, ρ_i is the amplitude of the fading process and η_i is additive white Gaussian noise with spectral density of $N_0/2$.

For Rician channels, the density function of ρ is denoted by

$$p_\rho(\rho) = 2\rho(1 + K)\exp\left[-(K + \rho^2(1 + K))I_0(\sqrt{K(1 + K)})\right]. \tag{2.15}$$

where K is the ratio of energy of the direct component to the energy in the diffused multipath component. $I_0(\cdot)$ is the zero-th order modified Bessel function of the first kind, i.e.,

$$I_0(x) = \frac{1}{2\pi}\int_0^{2\pi} \exp(x\cos t)dt. \tag{2.16}$$

The decoder's metric after observing l coded symbols is

$$m(r_l, s_l; \rho_l) = \sum_{i=1}^{l} \ln p_N(r_i|s_i, \hat{\rho}_i). \tag{2.17}$$

where $\hat{\rho}_i$ is the estimate of the fading amplitude at time i. The decoder will make an error if it decides $\hat{\mathbf{s}}_l = (\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_l)$ when $\mathbf{s}_l = (s_1, s_2, \ldots, s_l)$ was sent. The probability of this occurring is known as the the pairwise error probability and this is denoted as $P_2(s_l, \hat{s}_l)$.

For the case with ideal CSI where the fading amplitude estimates are always correct (i.e., $\hat{\rho}_l = \rho_l$), the metric can be expressed as

$$m(r_i, s_i; \rho_i) = -|r_i - \rho_i s_i|^2 \tag{2.18}$$

The decoder incorrectly decides $\hat{s}_l$ if

$$m(r_l, \hat{s}_l; \rho_l) \geq m(r_l, s_l; \rho_l). \tag{2.19}$$

The pairwise error probability is given by

$$P_2(s_l, \hat{s}_l) = \sum_{\rho_l}[P_2(s_l, \hat{s}_l|\rho_l)]. \tag{2.20}$$

where

$$P_2(s_l, \hat{s}_l|\rho_l) = P[m(r_l, \hat{s}_l; \rho_l) \geq m(r_i, s_i; \rho_i)]. \tag{2.21}$$

is the conditional pairwise error probability conditioned on the fading amplitude ρ_i.

Using the Chernoff bound technique the conditioned pairwise error probability averaged over the Rician density function is given by [20]

$$P_2(s_l, \hat{s}_l) \leq \prod_{i=1}^{l} \frac{1+K}{1+K+\frac{1}{4N_0}|s_i - \hat{s}_i|^2} \exp\left[\frac{-K\frac{1}{4N_0}|s_i - \hat{s}_i|^2}{1+K+\frac{1}{4N_0}|s_i - \hat{s}_i|^2}\right] \tag{2.22}$$

At high SNR, Equation 2.22 can be simplified to

$$P_2(s_L, \hat{s}_L) \leq \frac{((1+K)\epsilon^{-K})^L}{(\frac{1}{4N_0})^L d_p^2(L)} \tag{2.23}$$

where $d_p^2 = \prod_{i\in\eta} |s_i - \hat{s}_i|^2$. L is l_{eff}. η is the error event with the minimum number of symbols which differ from the correct path. $L = l_{eff}$ is the effective length of the code and is defined as the minimum number of differing symbols along any error path.

We introduce the notation of squared product distance and define it for two symbol sequences as

$$d_{prod}^2(x, \hat{x}) = \prod_{n\in\eta} |x_n - \hat{x}_n|^2 \tag{2.24}$$

where the set η is the set of all n where $x_n \neq \hat{x}_n$.

Using the pairwise error probability developed in Equation 2.23, we can upper bound the error probability on the Rician fading channel for high SNR as

$$P_e \leq \sum_{l_\eta} \sum_{d_{prod}^2(l_\eta)} \alpha((l_\eta), d_{prod}^2(l_\eta)) \frac{(1+K)\epsilon^{-K})^{l_\eta}}{(\frac{1}{4N_0})^{l_\eta} d_{prod}^2(l_\eta)}. \tag{2.25}$$

where $\alpha((l_\eta), d_{prod}^2(l_\eta))$ is the average number of code sequences having effective length l_η and squared product distance d_{prod}^2 and K is the ratio of power in the direct path to the power in the multipath [32]. The asymptotic performance of the code is approximated by the term with the smallest d_{prod}^2 and l_{eff}.

$$P_e \approx \alpha((L), d_{prod}^2(L)) \frac{((1+K)\epsilon^{-K})^L}{(\frac{1}{4N_0})^L d_{prod}^2(L)}. \tag{2.26}$$

Note that error rate is proportional to

$$P_b \approx \left(\frac{1}{\bar{E}_s/N_0}\right)^L. \tag{2.27}$$

where $\bar{E}_s/N_0$ is the average signal-to-noise ratio (SNR), and $L = l_{eff}$ is the minimum number of differing symbols along an error path. In an uncoded system, $l_{eff} = 1$ is typical. Thus, large gains could be achieved by maximizing the l_{eff} of the code. This being the case, we seek to maximize d^2_{prod} and l_{eff} for the class of codes under investigation.

2.7 Summary

In this chapter, we introduced some of the fundamental definitions and performance criteria that will be used in later chapters. In Section 2.2, definitions for rings and arithmetic on rings were given. The overall digital communication system was presented in Section 2.3. Definition and an introduction to the class of codes to be considered were presented in Section 2.4. In Section 2.6, the distance measures, performance estimates and the parameters that affect the codes' performance on fading and AWGN channels were presented.

Chapter 3

Some Rate 1/2 Convolutional Ring Codes

3.1 Introduction

In this chapter, the search algorithms are described and results are presented. An exhaustive search was carried out for short codes. However, this search technique is intractable for large rings and longer constraint lengths as the number of states and paths, that must be considered, increase exponentially. A reduced search algorithm searches over a subset of the possible codes. The reduced search criteria were developed from theory as well as empirical data from the partial results of the exhaustive search.

Rate 1/2 systematic codes are considered here to eliminate the possibility of finding catastrophic codes as well as to reduce the number of candidate codes. Codes with $f_0 = 1$ in the feedback polynomial $f(x)$ are considered. This condition ensures that the encoder is rational and formal long division of $g(D)$ by $f(D)$ is allowed [3, 8]. Consideration is also restricted to codes with a fully reachable trellis and no parallel transitions. Fully reachable trellis means that it is possible to reach all possible states in the trellis. The fully reachable condition was chosen as it was felt that when trying to maximize the l_{eff} or symbol distance a trellis with more states would have a longer effective path. Parallel transitions within the trellis limit the effective length of the code [20, 22] and thus the search was restricted to codes without parallel transitions.

The search was carried out to find codes with the maximum d^2_{prod} and l_{eff}. From this set of codes, the codes with minimum number of paths and the minimum number of errors along those paths were selected. When multiple codes with identical

l_{eff} existed then additional terms in the product transfer function were used in the selection process. Following that selection, codes with maximum squared Euclidean distance, d^2_{free} were chosen with a minimum number of paths with that distance, n_{free} are selected.

As previously stated, we were looking for codes that would work well in a variety of fading environments. As such, we chose to maximize the codes' performance for Rayleigh fading and then select the best codes for the AWGN channel from these codes. At low SNR, the performance is dominated by noise rather than by fading, thus it is important to have good performance in AWGN as well as in fading.

3.2 Exhaustive Search

It should be noted that the search space was reduced by half as the code defined by $g(x)/f(x)$ had the same characteristics as $-g(x)/f(x)$. Unless otherwise specified, the tables present only $g(x)/f(x)$ as the calculation of $-g(x)/f(x)$ given this information is trivial. For example, in the first line of Table 3.1 in $\mathbb{Z}_3$ the code 11/21 is presented. Thus, $g(x) = x + 1$ and $-g(x) = -x - 1 = 2x + 2$, so 22/21 is also a good code.

In the tables the polynomials $g(x)$ and $f(x)$ are denoted as $g_s g_{s-1} \ldots g_0 / f_s f_{s-1} \ldots f_0$. The asymptotic gain on an AWGN channel in dB is given by Equation 2.13.

The pseudocode in Section 3.2.1 is included to show the general outline of the search algorithm. The code is written to illustrate the functioning of the algorithm only. The output of the search algorithm is the truncated Euclidean transfer function and the truncated product transfer function. Both transfer functions contain up to 20 terms and contain the following information: the squared distance measure, the number of paths and the number of information symbol errors along that path, and for the product transfer function, the number of coded symbols that differ from the correct path.

The search was carried out in this way because the transfer function was desired. This information can then be used to select the best codes based on the performance estimate. As the codes are linear, we assume, without loss of generality that the all-zero sequence is transmitted.

In the following sections, the pseudo-code of the search algorithm is presented

and the search results are tabulated. The tables present the first three terms of the product and Euclidean transfer function for the resulting codes. The asymptotic coding gain over BPSK on the AWGN is presented in the tables with the Euclidean transfer function to enable a comparison between the various codes.

3.2.1 Search Algorithm

A brief explanation of the search algorithm is presented here. For each encoder considered, the algorithm initially advances in the trellis along the path from input $(1,0,0,\ldots)$ until either it merges with the all-zero path or the maximum depth of the search. The depth-to-search is an input parameter into the algorithm and is set to ensure that the first terms of the transfer function are included in the truncated transfer functions.

For each advance, the Euclidean distance and the product distance are calculated for each node in the trellis along the search path. When a merger with the all-zero path occurs, the distance characteristics are recorded and the number of paths for the distances is updated. After a merger (or going to the maximum search depth) the algorithm backtracks along the path by one node. The input symbol is incremented from the previous path that was considered. If it is equal to q then the algorithm sets it to zero and backtracks one more node. The algorithm again advances using the current node's distance characteristics.

If the algorithm hits the maximum depth to be searched without merging, it will backtrack along the search path by one node and continue the search. It does not record the distance information in this case.

The search terminates when after the search path $(q-1, q-1, q-1, \ldots)$ as it will back track to the beginning of the trellis and then increment the symbol to q which is an invalid input when working with elements from $\mathbb{Z}_q$.

The details of the search are outlined in the pseudo-code below:

```
begin;
        input parameters of search:
        initialize memory:

Coder:  Is there another encoder to test?
        if NO {
                goto Finish:
        }else {
                set next encoder taps:
        }
        create trellis:
        Is trellis fully reachable?
        if NO {
                goto Coder:
        }

Init:   set max_depth from input:
        set pointer to 0:
        set symbols for first error path to 100000...:
```

```
Search: Is pointer between 0 and max_depth?
        if YES {
                Move forward to next state:
                if pointer = max_depth-1 and no merge next state {
                        move pointer back one node along path:
                        get next symbol on path:
                        goto Search:
                }
        Update distance measures at trellis node:
        Is there a merger with the correct path?
        if YES {
                Is pointer less than max_depth?
                if YES {
                        record the distances for the error event:
                        move pointer back one node along path:
                        get next symbol on path:
                        goto Check:
                } else {
                        Is pointer equal to max_depth?
                        if YES {
                          record the distances for the error event:
                          move pointer back two nodes along path:
                          get next symbol on path:
                          goto Check:
                        }
                }
        } else {
                Is pointer equal to max_depth?
                if YES {
                        move pointer back one node along path:
                        get next symbol on path:
                        goto Search:
                }
        }
```

```
Check: Are distances worse than the best code on record?
        if YES{
                terminate search on this code:
                goto Coder:
        } else {
                goto Search:
        }

Finish: save to disk summary results and truncated transfer function:
        end:
```

3.2.2 Unit Memory Codes

In the tables, truncated transfer functions are given for the codes that were found. In Tables 3.1 and 3.2, the first three terms in the product transfer function are presented. The terms n, d^2, ϵ, l are the number of paths with the squared product distance of d^2, the number of errors with that distance and the length (number of non-zero symbols) for those paths, respectively. The Euclidean transfer function is presented in Tables 3.3 and 3.4. The search depth through the trellis for the constraint length one codes is 7 information symbols.

In some cases, there are multiple codes presented for a given ring. The first code has the best asymptotic performance as the performance is dominated by the l_{eff} and the squared product distance of paths with l_{eff}, (i.e., $d^2_{prod}(l_{eff})$). However, at low SNR, other terms in the transfer function will contribute to the error performance. The second code, which is marked with an asterisk, takes into consideration up to ten terms in the product transfer function. As these codes are considered for low SNR performance, maximizing l_{eff} and $d^2_{prod}(l_{eff})$ is less important than minimizing the summation in Equation 2.25. These codes have better Euclidean distance properties than the fading codes. This is beneficial when operating in low SNR conditions, as the noise will affect the performance of the code [32].

Table 3.1. *Product distance profiles for unit memory codes for $\mathbb{Z}_2$ to $\mathbb{Z}_{11}$*

$\mathbb{Z}_q$	Codes	l_{eff}	n_1	d_1^2	ϵ_1	n_2	d_2^2	ϵ_2	l_2	n_3	d_3^2	ϵ_3	l_3
2	01/11.10/11	3	1	64	2	1	256	2.0	4	1	1024	2.0	5
3	11/21.12/11	4	2	81	2	4	243	2.5	5	8	729	3	6
4	11/21.13/21	3	1	64	1	4	32	1.5	4	1	256	2	4
5	12/21.13/31	4	4	25	2	4	34.549	2.5	5	4	90.451	2.5	5
	22/41.23/11												
6	11/21.15/41	3	1	64	1	2	3	2	4	2	81	2	4
6*	12/31.14/31	3	2	27	1	1	64	2	3	4	12	2	4
7	12/31.15/41	4	2	5.27	2	2	17.12	2	4	2	26.61	2	4
	23/11.24/61												
	32/11.32/21												
	35/51.35/61												
8	13/21.15/61	3	1	64	1	2	2.343	2	4	2	13.656	2	4
	12/31.16/51												
	23/51.25/31												
9	12/41.17/51	4	2	1.404	2	2	4.958	2	4	2	11.64	2	4
	42/71.47/21												
9*	24/31.25/61	3	2	27	1	2	2.548	1	4	2	3.83	1	4
10	14/31.16/71	3	1	64	1	2	1.38	2	4	2	3.618	2	4
	23/71.27/31												
11	53/41.58/71	4	2	2.807	2	2	3.324	2	4	2	4.814	2	4

Table 3.2. *Product distance profiles for unit memory codes for* $\mathbb{Z}_{12}$ *to* $\mathbb{Z}_{16}$

$\mathbb{Z}_q$	Codes	l_{eff}	n_1	d_1^2	ϵ_1	n_2	d_2^2	ϵ_2	l_2	n_3	d_3^2	ϵ_3	l_3
12	2 5/11 1.2 7/11 1 5 2 /11 1. 5 5/2 1 5 7 /10 1. 5 10/1 1	3	1	64	1	2	.268	2	4	2	3	2	4
12*	2 5/3 1.2 5/9 1 2 7 /3 1.2 7/91	3	2	27	1	1	64	2	3	4	2.00	2	4
13	24/51.63/81 23/71.64/21 29/81.6 10/5 1 6 9/11 1.2 10/6 1	4	2	1.372	2	2	1.875	2	4	2	2.114	2	4
13*	3 5/2 1.4 5/7 1 2 8/3 1.2 5/10 1 6 5/4 1.6 8/9 1 3 8/11 1.4 8/6 1	4	4	1.217	2	4	8.556	2	4	4	16.227	2	4
14	3 5/12 1.3 9/2 1 5 4/11 1.5 10/3 1 6 3/5 1.6 11/9 1	3	1	64	2	2	0.753	2	4	2	2.445	2	4
15	2 11/8 1.7 4/2 1 2 4/7 1 7 11/13 1	4	8	1.0	2	4	25	2	4	2	81	2	4
15*	2 9/9 1	2	2	9.0	2	2	1.307	2	4	2	1.49	2	4
16	3 9/14 1.37/21	3	1	64	1	2	0.4237	2	4	2	0.9489	2	4
16*	3 6/2 1. 3 6/14 1	2	1	16.0	2	2	0.376	2	4	2	0.842	2	4

Table 3.3. *Truncated transfer functions for unit memory code for codes presented in Table 3.1 for $\mathbb{Z}_2$ to $\mathbb{Z}_{11}$*

$\mathbb{Z}_q$	Codes	n_1	d_1^2	ϵ_1	n_2	d_2^2	ϵ_2	n_3	d_3^2	ϵ_3	g_∞ dB
2	01/11.10/11	1	12.00	2	1	16.00	2	1	20.00	2	1.76
3	11/21.12/11	2	12.00	2	4	15.00	2.5	8	18.00	3	3.76
4	11/21.13/21	4	10.00	1.5	5	12.00	2.2	6	14.00	3	3.98
5	12/21.13/31	4	10.00	2	4	11.38	2.5	8	12.76	3	4.63
	22/41.23/11										
6	11/21.15/41	2	6.00	2	2	7.00	3	2	8.00	4	2.88
6*	12/31.14/31	8	9.00	1.75	4	10.00	2	21	12.00	3.62	4.64
7	12/31.15/41	2	7.75	2	2	7.90	3	2	8.51	2	4.35
	23/11.24/61										
	32/11.32/21										
	35/51.35/61										
8	13/21.15/61	2	6.59	2	4	9.17	3	2	9.41	2	3.93
9	12/41.17/51	2	5.18	3	2	6.36	2	2	6.47	2	3.12
	42/71.47/21										
9*	24/31.25/61	2	6.77	1	4	7.71	2.5	2	7.82	1	4.29
10	14/31.16/71	4	7.00	2	2	7.38	2	2	7.53	3	4.63
	23/71.27/31										
11	53/41.58/71	4	6.69	3	2	7.01	3	2	7.08	2	4.61

Table 3.4. *Truncated transfer functions for unit memory code for codes presented in Table 3.1 for* $\mathbb{Z}_{12}$ *to* $\mathbb{Z}_{16}$

$\mathbb{Z}_q$	Codes	n_1 d_1^2 ϵ_1	n_2 d_2^2 ϵ_2	n_3 d_3^2 ϵ_3	g_∞ dB
12	2 5/11 1.2 7/11 1 5 2 /11 1. 5 5/2 1 5 7 /10 1. 5 10/1 1	2 5.27 2	4 5.54 3	2 6.00 2	3.73
12*	2 5/3 1.2 5/9 1 2 7/3 1.2 7/9 1	2 6.00 2	6 6.54 3.3	6 7.00 2	4.30
13	2 3/7 1.6 4/2 1 2 4/5 1.6 3/8 1 2 9/8 1.6 10/5 1 13 6 9/11 1.2 10/6 1	2 5.96 3	4 6.48 4	2 6.57 5	4.40
13*	3 5/2 1.4 5/7 1 2 8/3 1.2 5/10 1 6 5/4 1.6 8/9 1 3 8/11 1. 4 8 6 1	4 6.34 4	4 6.35 2	4 6.58 2.5	4.67
14	3 5/12 1.3 9/2 1 5 4/11 1.5 10/3 1 6 3/5 1.6 11/9 1	2 5.74 4	2 5.75 2	4 5.96 4	4.37
15	2 11/8 1. 7 4/2 1 7 11/13 1. 2 4 /7 1	4 4.94 2.5	4 5.77 3.5	2 6.22 4	4
16	3 9/14 1 .37/21	2 4.21 3	4 4.34 5	2 4.38 3	3.23
16*	3 6/2 1. 3 6/14 1	4 5.33 4	2 5.39 2	2 5.48 3	4.25

Table 3.5. *Product distance profiles codes with constraint length two for $\mathbb{Z}_2$ to $\mathbb{Z}_8$*

$\mathbb{Z}_q$	Codes	l_{eff}	n_1	d_1^2	ϵ_1	n_2	d_2^2	ϵ_2	l_2	n_3	d_3^2	ϵ_3	l_3
2	101/111.111/101	5	1	1024	2	2	4096	2	6	4	16384	3.5	7
3	111/121.112/211	6	6	729	3	4	2187	3.5	7	26	6561	4	8
	121/111.122/221												
4	113/321.133/321	5	1	1024	2	2	128	3	2	2	4096	3	6
	123/311.123/331												
5	213/431.	6	12	125	3	84	625	4	8				
	243/421	6											
6	115/541.155/521	5	1	1024	2	2	3	3	6	6	729	3	6
7	145/531.123/351	6	6	49.00	3	8	7.31	3.5	7	4	36.90	3.5	7
	153/321.135/541												
7	144/221.152/431	6	4	15.09	3	4	31.51	3	6	6	49.00	3	6
8	355/761.325/711	5	1	1024	3	2	4.69	3	6	2	27.31	3	6
	335/761.355/721												

3.2.3 Constraint Length Two Codes

Tables 3.5 and 3.6 present results for codes that maximize the l_{eff} and d_{prod}^2. The exhaustive search was carried out for codes over $\mathbb{Z}_q$, where $q = \{2, 3, \ldots, 8\}$. The search through the trellis was set to at least seven symbols. As the number of paths increases exponentially with the depth of the search, the search is truncated between seven and nine symbols. It is possible that some long paths with small Euclidean or product distances are missed due to the truncated search. However, the search path was set to a reasonable length to find the paths that dominate the performance of the codes.

3.3 Reduced Search

As the exhaustive search time grows exponentially with the number of elements in the ring and the constraint length of the code, this search method soon becomes impractical due to the required computational time. In order to find good codes in

Table 3.6. *Truncated transfer functions codes with constraint length two for* $\mathbb{Z}_3$ *to* $\mathbb{Z}_8$

$\mathbb{Z}_q$	Codes	n_1	d_1^2	ϵ_1	n_2	d_2^2	ϵ_2	n_3	d_3^2	ϵ_3	g_∞
2	101/111.111/101	1	20.00	3	2	24.00	3	4	28.00	3.25	3.97
3	111/121.112/211	6	18.00	3	4	21.00	3.5	26	24.00	4	5.52
	121/111.122/221										
4	123/311.123/331	4	14.00	3	8	16.00	3.8	14	18.00	3.9	5.44
4	113/321.133/321	4	14.00	3.5	8	16.00	3.2	14	18.00	4	5.44
5	213/431.243/421	12	15.00	3	32	16.91	4.5	84	20.00	4	6.39
6	115/541.155/521	2	8.00	3	2	9.00	2.0	4	10.00	4	4.12
7	145/531.123/351	4	11.70	3.5	2	12.42	6	4	12.79	4	6.13
	153/321.135/541										
8	325/711.365/771	2	10.59	3	4	11.76	3	4	12.10	5	5.98
	335/761.355/721										

a reasonable amount of time, it is imperative that the number of candidate codes be reduced. In the following section, we develop bounds on the effective length of the rate 1/2 systematic recursive convolutional codes. Using this result and some properties of good fading codes from the exhaustive search in Section 3.2, we placed constraints on the tap polynomials $f(D)$ and $g(D)$. The reasoning and the restrictions will be presented in Section 3.3.2. Reducing the number of codes to be searched allows us to search for longer constraint length codes. The results of the reduced search are presented in Section 3.3.3.

3.3.1 Bounds on l_{eff}

In this section, we investigate the maximum l_{eff} possible for the encoder shown in Figure 2.2.

The message polynomial input into the encoder is defined as

$$m(x) = \ldots + m_s x^s + m_{s-1} x^{s-1} + \ldots + m_1 x_1 + m_0. \tag{3.1}$$

where $m_i \in \mathbb{Z}_q$.

With a convolutional encoder, the message polynomials can have infinite length. We follow the convention that the symbols are transmitted in ascending order. In other words, if the message begins to enter the encoder at time 0, the symbol entering the encoder at time i will be m_i where i can range from 0 to infinity. We also follow the convention that any zero term is left out of the expression and if m_s is non-zero and $m_i = 0, \forall i > s$ the degree of $m(x)$ is s or $\deg(m(x)) = s$ and the message length will be considered to be s.

Recall that the encoder performs simultaneous multiplication by $g(x)$ and division by $f(x)$. When working over a field, the division algorithm states that if $f(x) \neq 0$, then there exist polynomials q(x) and r(x) such that for any arbitrary polynomial $g(x)$

$$g(x) = q(x)f(x) + r(x). \tag{3.2}$$

where $deg(r(x)) < deg(f(x))$. The polynomials $q(x)$ and $r(x)$ are known as the quotient and remainder polynomials defined as

$$q(x) = \ldots + q_s x^s + q_{s-1}x^{s-1} + \ldots + q_1 x_1 + q_0. \tag{3.3}$$

$$r(x) = r_{d-1}x^{d-1} + r_{d-2}x^{d-2} + \ldots + r_1 x_1 + r_0. \tag{3.4}$$

with $d = \deg(f(x))$ and $q_i, r_i \in \mathbb{Z}_q$.

When $\mathbb{Z}_q$ is not a field, the existence of q(x) and r(x) is not guaranteed. We will deal with this case in the following section.

With respect to the encoder shown in Figure 2.2, $q(x)$ is the output x_2 of the encoder and $r(x)$ defines the state of the encoder. The output symbols of the encoder at time i are m_i and $(g_0 m_i + q_i) \bmod q$. Note that q is the size of the symbol alphabet and q_i is the coefficient of the quotient polynomial.

We want to find the shortest message, m(x), that leaves and reenters the zero state as $l_{eff} \leq 2(\deg(m(x) + 1))$. Stated another way, we wish to find the $m(x)$ with smallest degree such that

$$m(x)g(x) \equiv 0 \bmod f(x). \tag{3.5}$$

or equivalently

$$m(x)g(x) = q(x)f(x). \tag{3.6}$$

where $r(x) = 0$.

Consider in Equation 3.6 we are looking for the smallest multiple of g(x) which is divisible by f(x). A solution satisfying this restriction is

$$m(x)g(x) = c * \text{lcm}(f(x), g(x)) \tag{3.7}$$

where c is an arbitrary constant in $\mathbb{Z}_q$ and *lcm(f,g)* is the least common multiple of f(x) and g(x).

For notational convenience we will refer to m(x), f(x) and g(x) as m, f and g, respectively.

Consider that $a^{-1}fg = \text{lcm}(f,g)\,\text{gcd}(f,g)$ [50], where a is the leading coefficient of fg. Using this result, we can define that the degree of lcm(f,g)

$$\deg(\text{lcm}(f,g)) = \deg(f) + \deg(g) - \deg(\text{gcd}(f,g)) \tag{3.8}$$

From Equation 3.7 the degree of $m(x)$ is defined by

$$\begin{aligned} \deg(m) &= \deg(\text{lcm}(f,g)) - \deg(g). & (3.9)\\ &= \deg(f) - \deg(\text{gcd}(f,g)). & (3.10) \end{aligned}$$

Therefore the maximum l_{eff} for a code is bounded by

$$l_{eff} \leq 2(\deg(f) - \deg(\text{gcd}(f,g)) + 1). \tag{3.11}$$

This also gives an insight into how to find codes with maximum l_{eff}. When gcd$(f,g) = 1$ then $l_{eff} \leq 2(\deg(f) + 1)$.

For the case when $\mathbb{Z}_q$ is a field and gcd$(f,g) = 1$, we conclude that for the code, $g(x)/f(x)$, the maximum effective length is bounded by $l_{eff} \leq 2(\deg(f) + 1)$ and the shortest path is defined by a multiple of $f(x)$.

In the case when $\mathbb{Z}_q$ is not a field, we need to set restrictions on the divisor such that division is defined. Recall from Section 2.2 that if the leading coefficient of the divisor is a unit in the ring, then the division algorithm can be applied. Thus, the bound in Equation 3.11 holds as well in this case.

However, we can bound tighter if we look at the case when the divisor contains zero divisors. We assume that gcd$(f,g) = 1$ thus $m(x) = cf(x)$ is the smallest message which can set the remainder to zero. We choose c such that we achieve the

maximum number of zero coefficients in the message. Then we can lower the bound on l_{eff} by the number of zero divisors that produce zero when multiplied by c.

Thus, if the divisor, f(x), contains zero divisors, it is possible to reduce the bound to

$$l_{eff} \leq 2(\deg(f) + 1) - n_{zd}; \tag{3.12}$$

where n_{zd} is the maximum number of zero-divisors in f(x) which are set to zero by multiplication by an arbitrary constant c.

To illustrate this, consider the ring $\mathbb{Z}_8$ and $f(x) = x^3 + 2x^2 + 4x + 1$ $\gcd(f, g) = 1$ so $m(x) = cf(x)$. If $c = 4$, then $m(x) = 4x^3 + 4$ and two zero divisors become zero. Thus, the maximum l_{eff} is bounded by 6 from Equation 3.12 and not 8 as in Equation 3.11.

From this section we can conclude from Equation 3.12 that the bound on l_{eff} is maximized when f(x) and g(x) have no common factors, and Equation 3.12 provides a bound on l_{eff}.

3.3.2 Search Definition

Examining the results from the exhaustive search in Section 3.2 for codes with constraint length two, presented in Tables 3.5 and 3.6, one finds that for the tap polynomials defining the codes, many of the polynomials are irreducible. In Section 3.3.1, it was shown that l_{eff} was maximized when the greatest common divisor of the numerator and denominator polynomials was one. This condition is guaranteed when both polynomials are irreducible as the polynomials have no divisors of lesser degree.

Restricting the search to irreducible polynomials for $f(D)$ and $g(D)$ is more restrictive than requiring $\gcd(f(D), g(D)) = 1$, however, results from Section 3.2 indicate that irreducible polynomials are a good choice to search for good codes.

Another reason for this restriction is the connection to shift register sequences. Leaving a non-zero state will cycle through a number of states before returning to the original state. If the polynomial is primitive than the length of the cycle is maximized.

From the results in Section 3.2, the good codes were not necessarily primitive and thus the problem here is to find irreducible polynomials over $\mathbb{Z}_q$. There are several tables of irreducible polynomials in $\mathbb{Z}_p \equiv GF(p)$ where p is prime that are presented in [50]. However, the tables did not provide all of the polynomials that were required.

To obtain the necessary polynomials, we used the Eratosthenes sieve method to find them. To find the irreducible polynomials of degree n using this method, one first creates a list of all possible polynomials of degree n. Then all polynomial factors that form a degree n polynomial are multiplied together, and the resulting polynomial is removed from the list. This is repeated until all possible combinations of factors have been used. After this process the polynomials remaining in the list have no factors with degree less than n.

This method is inefficient to find polynomials with a high degree, due to the large number of multiplications of polynomials required. The Eratosthenes sieve method was acceptable for finding the polynomials used in this dissertation as they did not require polynomials with large degree. Tables of the polynomials can be found in [50] when q is prime and in Appendix A for selected degrees and values of q when q is not prime.

As the number of states increases, so does the search time, we will restrict our search to codes with 256 states or less.

The search was carried out in the same manner as the exhaustive search, except for the additional condition that $f(D), g(D)$ were forced to be irreducible in the ring. This is in addition to the conditions already used for the exhaustive search, i.e., fully reachable trellis, $f(0) = 1$, $\deg(f(D)) = \deg(g(D))$

3.3.3 Search Results

The results are presented in the following tables. The tables present the code characteristics n_{free}, d^2_{free}, $N_{g,e}$ which are the number of paths with the free distance, the square of the free distance and the average number of errors for paths with the free distance, respectively. The parameters n_{prod}, d^2_{prod} and $N_{p,e}$ are the number of paths with the minimum squared product distance, the squared product distance and the average number of errors along paths with the squared product distance, respectively. The effective minimum length of the code l_{eff} is also presented.

In Table 3.7, codes with constraint length two for $\mathbb{Z}_9$ and $\mathbb{Z}_{10}$ are presented. The codes with maximum d^2_{free} are marked with a *. They do not have the maximum l_{eff} as shown by the other codes also included in the table. Here, g_∞ is the asymptotic gain over BPSK. The table does not include many codes above $\mathbb{Z}_{10}$ as the search

time was prohibitive. The results for $\mathbb{Z}_{12}$ are from a partial search of the codes. The codes over $\mathbb{Z}_{12}$ presented have the best distance properties for the partial search. The search was terminated early because of time restrictions. thus. there may be better codes available in $\mathbb{Z}_{12}$.

The restriction that the codes must have a fully reachable trellis forces the code for constraint length two to have q^2 states in the trellis. Each state has q branches exiting and entering the state. Thus a search of depth m has $(q-1)*q^{m-1}$ possible paths to search. As an example. consider $q = 11$ and a search depth $m = 7$. has 1.77×10^7 branches to consider. when $q = 16$ the algorithm must consider 2.51×10^8 paths. The algorithm eliminates a large number of these paths as it does not continue to search paths that have merged with the all-zero path.

In Table 3.8 codes from the reduced search of constraint length three codes are presented. It should be noted from the table that when q is prime l_{eff} is less than or equal to 8. Equality occurs in this table when q is 5 and 7. except. for $q = 3$ the codes have $l_{eff} = 7$. When q is non-prime the maximum $l_{eff} = 6$. The codes also had maximum d^2_{free} for the reduced set of codes except for the $q = 7$ case.

Table 3.9 presents constraint length 4 codes for $\mathbb{Z}_3$ and $\mathbb{Z}_4$ and constraint length 5 codes for $\mathbb{Z}_3$. The encoders for $\mathbb{Z}_3$ have 81. and 243 states for defining polynomials of degrees 4 and 5. respectively. The $\mathbb{Z}_4$ codes have 256 states. For the 256 state codes in $\mathbb{Z}_4$ there are a large number of paths with d^2_{free} and the l_{eff} does not increase as the number of states increases from 64 (see Table 3.8) to 256 states. Also. the 256 state codes have four times the number of branches as the 64 state code as well as a commonly used binary convolutional code. As it provides no gain in asymptotic fading performance over the 64-state code. it is highly unlikely that the 1.2 dB gain on the AWGN channel is worth the added complexity to implement the 256-state code rather than the 64-state code.

Table 3.7. *Codes over $\mathbb{Z}_q$ for with constraint length 2*

Ring	polynomials	n_{free}	d^2_{free}	$N_{g.e}$	g_∞ dB	n_{prod}	d^2_{prod}	$N_{p.e}$	l_{eff}
9	175/571.125/521	2	8.48	4	6.79	6	9.00	3	6
	152/251.142/241								
9*	262/551.232/541	2	10.24	3	7.60	2	243	3	5
	434/211.472/161								
	272/161.422/131								
	224/131.274/161								
10	367/991.337/981	2	9.91	5	7.92	1	1024	3	5
	377/921.347/911								
	183/771.123/771								
	193/761.113/761								
	193/741.113/741								
	183/731.123/731								
	167/391.147/391								
	177/381.137/381								
	177/321.137/321								
	167/311.147/311								
10 *	4 7 6/ 9 8 1	4	10.06	6	7.98	1	64	2	3
	4 3 6/ 9 1 1	8	10.06	5		1	256	2	4
12	5 10 11/1 7 1 †	2	6	2		1	1024	3	5
	1 2 7/1 7 1 †								

† Search not complete

Table 3.8. *Codes over $\mathbb{Z}_3$ to $\mathbb{Z}_8$ with constraint length 3*

Ring	polynomials	n_{free}	d^2_{free}	$N_{g.e}$	g_∞ dB	n_{prod}	d^2_{prod}	$N_{p.e}$	l_{eff}
3	1021/2211.1022/1221 1102/1221.1201/2211	4	21	3.0	6.19	4	2187	3.0	7
4	1311/3321.1131/3211 1113/1321.1333/1231	2	18	4	6.54	1	4096	4	6
5	1223/2131.1213/2231 1143/2421.1343/2441 1322/3121.1312/3221 1312/3221.1242/3411 1442/3431	2	17.76	4	7.12	2	238.729	4	8
6	1154/5231. 1154/5341	6	15.00	4	6.85	4	4096	3	6
7	1325/3621.1263/5231 1422/4651.1564/2241	4	14.90	4.5	7.19	6	90.2172	4.33	8
7^*	1654/5521.1552/3511	2	15.51	4	7.36	2	27.7849	8	4
8	3655/7761. 3253/1721 3523/1271	4	15.51	5.5	7.65	2	4096	3	6

Table 3.9. *Codes over $\mathbb{Z}_3$ and $\mathbb{Z}_4$ with constraint length greater than 3*

$\mathbb{Z}_3$									
degree	polynomials	n_{free}	d^2_{free}	$N_{g,e}$	g_∞ dB	n_{prod}	d^2_{prod}	$N_{p,e}$	l_{eff}
4	10111/21121	8	27	4.3	7.28	8	19683	3.8	9
	11101/22111								
	10121/22111								
	12101/22211								
5	112111/102101	4	30	4.5	7.74	4	59049	4.5	10
	121112/202201								
$\mathbb{Z}_4$									
degree	polynomials	n_{free}	d^2_{free}	$N_{g,e}$	g_∞ dB	n_{prod}	d^2_{prod}	$N_{p,e}$	l_{eff}
4	12213/11021	73	24	5.44	7.78	1	4096	3	6
	13021/32211								
	12031/33221								
	12011/31221								
	12233/13021								
	11223/12031								
	13223/12011								

3.4 Gaussian Codes

Although the main thrust of this dissertation was to find codes for use on the fading channels, the search routines also looked for codes with maximum d^2_{free}. In some cases, the best codes found for fading also have maximum d^2_{free}. In the following tables, the characteristics are presented for the codes, and l_{eff} is included to allow for a comparison with the codes found for the fading channel. In many cases the best code for the Gaussian code has a smaller l_{eff} for the same ring and constraint length than the code for fading. This is to be expected as l_{eff} is not being optimized in the search.

In the reduced search, the polynomials were chosen to maximize the effective length of the code. Because of this the Gaussian codes found in the reduced search may not be the optimal code for the AWGN channel for the given ring and constant length. Here, optimal means having the maximum d^2_{free} over the set of all codes with the given constraint length and given ring. The codes do have the maximum d^2_{free} in the search space. The AWGN codes found by the reduced search for codes over $\mathbb{Z}_q$ where $q \geq 9$ for degree 2 and for $q \geq 4$ for constraint length 3 codes are presented in Tables 3.12 and 3.13.

Table 3.10. *Gaussian unit memory codes over $\mathbb{Z}_3$ to $\mathbb{Z}_{11}$*

Ring	polynomials	n_{free}	d^2_{free}	$N_{y,e}$	n_{prod}	d^2_{prod}	l_{eff}	$N_{p,e}$
2	11/01	1	12.00	1	1	64.00	3	2
3	11/21.12/11	2	12.00	2	2	81.00	4	2
4	11/21.13/21	4	10.00	1.5	1	64.00	3	1
5	12/21.13/31 22/41.23/11	4	10.00	2	4	25.00	4	2
6	12/31.14/31	8	9.00	1.25	2	27.00	3	1
7	12/31.13/21 14/51.15/41 22/31.23/11 23/41.24/61 25/41.32/11 32/21.33/51 34/21.35/51 35/61	2	7.75	2	2	5.2711	4	2
8	32/41	4	7.17	2	1	16.00	2	1
9	24/31.25/61	2	6.77	1	2	27.00	3	1
10	13/41.17/61 14/31.16/71 23/71.27/31 33/81.34/11 36/91.37/21 43/11.47/91	4	7.00	2	1	64.00	3	1
11	24/81.27/31 53/41.58/71	4	6.69	3	2	2.8068	4	2

Table 3.11. *Gaussian unit memory codes over $\mathbb{Z}_{12}$ to $\mathbb{Z}_{16}$*

Ring	polynomials	n_{free}	d^2_{free}	$N_{g,e}$	n_{prod}	d^2_{prod}	$N_{p,e}$	l_{eff}
12	2 5/3 1.2 5/9 1 2 7/3 1.2 7/9 1 3 5/2 1.3 5/10 1 3 7/2 1.3 5/10 1	2	6.00	2	2	27.00	1	3
13	35/21.45/71	4	6.34	4	4	1.2173	2	4
14	1 8/10 1	2	6.06	2	1	16.	1	2
15	2 9/ 9 1	2	5.63	2	2	9	1	2
16	3 6/2 1	4	5.33	4	1	16	2	2

Table 3.12. *Gaussian codes with constraint length over $\mathbb{Z}_3$ to $\mathbb{Z}_8$*

Ring	polynomials	n_{free}	d^2_{free}	$N_{g,e}$	n_{prod}	d^2_{prod}	$N_{p,e}$	l_{eff}
2	111/101	1	20.00	2	1	1024.00	2	5
3	112/211.111/121 121/111.122/221	6	18.00	3	6	729.00	3	6
4	212/111.212/131 232/331.232/131	14	16.00	3.5	2	256.00	2.5	4
5	213/431.243/421 233/441.223/411	4	15.00	2	12	125.00	3	6
6	214/131.254/131	10	13.00	3.4	2	256.00	2.5	4
7	235/651.245/621 314/651.364/621	2	12.31	3	2	15.0909	3	6
8	232/741	8	11.51	6	1	64	2	3
9	262/551.464/281 434/211.232/541	2	10.24	3	2	243	3	5
10	357/981.357/921 367/951.347/951	2	10.53	4	2	34.549	3	5
12	3 11 7/1 8 1†	2	9.07	4	2	81	4	2

† Partial Search

Table 3.13. *Gaussian codes with constraint length three over $\mathbb{Z}_3$ to $\mathbb{Z}_8$*

Ring	polynomials	n_{free}	d^2_{free}	$N_{g.e}$	n_{prod}	d^2_{prod}	$N_{p.e}$	l_{eff}
3	1021/2211.1022/2111 1022/2221.1102/1221 1201/2211	4	21.00	3	4	2187	3	7
4	1123/1311	2	18.00	4	1	4096	2	6
5	1213/2231.1343/2441 1312/3221.1242/3411	2	17.76	4	2	239	4	8
6	1252/3121	30	17.00	4.47	1	1024	3	5

3.5 Summary

In this chapter, details of the exhaustive search were given in Section 3.2. Codes found using the exhaustive search were presented in Sections 3.2.2 and 3.2.3. The results from the exhaustive search showed that desirable codes shared similar characteristics. In Section 3.3, a reduced search using these results and a bound on l_{eff} was developed. The code generator polynomials were restricted to irreducible polynomials as it was shown in Section 3.3.1 that l_{eff} could be maximized if the feedforward and feedback polynomials had no common divisors. The reduced search was used to search for codes with higher constraint lengths and values of q which were impractical for an exhaustive search.

Codes which have maximum d^2_{free} were presented in Section 3.4. These were included for two reasons: first, at low SNR values, d^2_{free} plays a significant role in the error probability, and second, to be able to compare the codes found with maximum l_{eff} and the codes found with maximum d^2_{free} on the Rayleigh fading channel.

The codes found in the reduced search are not necessarily optimum. However, most of the best codes found in this chapter have the maximum effective length for the constraint length. This indicates that they will have a good asymptotic performance in severe fading.

The observation was made that the codes over $\mathbb{Z}_q$ when q is prime have a larger l_{eff} than when q is non-prime. Also the bound in Equation 3.12 is loose when q is non-prime and for larger constraint lengths. For example, for constraint length 3 the codes found have one zero divisor in the denominator and Equation 3.12 predicts the maximum of 7. The maximum number found was 6. Also, the only non-prime q code with constraint length 4 that was investigated had $l_{eff} = 6$ while the bound for this code was 8.

The search routine stored the first 20 terms of the transfer function, and selected the codes based on l_{eff}, d^2_{free} and d^2_{prod}. These terms can be used to estimate the asymptotic performance, however, there are other terms that may affect the performance at low SNR values. Thus, we simulate the performance of of selected codes in the next chapter.

Chapter 4

Performance Results for Selected Codes

In this chapter, we present simulation results of selected codes and comparisons with previously found codes in the literature, where applicable.

As the bounds or estimates approximate the performance of the codes, we simulated the coded system to verify the expected performance of the various coded systems. Some of the results for codes from the literature are included. In some cases the original authors' simulation results are used or a bound was computed using their results.

In the following section, the simulations are described. The block diagram of the fading channel simulator is included. The simulator was designed to handle many different channel conditions, including correlated fading and shadowing for Rician and Rayleigh channels. The following sections present the simulation results for several codes over different rings. The results are compared with other codes in the literature, the best AWGN codes of the same ring or the uncoded modulation.

4.1 System Simulation

The simulation results are from Monte-Carlo simulations of the system. In these simulations, a random symbol enters the encoder and the coded symbols are sequentially mapped onto an MPSK signal. The symbol is multiplied by the channel gain which is supplied by the fading channel simulator. Gaussian noise is added to the resultant signal before entering the receiver. A Viterbi decoding algorithm is used to decode the information symbols. The decoder observes approximately 6 times the constraint

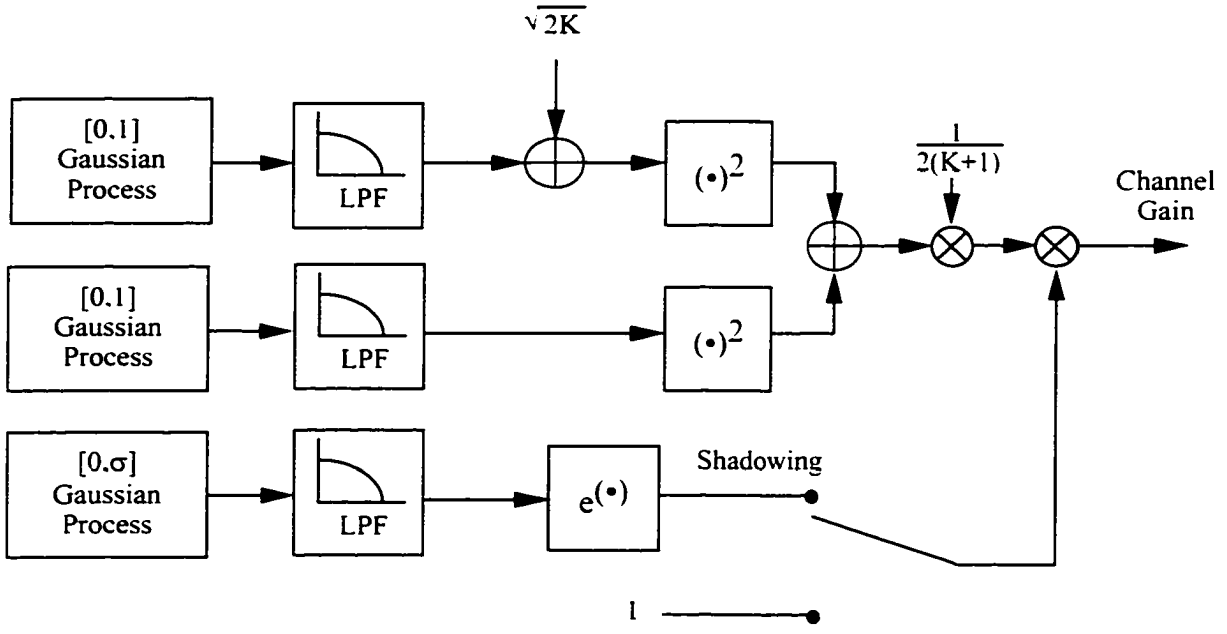


Figure 4.1. *Block diagram for fading simulator used in simulations*

length of the code before making a decision on an information symbol [35]. An error counter is incremented, if the symbol decision is in error. The entire process is repeated until one of the termination criteria is met.

The program stops when 300 independent error bursts are observed. An error burst is defined as one or more symbol errors preceded and followed by a number of correct symbol decisions. This number of correct symbols is referred to as a guard interval [22] and in the simulations this interval was set to 12 symbols. The program also stops if an upper limit on the number of simulation intervals is reached. This usually occurred at the lower error rates. In this case, the point was included in the plot if there were at least 100 symbol errors observed

The fading channel simulator is capable of simulating the Rician channel, lognormal shadowed Rician and lognormal shadowed Rayleigh channels [60]. The model originated in [55] and has been used in several studies for fading channels for satellite and terrestrial digital communications such as [59, 58, 57, 60]. A basic system block diagram is shown in Figure 4.1.

We used a K=0 dB Rician setting for this model to simulate Rayleigh fading. For the simulations, ideal interleaving was employed that adjacent symbols are affected by

independent Rayleigh fading. As in [59], we make the slow fading assumption, that is, that the fading is constant over at least one symbol interval and can be approximated by a single fading sample per channel symbol interval.

The ideal channel state information (CSI) of the channel (i.e., the fading amplitude) is provided to the decoder. The decoder uses the CSI information to compute the metric using Equation 2.18. In practical implementation, the CSI information must be estimated and the estimate is noisy or imperfect. When the CSI is imperfect, the code performs worse than in the case of perfect CSI. The amount of loss is dependent on the channel and the method used to estimate the CSI. The ideal CSI case will lower bound the performance of an actual system.

4.2 Codes over $\mathbb{Z}_3$

Simulation results for the 3, 9, and 27 state codes over $\mathbb{Z}_3$ are presented in Figure 4.2. The codes are 11/21, 111/211 and 1102/1221. The codes have the best d^2_{free} and l_{eff}, which means that they are the best codes found for Rayleigh fading and AWGN. The gains over uncoded 3PSK at the Symbol Error Rate (SER) of 10^{-3} on the Rayleigh fading channel are 21.4 dB, 22.7 dB, and 23.9 dB, respectively. The efficiency of the code is 0.7925 bit/symbol, where efficiency is defined as the number of bits transmitted during each channel use.

4.3 Codes over $\mathbb{Z}_4$

The rate 1/2 ring codes for $\mathbb{Z}_4$ presented in Section 3.2 have an efficiency of 1 bit/symbol. In Rhee et al. [26], a four state binary trellis code is presented with the same efficiency. This code is a rate 1/2 binary code which was designed for the Rayleigh fading channel. Their $(5,2)_8$ has a $l_{eff} = 3$ and $d^2_p = 32$ [26]. The code was designed for the Rayleigh channel as was our 11/21 code. The octal representation of the code means that the top tap polynomial is 101 and the bottom polynomial is 010. Note that there is no feedback in their trellis codes. From Table 3.1 the 11/21 code has a $l_{eff} = 3$ and $d^2_p = 64$.

The asymptotic coding gain difference in the Rayleigh channel associated with a

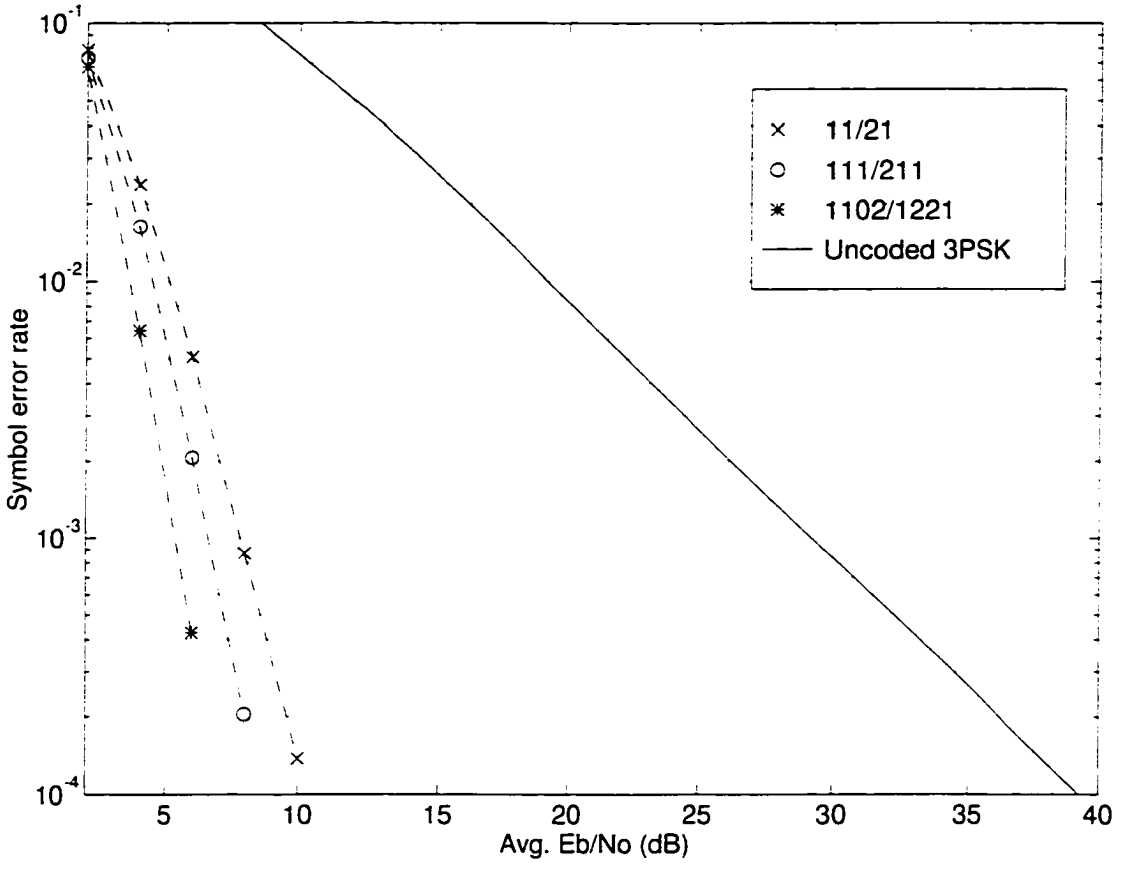


Figure 4.2. *Performance results for codes over* $\mathbb{Z}_3$

different d_p^2, but the same l_{eff} is given by [32]

$$\Delta g_\infty = \frac{10}{L} \log \left(\frac{d_{p2}^2(L)}{d_{p1}^2(L)} \right) \frac{\alpha_1}{\alpha_2}. \tag{4.1}$$

where L is the length of the minimum error event and α_i is the average number of code sequences having the effective length and squared product distance d_{pi}^2. From Equation 4.1 we would expect a performance difference of $3.3 * \log(2) = 1$ dB. The system was simulated to verify the performance of the system on the Rayleigh fading channel. In Figure 4.3, the results of our simulation are shown. The simulation results for the trellis code are taken from [26]. As seen in the figure at 10^{-3}, the difference in performance is slightly above 1 dB. Here, we assumed in our system the binary bit stream is mapped to $\mathbb{Z}_4$ to achieve a Gray mapping on the 4PSK modulation.[1]

With a Gray mapping, we approximate the bit error rate assuming 1 bit error per symbol. (If we assume a natural mapping[2] , then we would approximate 1.5 bit errors per symbol error and the gain would be reduced to approximately 0.7 dB at 10^{-3}). Regardless of the mapping, the $\mathbb{Z}_4$ code is still better than the optimal rate 1/2 4PSK binary trellis code from [26] designed for the same channel conditions as the 11/21 code.

A four state feedback convolutional code for $\mathbb{Z}_4$ using 4PSK is presented in [3]. The search was for good ring codes over the AWGN channel. However, the 11/21 code over $\mathbb{Z}_4$ presented here has the same free distance as their 0221/2231 code. We compare the first three terms in the Euclidean and product transfer functions in Table 4.1. The error columns in the table are symbol errors. The table shows that when the number of errors are considered the 11/21 code should have better performance than the 0221/2231 for both the AWGN and Rayleigh fading channels. The reason is the 0221/2231 [2] and the 11/21 code have the same number of paths for the first four terms, but the 0221/2231 code has more symbol errors associated with the paths with the exception of one path. In the code search presented citeBaldini1, they maximized the d_{free}^2 and minimized the number of paths with the squared free

[1] A Gray mapping ensures that the nearest signals in a modulation differ by only one bit. To achieve this for this system, there is a mapper before the encoder which maps a binary input stream onto 4-ary symbols by: $00 \mapsto 0, 01 \mapsto 1, 11 \mapsto 2, 10 \mapsto 3$

[2] The natural mapping uses the binary representation of the symbol. That is. $00 \mapsto 0, 01 \mapsto 1, 10 \mapsto 2, 11 \mapsto 3$

Table 4.1. *Transfer functions for the 0221/2231 code and 11/21*

0221/2231 from [3]						
Euclidean			Product			
n_e	d_e^2	errors	n_p	d_p^2	errors	⋮
4	10	2.5	1	64	2	3
5	12	2.4	4	32	2.5	4
6	14	2.7	1	256	2	4
19	16	3.6	4	64	2.5	5
11/21 code						
Euclidean			Product			
n_e	d_e^2	errors	n_p	d_p^2	errors	l
4	10	1.5	1	64	1	3
5	12	2.2	4	32	1.5	4
6	14	3	1	256	2	4
19	16	2.9	4	64	2.5	5

distance. They did not consider multiple terms in the transfer function or the number of errors associated with the paths with d_{free}^2. Both the 11/21 and the 0221/2231[2] were simulated and the results are shown in Figure 4.3. On the AWGN channel the 11/21 code is about 0.2 dB better than the 0221/2231 code and on the Rayleigh fading channel the performance is about 0.5 dB better at a BER of 10^{-3}. The gains are also achieved with a less complex encoder. For example, the 11/21 encoder has 1 delay element, 1 multiplier, and 2 adders while the 0221/2231 has 3 delay elements, 4 multipliers and 3 adders.

We compare the performance of the 16 state code over $\mathbb{Z}_4$, 123/331 presented in Table 3.5 with the codes in literature. Here, we compare the 16 state binary code for 4-PSK, namely, the $(64, 52)_8$ code [26], with the $d_p^2 = 128.0$ and $l_{eff} = 5$. For the 123/331 from Table 3.5, $d_p^2 = 1024$ and $l_{eff} = 5$. Comparing the two codes using Equation 4.1 we expect about 1.8 dB coding gain by using the $\mathbb{Z}_4$ code. Figure 4.4 shows the bit error performance on the AWGN and Rayleigh channels. The figure shows that the $\mathbb{Z}_4$ 16-state codes outperforms the 16-state TCM code $(64, 52)_8$ [26] by

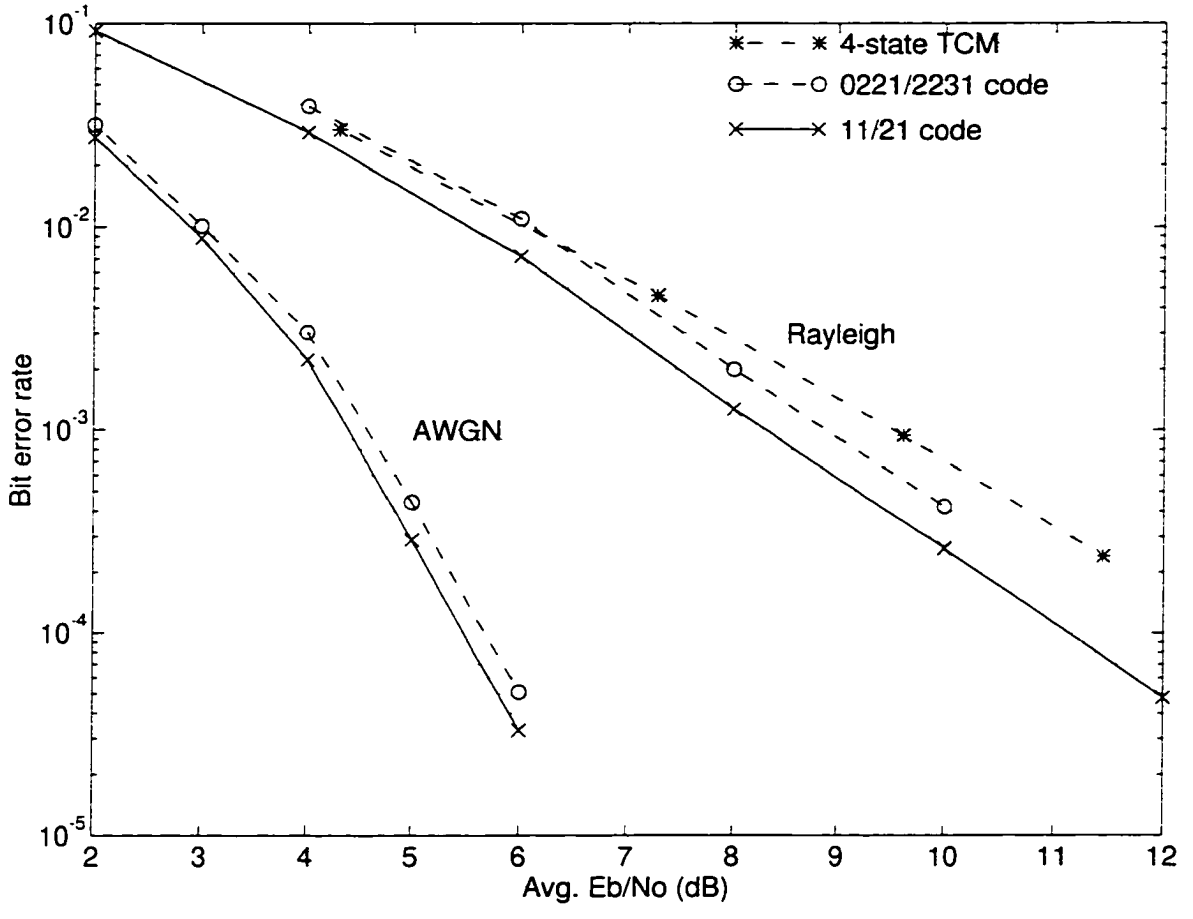


Figure 4.3. *Simulation results for 4-state 4PSK rate 1/2 codes with 1 bit/symbol efficiency.*

1.8 dB at a bit error rate of 10^{-3}. The 232/331 $\mathbb{Z}_4$ code from Table 3.12 is also shown in the figure. This code was selected for its performance on the AWGN channel. In the figure, this code performs better than the 123/331 code on the AWGN, but worse on the Rayleigh channel. The performance difference will be more pronounced at higher SNR as the 232/331 code's performance will be proportional to SNR^{-4} while the 123/331 codes performance is proportional to SNR^{-5}.

The squared free distance of the 123/331 code is 16 which is better than the Ungerboeck 16-state TCM code [3] ($d^2_{free} = 12$) but has a higher number of paths with the squared free distance. As found in [2], the number of nearest neighbours for the ring code is 14 times greater than the 4-state TCM code.

The comparison is presented in Table 4.2. For the AWGN channel, we use Ungerboeck's code as the reference, and for the Rayleigh fading channel we use Rhee's 16-state TCM [26] as the reference as it was designed for Rayleigh fading. Note that Rhee et al. claimed this code to be the optimal rate 1/2 binary TCM scheme for 4PSK. We have achieved a 1.8 dB gain over their code using the ring code working over $\mathbb{Z}_4$. We have achieve simulation results for the 16-state TCM code are from Figure 1 in [26].

For the comparison of codes with 64 states, the 64 state code $(77, 224)_8$ [26] is better asymptotically with l_{eff} =8 and $d^2_{prod} = 256$. The codes in Table 3.9 have all $l_{eff} = 6$, thus have worse performance at high SNR. However, the 64-state codes, presented in Table 3.9, have other paths that are dominant at low SNR. The first three terms of the transfer functions are used to approximate the performance of the $\mathbb{Z}_4$ 1311/3321 code (4096, 19654, 512 with symbol distances of 6, 7, and 8) and the first term for the $(77, 224)_8$ code. As shown in Figure 4.5, the codes perform very similarly on the Rayleigh channel in the bit error range of 10^{-3} to 10^{-4}. The gain of both codes at 10^{-3} over uncoded BPSK is about 18.5 dB. The 1311/3221 code appears to be slightly better in performance at the error rates shown. The complexity of the ring code is greater than the trellis code, so the slightly better performance is offset by the additional complexity required to achieve it. For example, the decoder for the TCM code from [26] requires 2 additions and 2 comparisons per state per bit and for the ring code there are 4 additions and 2 comparisons per state per bit.

AWGN codes			
code	n_e	d_e^2	g_∞ dB over BPSK
Ungerboeck TCM 16-state 4-PSK	1	12	4.77
232/331	14	16	6.02
123/331	4	14	5.44
Rayleigh fading			
code	l_{eff}	d_{prod}^2	gain dB over BPSK BER= 10^{-3}
$(64,52)_8$ [26] feedforward TCM	5	128	16
232/331	4	256	17.3
123/331	5	1024	17.8

Table 4.2. *Comparison of r=1/2 ring codes with other codes with the same number of states and efficiency.*

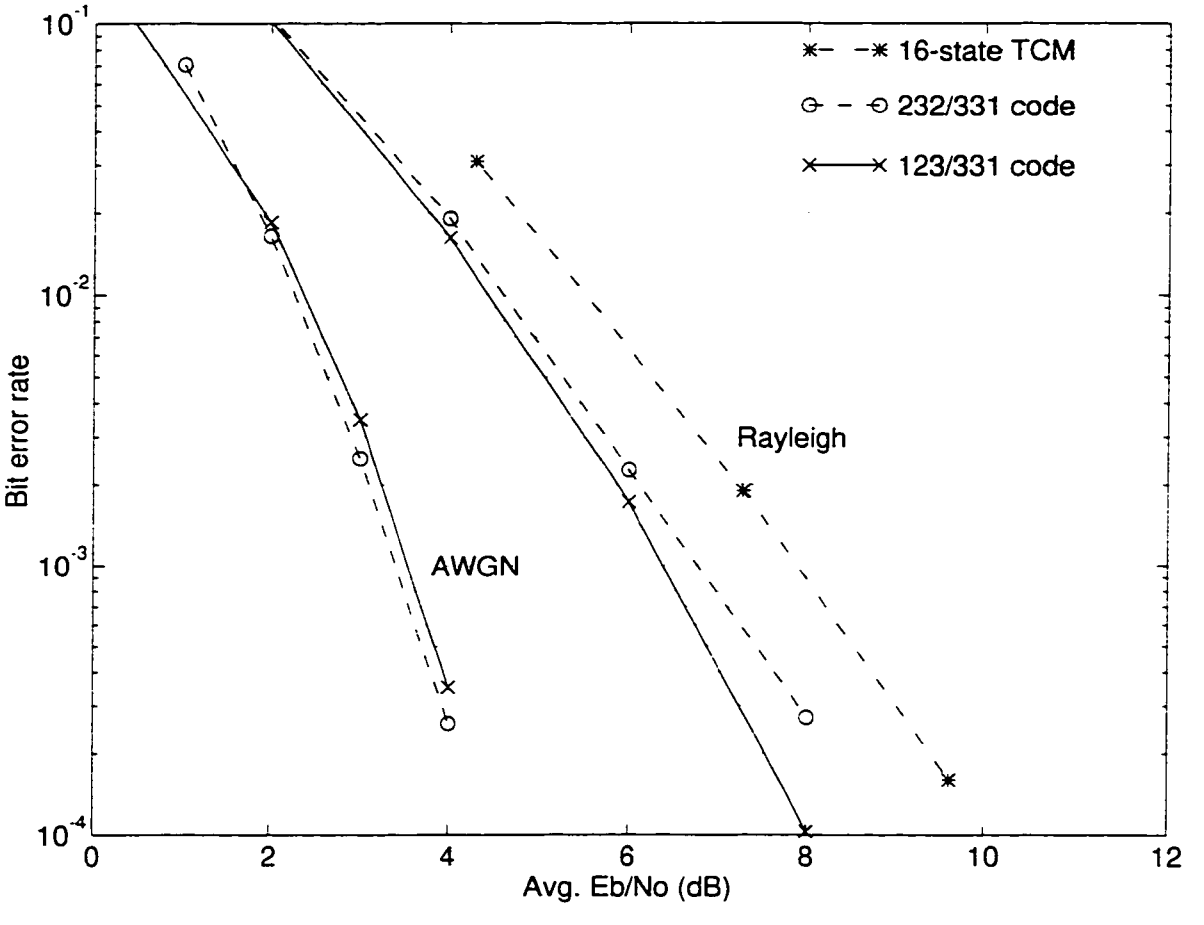


Figure 4.4. *Simulation results for 16-state 4PSK rate 1/2 codes with 1 bit/symbol efficiency.*

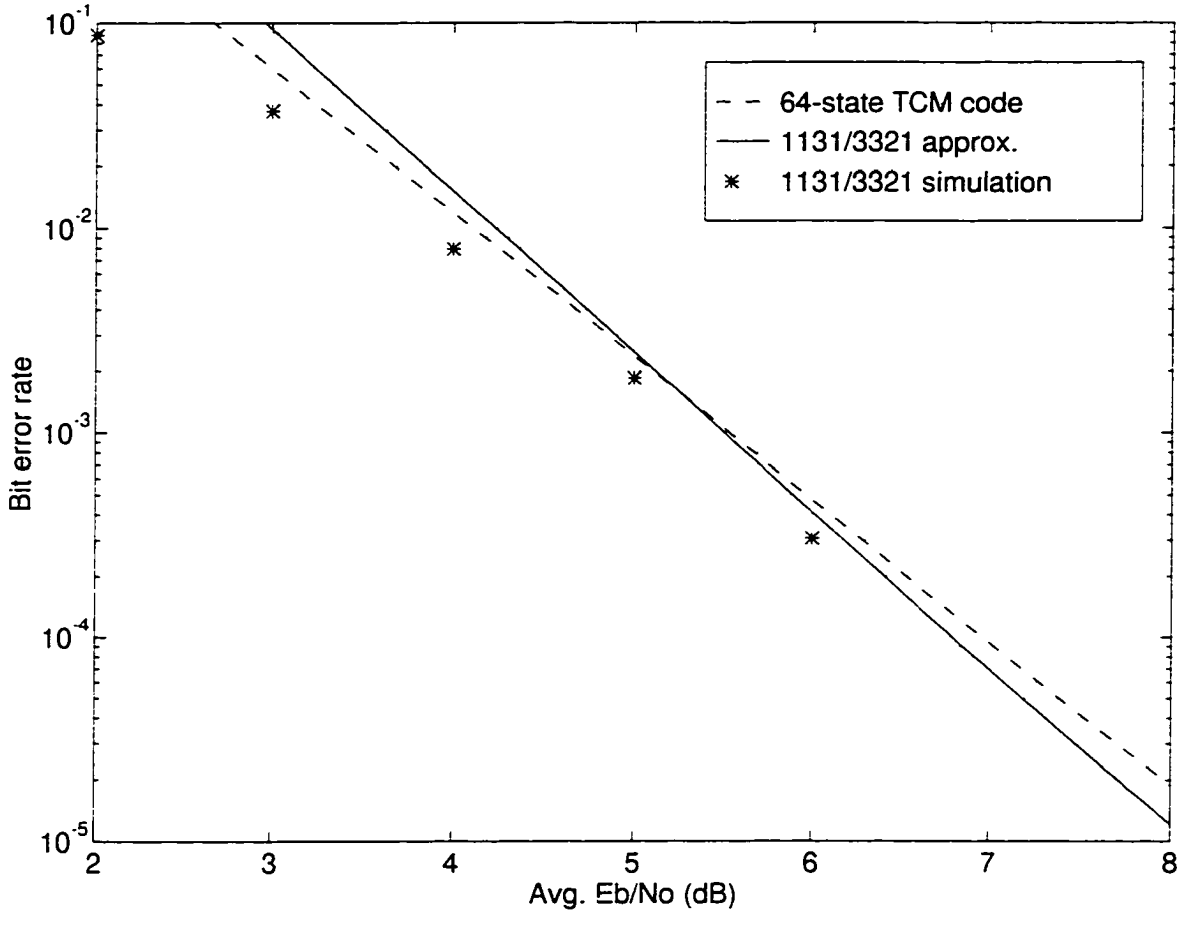


Figure 4.5. *Simulation results for 64-state 4PSK rate 1/2 codes with 1 bit/symbol efficiency.*

4.4 Codes over $\mathbb{Z}_6$

In [27], Zetterberg found two linear 6-state codes for 6-PSK with $d^2_{free} = 9$ and $n_e = 6$. This compares well with the 6-PSK 6-state codes found here. In Table 3.10, the AWGN codes have the same d^2_{free} but have more paths with the squared free distance, (i.e., $n_e = 8$). The codes have the same rate of 1.29 bits per symbol and an asymptotic coding gain of 4.64 dB over BPSK on the AWGN channel.

The codes designed for the Rayleigh fading channel, which are presented in Table 3.1, have $d^2_{free} = 6$ but a greater $d^2_{prod} = 64$. Comparing the fading code with the code for the AWGN channel, which is presented in Table 3.10 with $d^2_{free} = 9$ and $d^2_{prod} = 27$, we would expect a 1.79 dB asymptotic loss on the AWGN channel but should achieve an asymptotic gain of 1.25 dB on the Rayleigh channel. However, from simulations of the 11/21 and the 12/31 codes on the Rayleigh fading channel the 12/31 performs better on the Rayleigh fading channel at low SNR. The simulation results and the upper bound calculated from the truncated transfer function are shown in Figure 4.6. The upper bounds show that there is a cross-over in performance of the two codes around 12.5 dB. Thus, the 11/21 code is expected to do better at higher SNR. The simulations show the 11/21 code's performance descending at a higher rate than the 12/31 code and the performance is very similar at the 12-13 dB range.

The main reason for this is that other error events contribute to the error performance and there are events with small d^2_p in the transfer function. Also at low SNR, noise affects the performance more than the fading and thus the free distance of the code will be a factor in the the error performance of the code. For the codes presented in the tables, when there was a tie on the first term of the product transfer function, multiple terms were considered for the selection of the codes. The main selection process maximized the asymptotic error performance on fading (i.e., by finding the code with the maximum d^2_{prod} and l_{eff}. However other codes were included when the selected code had other terms which dominated the performance at low SNR.

In the case of the 115/541 code over $\mathbb{Z}_6$, the code with the maximum l_{eff} does not have better performance than the code for the AWGN channel. This is because in AWGN the performance is influenced by other terms in the transfer function. In Table 4.3, for symbol distances between 4 and 8, the minimum $d^2_{prod}(l)$ are shown for the 115/541 and 214/131 codes. When $l = 6, 7, 8$ the $d^2_{prod}(l)$ for the 115/541 code

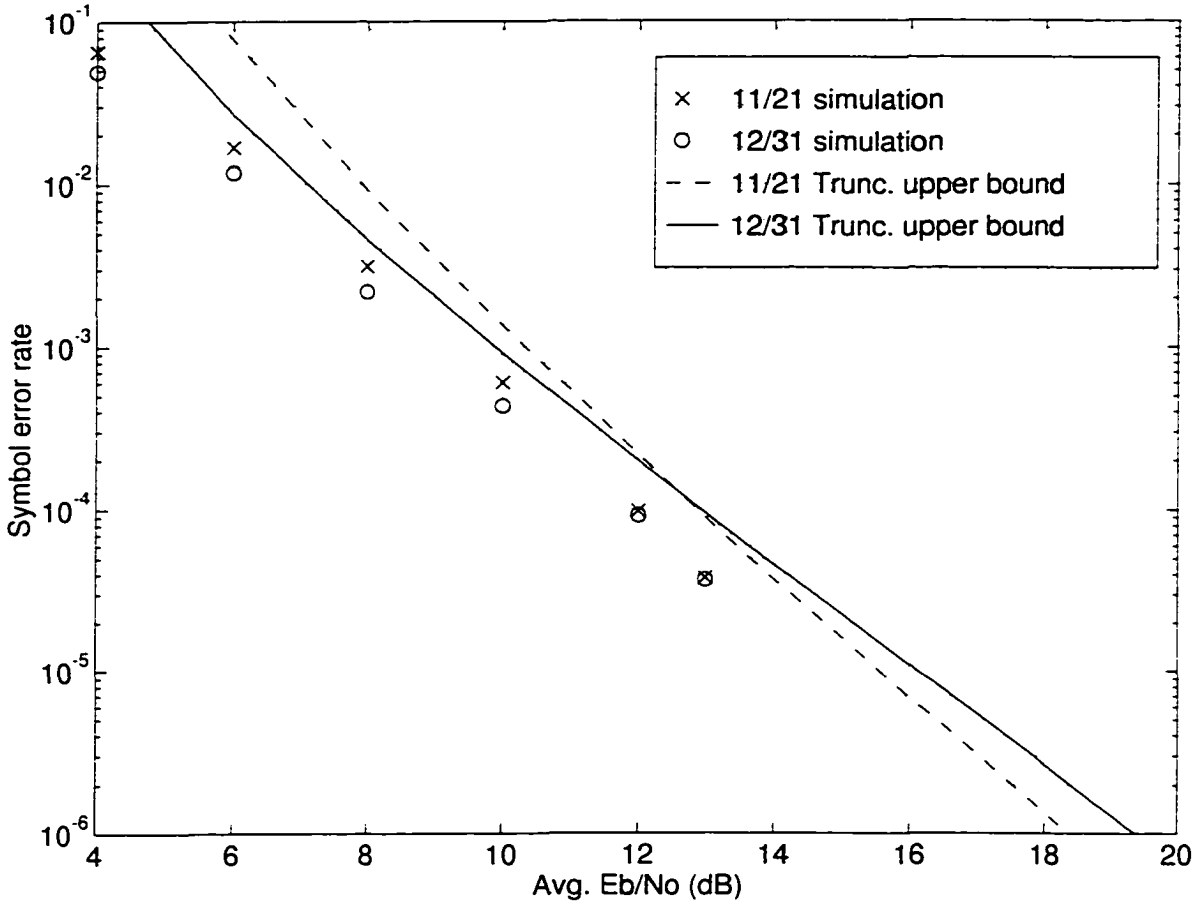


Figure 4.6. *Performance comparison on the Rayleigh fading channel between the 6-state $\mathbb{Z}_6$ codes over 6PSK with maximum d^2_{free} (12/31) and maximum d^2_{prod} (11/21).*

Table 4.3. *Terms from transfer function of the 115/541 and 214/131 code*

115/541						
Euclidean			Product			
n_e	d_e^2	errors	n_p	d_p^2	errors	l
2	8	3	1	1024	2	5
2	9	2	4	3	3	6
4	10	4	1	3	2	7
2	11	6	4	3	4	8
214/131 code						
Euclidean			Product			
n_e	d_e^2	errors	n_p	d_p^2	errors	l
10	13	3.4	2	256	2.5	4
2	14	6	2	243	3	5
22	15	4.1	6	36	3.7	6
6	16	3.5	4	27	3	7
			8	36	4	8

are less than those of the 214/131 code. At low SNR these terms will contribute to the error performance. Although the 115/541 code will perform better asymptotically in Rayleigh fading, the 214/131 code is about 0.5 dB better in the SER range of 10^{-3} to 10^{-4}. On the AWGN channel the 214/131 has a 2.1 dB gain over the 115/541 code. The simulation results and performance estimates are shown in Figure 4.7. As shown in the figure, there is a cross-over in the performance on the Rayleigh channel around 13 dB. Thus, the 115/541 code is better in fading asymptotically, however in a practical range of interest the 214/131 code is better.

For the constraint length 3 $\mathbb{Z}_6$ codes shown in Figure 4.7, the Gaussian code performs only slightly worse in the symbol error rate range 10^{-3} to 10^{-4}. The codes could be considered identical in this error range on the Rayleigh fading channel. Similar to the previous case, the upper bound indicates that there will be a cross-over and the fading code (1154/1431) will perform better than the Gaussian code (1252/3121). However, the Gaussian code has a 0.5 dB performance gain over the

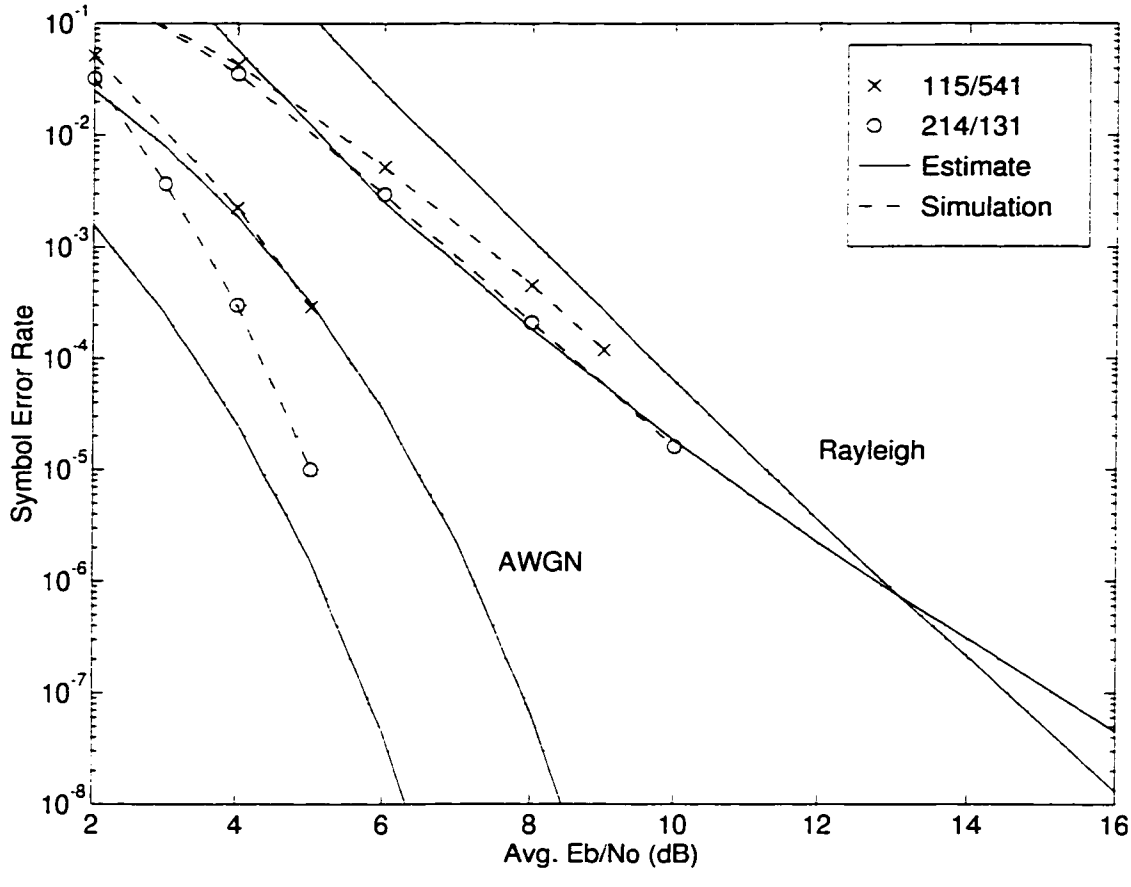


Figure 4.7. *Simulation results for the 36-state code over $\mathbb{Z}_6$. Comparison between 214/131 and 115/541 codes on the Rayleigh fading channel.*

fading code on the Gaussian channel. As the two codes perform similarly in Rayleigh fading, the Gaussian code is recommended at an E_b/N_0 below 8 dB as the code will have better performance in less severe fading since d^2_{free} is greater.

In Figure 4.9, the simulated symbol error performance on the Rayleigh fading channel is shown for three codes over $\mathbb{Z}_6$. The codes are the 11/21, 115/541 and 1154/5341 and have 6, 36 and 216 states, respectively. The figure also shows the asymptotic estimate of the performance which was obtained using two terms of the product transfer function. The figure shows that the performance of the code approaches the theoretical estimate. The coding gains achieved by these codes at a symbol error rate of 10^{-3} over uncoded 6-PSK are 19.6, 21.8, and 23.6 dB respec-

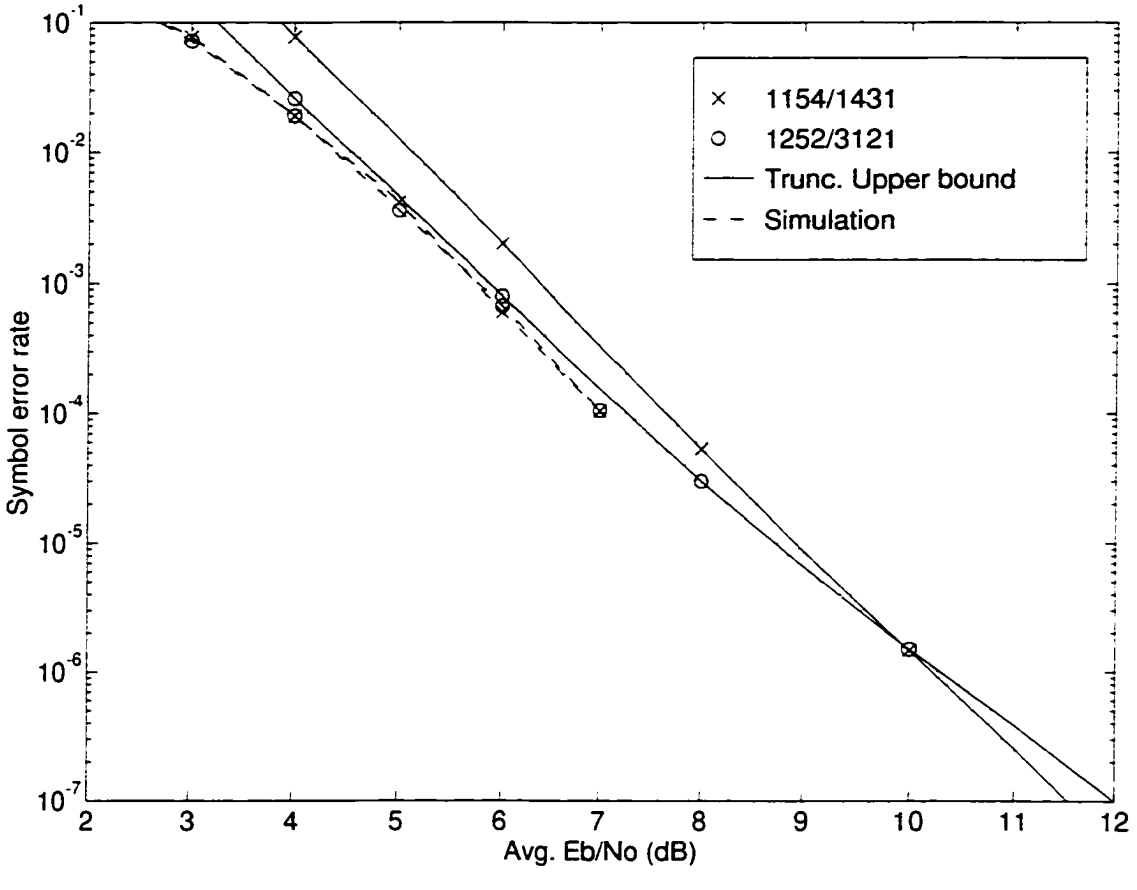


Figure 4.8. *Simulation results for the 216-state code over $\mathbb{Z}_6$. Comparison between 1154/1431 and 1252/3121 codes on the Rayleigh fading channel.*

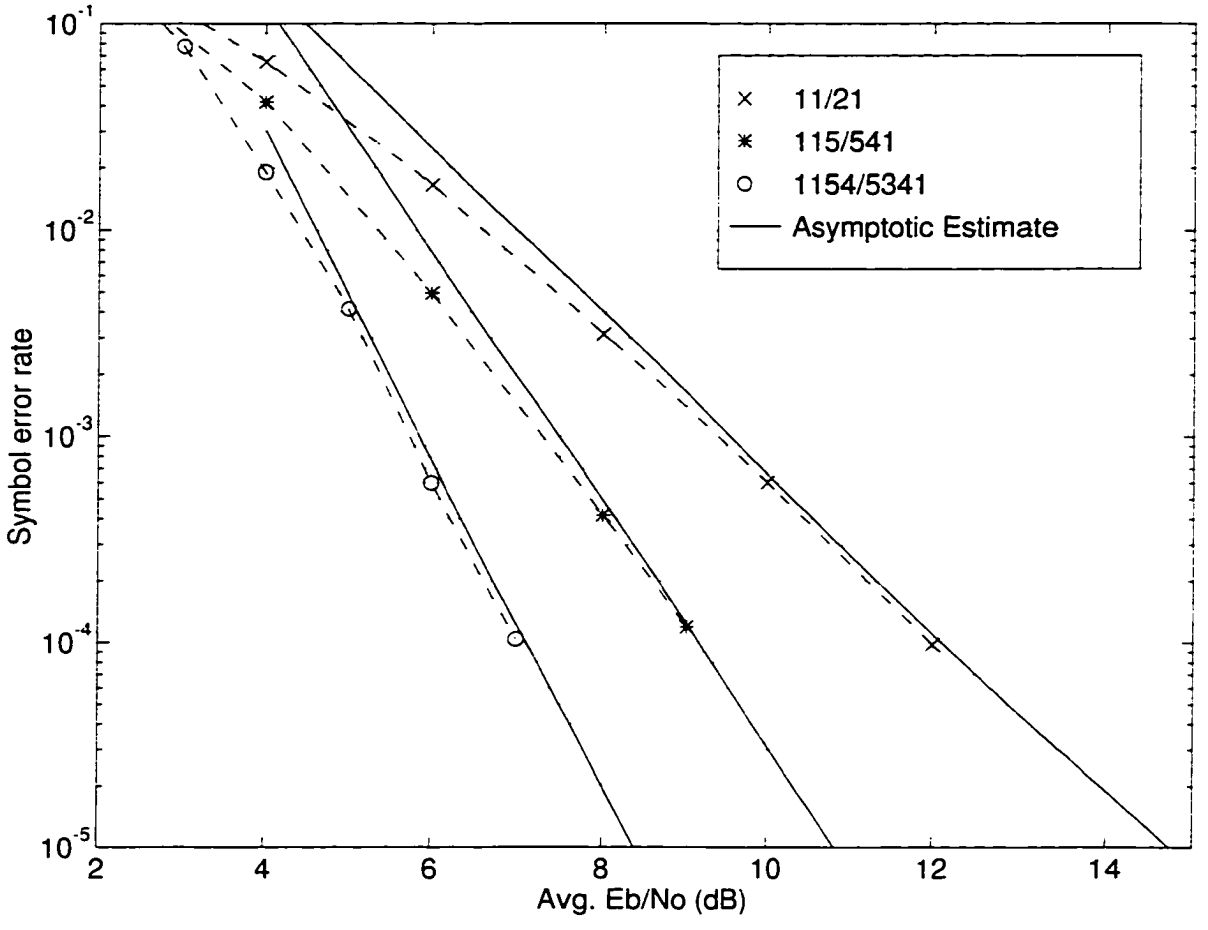


Figure 4.9. *Simulation and asymptotic estimates for codes over $\mathbb{Z}_6$ with 6, 36, and 216 states.*

tively. For a symbol error rate of 10^{-4} the gains are 26.5, 29.7 and 31.9 dB over uncoded 6-PSK.

4.5 Codes over $\mathbb{Z}_8$

In Figure 4.10, the simulation results over the Rayleigh fading channel are presented for an 8-state code, 13/21, and a 64-state code 325/711. The first two terms of the product transfer function are used to form a truncated upper bound for the symbol error performance. In this case, we find the second term of the transfer function is dominant in the range of interest for both codes. The dominant paths in the range of interest have a length of 4 and 6 for the 8 and 64 state codes, respectively. At a symbol error rate of 10^{-3} the gains over uncoded 8-PSK on the Rayleigh channel are 24.4 dB and 26.4 dB for the 8 and 64 state code. Considering that 3 bits/information symbol are sent, and assuming that half the bits are in error (i.e., 1.5 bit errors / symbol error) the gain of this system over uncoded BPSK is 11.5 and 29 dB at bit error rates of 10^{-3} and 10^{-4}, respectively for the 8-state code. For the 64 state code the gains at the respective thresholds are 13.2 and 31.7 dB.

4.6 Codes over $\mathbb{Z}_9$ and $\mathbb{Z}_{12}$

A few 9-state 9PSK codes were presented in [27] with $d^2_{free} = 6.97$. The ring codes (for AWGN) presented in Table 3.11 have $d^2_{free} = 6.77$, or a performance difference of 0.06 dB on the AWGN channel. The actual code was not specified in [27], so the performance in fading could not be calculated.

For the fading case, the fading code (12/41) presented in Table 3.1 has $l_{eff} = 4$ and $d^2_{prod} = 1.404$. The parameters for the AWGN ring code 24/41 presented in Table 3.11 are 3 and 27, respectively. Figure 4.11 shows the simulation results for the 12/41 and 24/41 code. The truncated upper bound shows a cross-over in performance at 13 dB. The simulation results indicate that the AWGN code performs about 0.4 dB better than the 12/41 code at an SER of 10^{-3}.

These codes have the same rate as uncoded 3-PSK or 1.585 bits/symbol and the asymptotic gain over 3-PSK is 3.53 dB (4.28 dB over BPSK) on the AWGN channel.

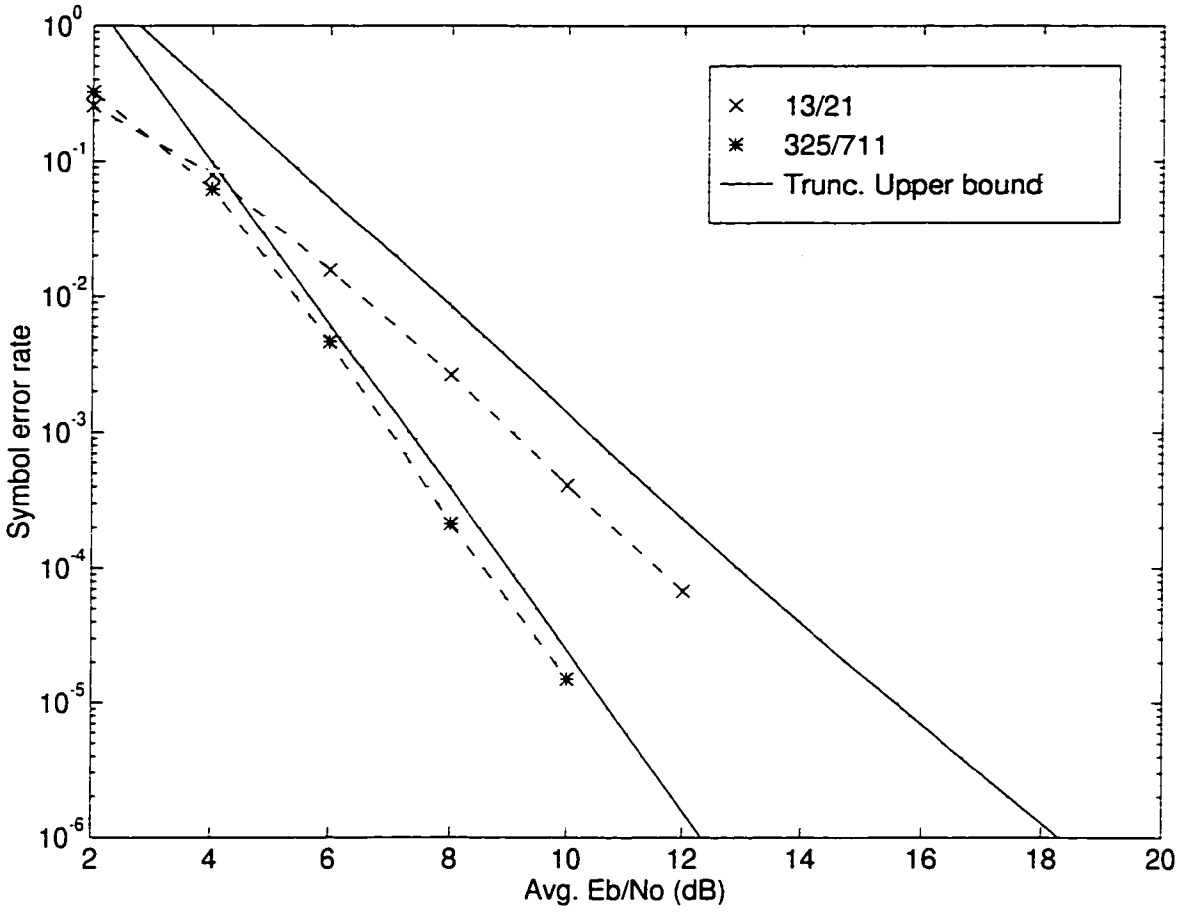


Figure 4.10. *Simulation and truncated upper bounds for codes over $\mathbb{Z}_8$ with 8 and 64 states.*

At an SER of 10^{-3}, the 12/41 code has a 19.5 dB gain over uncoded 3-PSK.

Figure 4.12 shows the truncated upper bound and the simulation results for the 2 5/3 1 and the 2 5/11 1 codes. The 2 5/ 11 1 code has maximum $d^2_{prod}(l_{eff})$ but the 2 5/ 3 1 code performs better at low SNR. There are two possible reasons: one, the 2 5/3 1 code does have a larger d^2_{free} than the 2 5 /11 1 code, two, the 2 5/ 11 1 code has much smaller term for the first term of the error event with effective length of 4 than the 2 5/3 1 code. The terms are 0.268 for the 2 5/11 1 code and 2 for the 2 5 / 3 1 code and at low SNR these terms are dominant rather than the terms with the effective length of 3. The 2 5/ 11 1 code will perform better at high SNR, however, the 2 5/3 1 code is better at low SNR values. The simulation results shown in Figure 4.12 indicate that the codes perform similarly on the Rayleigh fading channel, with the 2 5/3 1 performing slightly better in the range simulated.

The 144-state $\mathbb{Z}_{12}$ codes 3 11 7/1 8 1 and 5 10 11/ 1 7 1 were found from a partial reduced search. The search algorithm did not search over all of the irreducible polynomials due to time constraints, however, a large number were searched and the best of the partial search are included here. The performance estimates for both codes indicate that the 3 11 7/1 8 1 has a 1.76 dB gain over the 5 10 11/ 1 7 1 code on the AWGN channel. The simulation results and the truncated upper bound are shown in Figure 4.13

As seen in the figure, the two codes perform similarly on the Rayleigh fading channel until about $E_b/N_0 = 15$ dB. For SNR greater than 15 dB the 5 10 11/1 7 1 code performs better than the 3 11 7/1 8 1 code. However, this is around an SER of 10^{-7}. Thus, for performance around 10^{-4} the 3 11 7 / 1 8 1 code is recommended. The simulation results do not show the cross-over in performance as the lower error rates are impractical to simulate.

4.7 Codes over $\mathbb{Z}_{16}$

The 3 9/14 1 code from Table 3.2 has a $l_{eff} = 3$ and a $d^2_{free} = 4.21$. In contrast, the 3 6/14 1 code presented in Table 3.11 has a $l_{eff} = 2$ and $d^2_{free} = 5.33$. Here it would be expected that the 3 6/14 1 code would perform better at low SNR, but worse at high SNR. Figure 4.14 shows simulation results and a truncated upper bound (includes 3

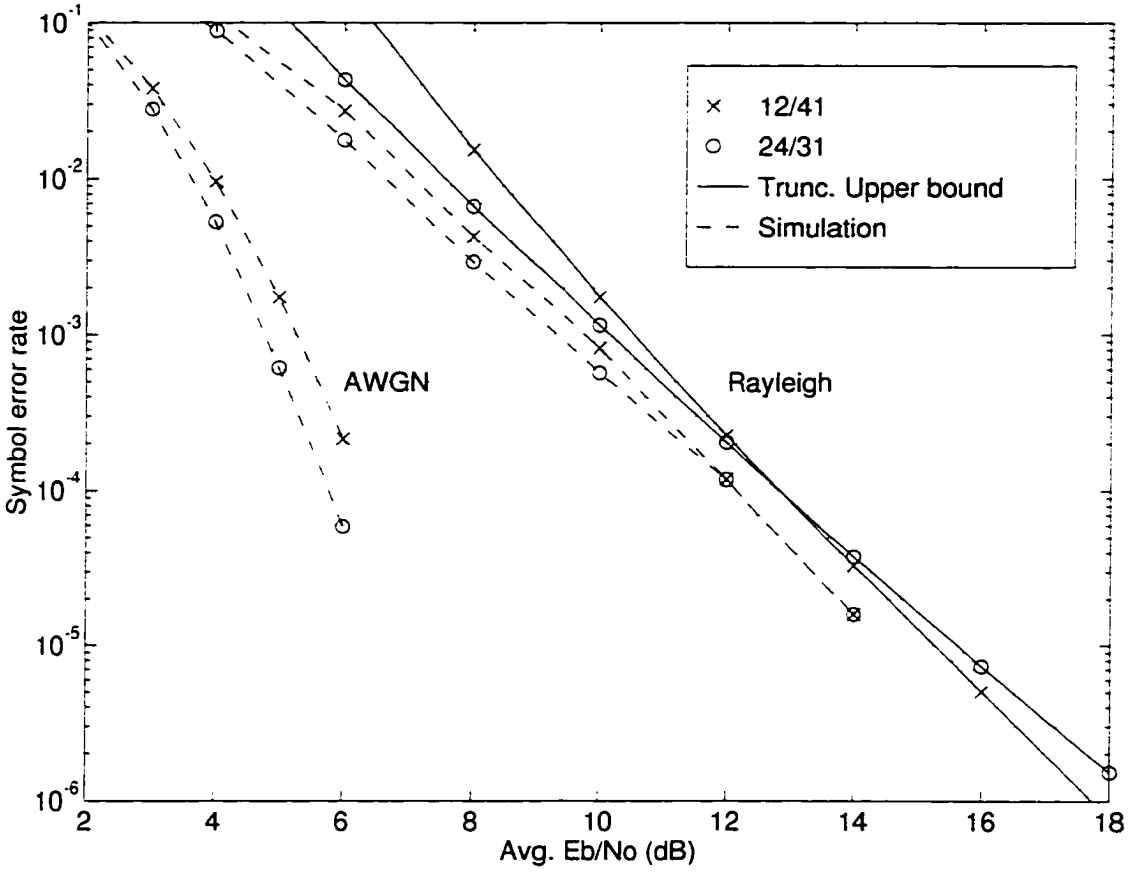


Figure 4.11. *Simulation and truncated upper bounds for codes over $\mathbb{Z}_9$ with 9 states.*

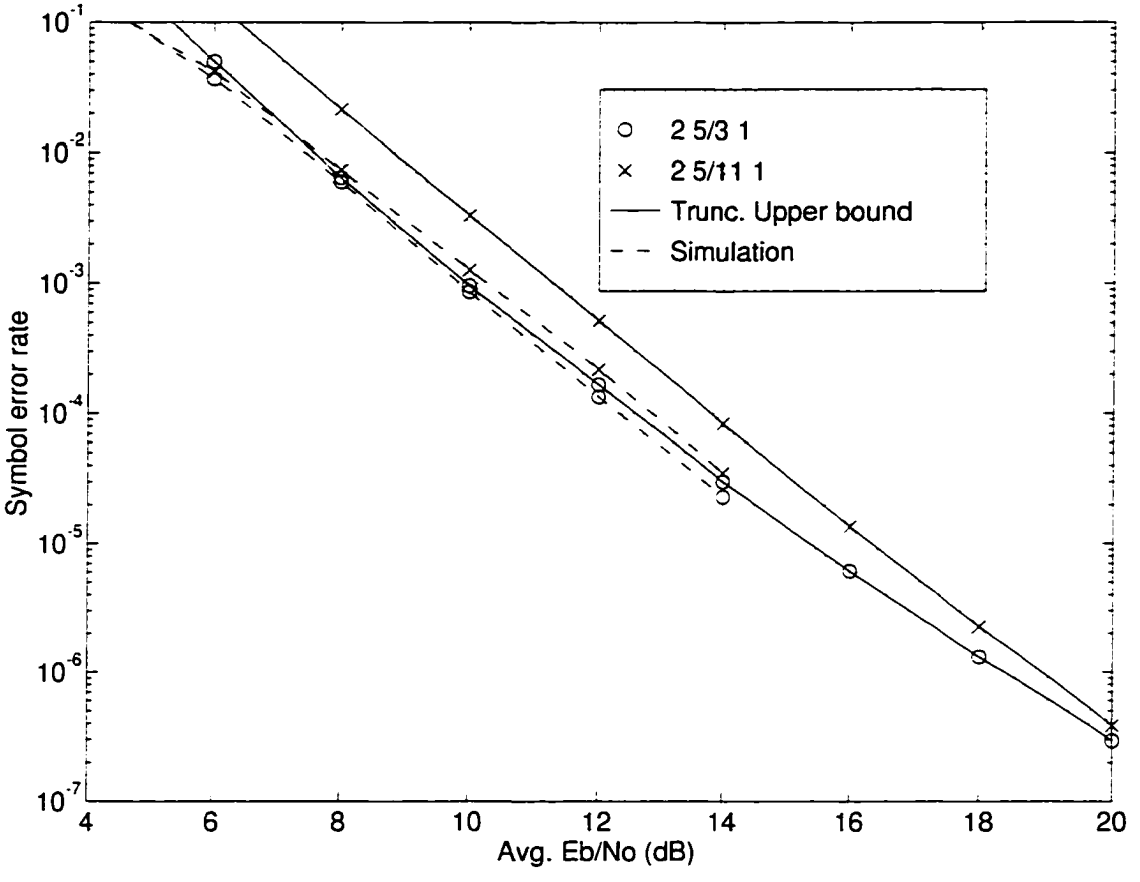


Figure 4.12. *Simulation and truncated upper bounds for codes over* $\mathbb{Z}_{12}$ *with 12 states.*

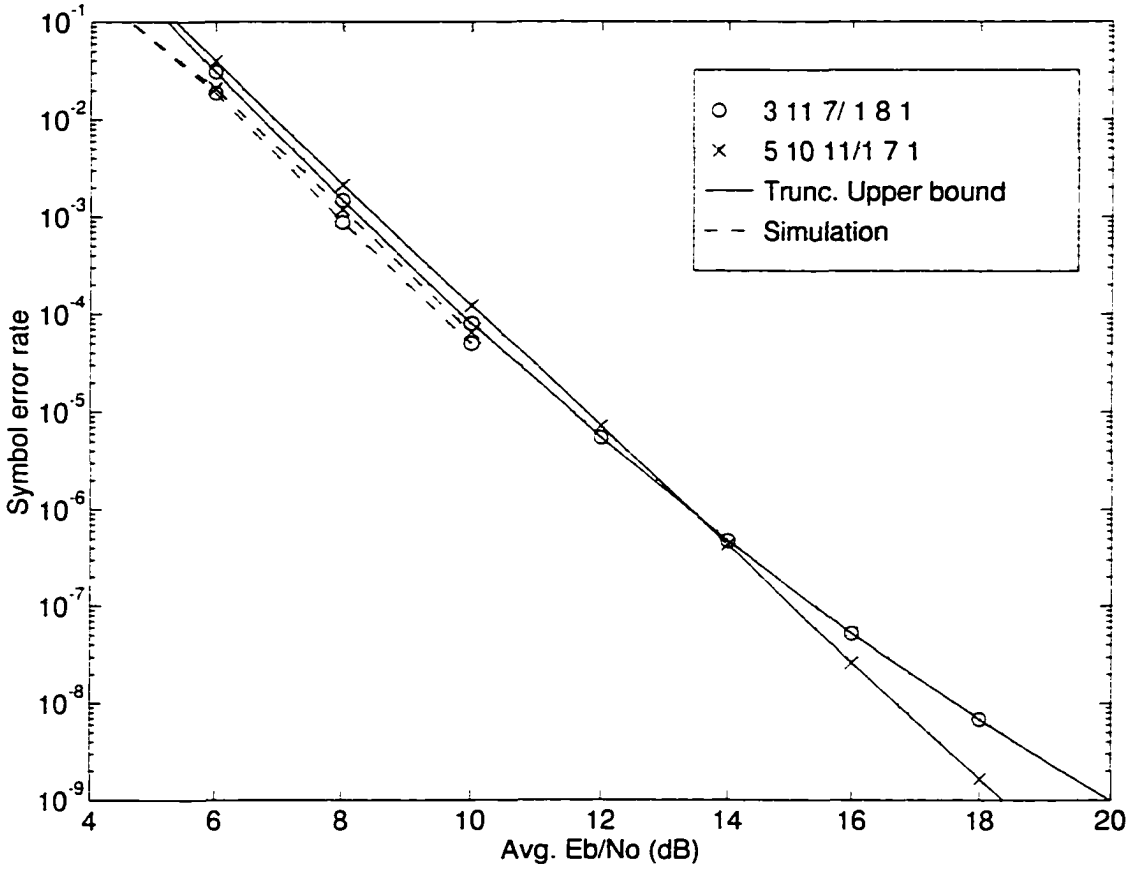


Figure 4.13. *Simulation and truncated upper bounds for codes over* $\mathbb{Z}_{12}$ *with 144 states.*

terms from transfer function) for both codes. The figure shows that the 3 6/14 1 code and the 3 9/14 1 code perform similarly at low SNR, however around $\bar{E}_b/N_0 = 10$ dB, the 3 9/14 1 code performs better as the other code begins to diverge. This would indicate that higher SNR the 3 9/14 1 code would perform much better on Rayleigh fading.

As uncoded QPSK has the same efficiency of 2 bits/symbol, we compare the performance of the above codes with that of uncoded QPSK. Both codes have a gain of 18.2 dB at 10^{-3} and the 5 7/14 1 and 3 6/14 1 codes have gains of 25.7 dB and 25 dB at 10^{-4}, respectively.

Comparing these codes with known codes, the 3 6/14 1 code has the same distance properties as the 13 6/2 1 found in [9]. The full search algorithm did not find the 13 6/2 1 as it only searched half of all possible codes as the other half would be equivalent to a code that had been searched. Here, 3 6/14 1 and 13 6/2 1 are equivalent codes. The latter code has the advantage of being rotationally invariant. This will be discussed in Chapter 5.

4.8 Summary

Tables 4.4 and 4.5, results are present a comparison and summary of the results for the fading and AWGN codes found. In the tables, one or two codes have been chosen to represent the performance of the fading codes which have maximum l_{eff} and d^2_{prod} and the AWGN codes which have maximum d^2_{free}. The number of delay elements, v and the number of states are shown and the fading and AWGN code polynomials are given. The effective length is shown for both codes. Also shown for both codes is the asymptotic gain, $\Delta_g\infty$, for AWGN and where applicable on the Rayleigh fading channel. $\Delta_g\infty$ is defined as the asymptotic gain of the fading code over the AWGN code on the AWGN channel. As can be seen when the codes differ, the gains are negative which indicates that the AWGN codes perform better on the Gaussian channel. This is an expected result since codes designed for fading channels should be sub-optimal on the AWGN channel. In most cases, the fading code loses less than 1 dB in performance, but has a greater l_{eff}. In the case where $\Delta_g\infty$ is shown for the Rayleigh case, the code with the maximum d^2_{free} has the same l_{eff} as the fading

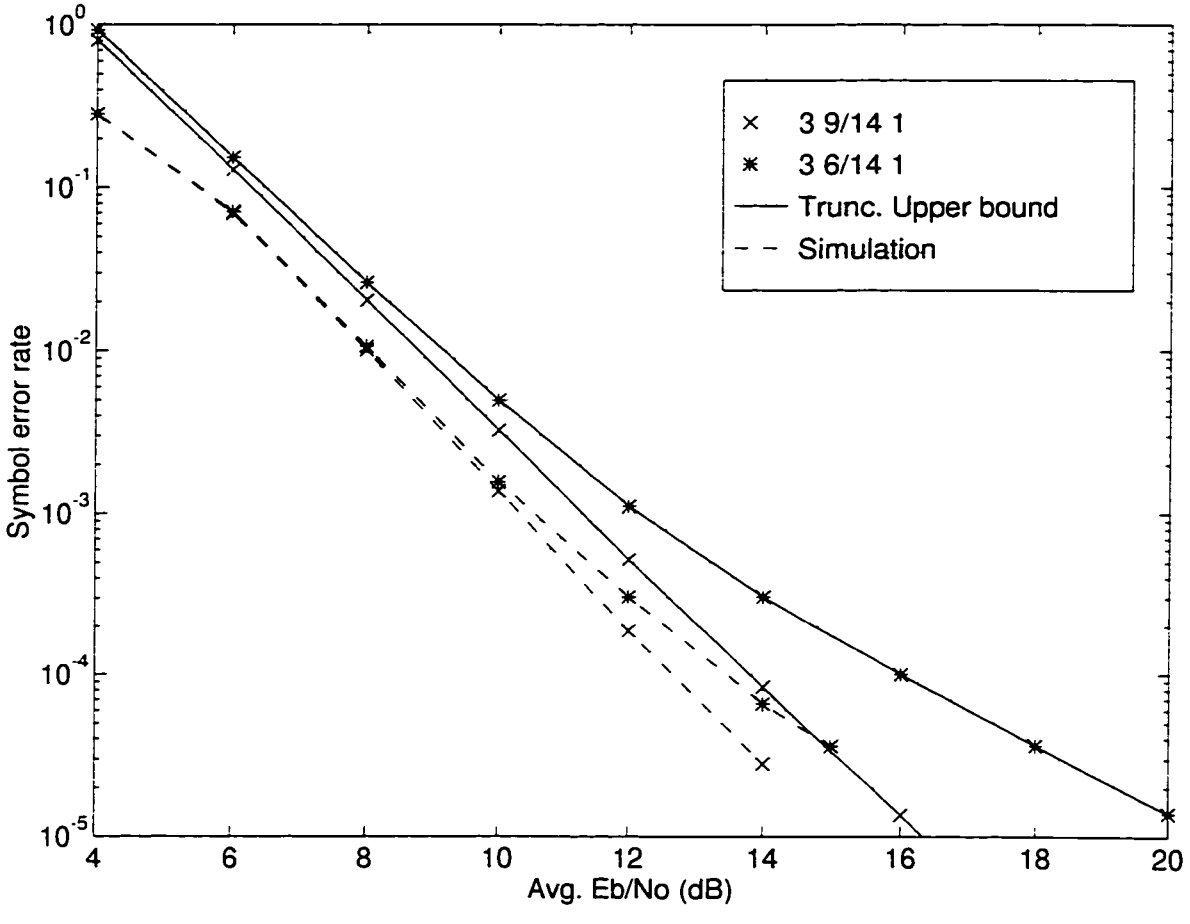


Figure 4.14. *Simulation and truncated upper bounds for codes over $\mathbb{Z}_{16}$ with 16 states. 3 9 /14 1 has maximum l_{eff} and 3 6/14 1 has maximum d^2_{free}.*

code but has a smaller d^2_{prod}. The gain is defined by Equation 4.1 and is only defined when the codes have the same effective length. When the fading code has a greater effective length than the AWGN code, the asymptotic error performance in fading will decrease faster with respect to SNR than the AWGN code. This is a result of the asymptotic error performance being proportional to $\mathrm{SNR}^{-l_{eff}}$. Therefore, a code with a larger l_{eff} is expected to perform better asymptotically on the fading channel.

Looking at several examples from the tables, the performance losses associated with using a fading code on the AWGN channel are generally less than 1 dB. However, in the case of $\mathbb{Z}_6$ there is a loss of 1.76 and 2.1 dB for the 6 and 36 state codes, respectively. In Figure 4.6, the code with the maximum d^2_{prod} performs slightly worse at a SER= 10^{-5} but is expected to peform better at higher SNR levels as the first term in the product transfer function becomes dominant. At lower SNR levels, Gaussian noise and other terms in the product function contribute to the error performance. In the case of the 36 state code, the code with the maximum l_{eff} and d^2_{prod} performs about 0.5 dB worse than the AWGN code at an SER of 10^{-3}. There is a cross-over around the 10^{-6} SER level. However, the code does not make up for the loss on the AWGN channel and does not approach the asymptotic performance quickly. Thus, it is recommended that the 214/131 code be used if the desired error rate is above 10^{-6}.

The 9 state $\mathbb{Z}_9$ code for the AWGN code also performs better than the fading code at low SNR. The simulation results show that the two codes perform identically for the 12 to 14 dB range on the Rayleigh fading channel. This agrees well with the truncated upper bound for the codes.

The codes presented over $\mathbb{Z}_{12}$ show that the maximum effective length and d^2_{prod} do not guarantee the best performance at low SNR. These terms dominate the asymptotic fading performance, but other terms will dominate at lower SNR values. Thus, it is important to consider many terms in the distance spectrum and the performance of the codes at the SER or BER of interest.

The 16 state $\mathbb{Z}_{16}$ fading code performed better than the AWGN code in fading. The performance difference is more pronounced at higher SNR levels. This is a good code as it loses only 0.1 dB on the AWGN channel and performs better than the AWGN code found with the maximum d^2_{free}.

From Tables 4.4 and 4.5, it is clear that in many cases the code with the maximum d^2_{free} also has the maximum d^2_{prod} and l_{eff}. When these codes are not the same, we find that in most cases, the effective length of the AWGN code is less than the code for fading. As a result, we have found codes which are expected to perform better on a fading channel than the codes optimized for the AWGN channel. The codes for fading will outperform the AWGN codes at high SNR on the fading channel and perform only slightly worse on the AWGN channel. In the case of $\mathbb{Z}_6$ codes the loss on the AWGN channel is above 1 dB and the AWGN code performs better than the fading code at low SNR on the fading channel.

Table 4.4. *Fading and AWGN comparison for $\mathbb{Z}_2$ to $\mathbb{Z}_8$ codes*

Ring	ν	states	Fading codes	l_{eff}	AWGN codes	l_{eff}	Δg_∞ dB AWGN	Δg_∞ dB Rayleigh
2	1	2	01/11	3	11/01	3		
	2	4	101/111	5	Same			
3	1	3	11/21	4	Same			
	2	9	111/121	6	Same			
	3	27	1021/2211	7	Same			
	4	81	10111/21121	9	N.A.			
4	1	4	11/21	3	Same			
	2	16	123/331	5	232/331	4	-0.58	
	3	64	1311/3321	6	Same			
	4	256	12214/11021	6	N.A.			
5	1	5	12/21	4	Same			
	2	25	213/431	6	Same			
	3	125	1213/2231	8	Same			
6	1	6	11/21	3	12/31	3	-1.76	1.25⋆
	2	36	115/541	5	214/131	4	-2.10	
	3	216	1154/5231	6	1252/3121	5	-0.54	
7	1	7	12/31	4	Same			
	2	49	145/531	6	235/651	6	-.22	0.85
	3	343	1325/2621	8	1654/5521	8	-.1	0.63
8	1	8	13/21	3	32/41	2	-.36	
	2	64	356/761	5	232/741	3	-.36	
	3	256	3655/7761	6	N.A.			

† AWGN code may perform better at low SNR

⋆ did not achieve this gain

N.A. Not Available

Table 4.5. *Fading and AWGN comparison for $\mathbb{Z}_9$ to $\mathbb{Z}_{16}$ codes*

Ring	v	states	Fading codes	l_{eff}	AWGN codes	l_{eff}	Δg_∞ (dB) AWGN	Δg_∞ (dB) Rayleigh
9	1	9	12/41	4	24/31	3	-1.1	
	2	81	145/211	6	262/551	5	-.2	
	2	81	142/241	6	262/551	5	-.8	
10	1	10	14/31	3	Same			
	2	100	367/991	5	257/981	4	-.26	.151†
11	1	11	53/41	4	Same			
12	1	12	2 5/11 1	3	25/31	3	-.5	.124†
	2	144	5 10 11/1 7 1⋆	5	3 11 7/1 8 1	4	-1.76	
13	1	13	2 4/5 1	4	35/21	4	-.27	.014†
14	1	14	3 5/ 12 1	3	1 8 /10 1	2	-.23	
15	1	15	2 11/ 8 1	4	2 9 / 9 1	2	-0.57	
16	1	16	3 9 / 14 1	3	3 6 14 1	3	-1.02	
			3 6 /2 1	2	3 6 14 1	3	0	0

† AWGN code may perform better at low SNR

⋆ Partial search result

4.9 Conclusion

At a high SNR, several of the codes designed for the Rayleigh fading channel outperform the codes designed for the AWGN channel. At low SNR, other terms in the Euclidean and product transfer functions affect the performance of the code and in some cases the code designed for the AWGN channel will perform slightly better than the fading code at low SNR.

The $\mathbb{Z}_4$ codes presented outperform comparable binary trellis codes designed for Rayleigh fading. Also, a 4-state code was found which improves on the performance of the 4-state code presented in Baldini and Farrell [3] in both AWGN and Rayleigh fading. The 16-state $\mathbb{Z}_4$ fading code is also better in fading than the 16-state $\mathbb{Z}_4$ code designed for AWGN. This AWGN code had the same d^2_{free} as the 16-state $\mathbb{Z}_4$ code presented in [3].

The simulations showed that for $\mathbb{Z}_6$ and $\mathbb{Z}_9$, the AWGN code performs better than the fading code at low SNR. The truncated upper bounds show that there is a cross-over in performance at higher SNR. For the cases of the 6-state $\mathbb{Z}_6$ and 9-state $\mathbb{Z}_9$ codes, the simulation results behave similar to the bounds and the fading and AWGN codes perform identically at the high SNR levels simulated. Although the AWGN code has an $l_{eff} = 5$ and the fading code has an $l_{eff} = 6$, the 216-state $\mathbb{Z}_6$ fading and AWGN codes perform identically on the Rayleigh fading channel at the error rates simulated. In this case, as well as in other cases when q is non-prime, there is only one path with l_{eff}, and other paths with longer lengths will dominate in the error performance at low SNR.

Similar results were found in a partial search for codes with constraint length 2 over $\mathbb{Z}_{12}$. For an SER of 10^{-4} the code with the maximum d^2_{free} is recommended and the fading code should be used at when the SNR greater than 15 dB.

The 16-state $\mathbb{Z}_{16}$ fading code found in the search performs better in fading than the best known rate 1/2 16-state code over $\mathbb{Z}_{16}$ [8].

Chapter 5

Rotational Invariance

Codes which are rotationally invariant (RI) are of great practical interest as a phase slip in the demodulator does not cause a large number of errors [2]. They are also practical because they eliminate the need to determine the absolute phase at the receiver [8]. For these reasons, it is beneficial to find codes over $\mathbb{Z}_q$ which are invariant to multiples of $2\pi/q$ phase rotations. These codes are referred to as *transparent codes* [2, 3, 25, 10] and can be used in conjunction with a differential encoder to eliminate errors caused by phase rotations [3].

In this chapter, we continue the search for codes which are good over fading channels with the additional criteria of rotationally invariance.

5.1 Background

In [2, 3] the definition of rotational invariance just a requirement that the all-one codeword exist within the code. The reason for this condition is a phase rotation of $2\pi r/q$ is equivalent to adding r times the all-one codeword to the transmitted codeword. If the all-one codeword is a codeword, then by linearity, the sum of the transmitted codeword and multiple of the all-one codeword is a valid codeword.

The all-one codeword can be defined as follows.

$$\begin{aligned} u(D) &= \frac{1}{1-D} \\ &= 1 + D + D^2 + \dots \ . \end{aligned} \tag{5.1}$$

If $x(D)$ and $u(D)$ are a codewords, then $(x(D) + ru(D)) \bmod q$ is also a codeword.

In Figure 5.1, a convolutional encoder with additional differential encoder and decoder processes is shown. To illustrate how the differential encoding eliminates

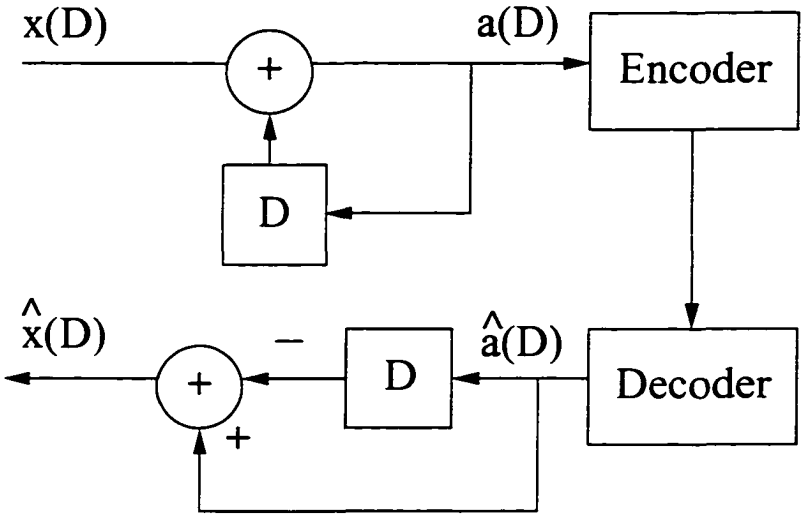


Figure 5.1. *Block diagram of transparent encoder/decoder*

the phase rotation, consider the transmitted codeword, $x(D)$. The sequence, $a(D)$, is formed by $a(D) = x(D) + a(D)D \bmod q$ and then encoded and transmitted. The channel introduces a phase rotation of $2\pi r/q$ and we obtain at the output of the decoder $\hat{a}(D)$. If all of the errors due to noise have been corrected then $\hat{a}(D)$ is just a rotated version of $a(D)$, i.e., $\hat{a}(D) = (a(D) + ru(D)) \bmod q$.

The output of the encoder is

$$\begin{aligned} \hat{x}(D) &= (1 - D)\hat{a}(D) \bmod q \\ &= (a(D) + ru(D)) - (a(D) + ru(D))D \bmod q \\ &= a(D) - a(D)D + r(u(D) - u(D)D) \bmod q \\ &= x(D) + r(u(D) - u(D)D) \bmod q. \end{aligned} \tag{5.2}$$

As $u(D)$ is the all-one sequence and subtracting a delayed version of it results in the zero sequence. This removes the phase rotation from the channel and results in $\hat{x}(D) = x(D)$

From [8] the definition for rotationally invariant (RI) is taken from trellis-coding for phase modulation. That is, the minimum phase shift when applied to all components starting at time 0 or later, yields a word that differs in at most a finite number of positions from another codeword.

From [8], a code is RI if for every codeword $x(D)$ there exists a polynomial $p_i(D)$

where $i = 1, 2, \ldots, n$ such that

$$y_i(D) = x_i(D) + \frac{1}{1-D} + p_i(D) \tag{5.3}$$

are components of another codeword $y(D)$.

Lemma [8]

A rational function $r(D) \in R(D)$ differs from $\frac{1}{1-D}$ by a polynomial iff

$$r(d) = \frac{p(D)}{1-D}. \tag{5.4}$$

where p(D) is a polynomial with $p(1) = 1$.

Here we modify Theorem 1 from [8] to the rate 1/2 case only by changing the notation to maintain consistency.

Theorem [8] Suppose the coefficients of the $(n-1) \times n$ systematic encoding matrix

$$G(D) = \begin{bmatrix} 1 & \cdots & 0 & \frac{f_1(D)}{g(D)} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & \frac{f_{n-1}(D)}{g(D)} \end{bmatrix} \tag{5.5}$$

satisfy the two conditions that

$$\sum_{j=1}^{n-1} f_j(1) = g(1) \tag{5.6}$$

and that there is at least one unit among the elements $g(1), f_1(1), \ldots, f_{n-1}(1)$. For rate 1/2 codes this gives us conditions on $f(D)$ and $g(D)$. i.e., $f(1) = g(1)$ and require it to be a unit then the code is RI.

We can "almost" generate the all-one codeword by choosing the input to be

$$u(D) = \frac{g(1)^{-1} g(D)}{1-D}. \tag{5.7}$$

Then

$$\begin{aligned} x(D) &= u(D)\frac{f(D)}{g(D)} \\ &= \frac{g(1)^{-1} f(D)}{1-D} \\ &= \frac{p(D)}{1-D}. \end{aligned} \tag{5.8}$$

The numerator polynomial $p(D)$ satisfies $p(1) = 1$ since $p(D) = g(1)^{-1}f(D)$ and $g(1) = f(1)$.

Since $x(D)$ in Equation 5.8 differs from the all-one sequence in finitely many places, the code is said to be RI.

Thus, in searching for the codes, we restrict the search to codes where $f(1) = g(1)$ and g(1) is a unit in $\mathbb{Z}_q$. Rather than searching for the all-one codeword, the above conditions on the code polynomials are used. These conditions allow the search routine to quickly eliminate codes which do not meet these criteria. As a result, the search routine can perform an exhaustive search through all possible codes.

As an example, we consider the convolutional code G(D) = [1 (1D+2)/(2D +1)] over $\mathbb{Z}_4$. From Equation 5.7 the input sequence for this code is

$$\begin{aligned} u(D) &= \frac{3+2D}{1-D} \\ &= 3 + D + D^2 + D^3 \ldots \ . \end{aligned} \tag{5.9}$$

When the input is encoded, the encoded sequence differs from the all-one sequence only in the first two positions. As an example of how the encoder works with a phase shift on the channel, we will consider an arbitrary symbol sequence as an input to the encoder as shown in Figure 5.1. In Table 5.1 the encoding, phase rotation and decoding process is shown. The input sequence, $x(D)$, is differentially encoded to form $a(D)$. This sequence is input into the convolutional encoder to form the encoded sequence. A phase shift of $\pi/4$ is added to the symbols at the beginning of the sequence. In Figure 5.1, the trellis diagram for the code is shown. Using this diagram we can decode the most likely sequence. The decoded state path is shown along with the estimate of $a(D)$. The estimate $\hat{a}(D)$ is then differentially decoded to form $\hat{x}(D)$, which is the estimate of the original symbol sequence. As seen in the table, the $x(D)$ and $\hat{x}(D)$ differ by only one symbol.

5.2 Search Results

The trellis search algorithm was carried out as the exhaustive search presented in Section 3.2. The differences between the searches were that for this one, the search space was not reduced by half and the tap polynomials were required to have $f(1) =$

Table 5.1. *Example of decoding with a phase shift for 12/21 code on* $\mathbb{Z}_4$

$x(D)$	1	2	3	2	2	0	1	2
$a(D)$	1	3	2	0	2	2	3	1
Encoded	12	33	21	00	20	22	30	11
$\pi/4$ phase shift	23	00	32	11	31	33	01	22
Decoded state path	3	0	0	3	3	1	1	2
$\hat{a}(D)$	2	0	3	1	3	3	0	2
$\hat{x}(D)$	2	2	3	2	2	0	1	2
Error	1	0	0	0	0	0	0	0

$g(1)$ and $f(1)$ a unit. The tap polynomials were checked for this condition before the search of the trellis started. If the code did not meet these conditions, the code was discarded and the algorithm tested the the next code.

Results for the rotational invariant (RI) unit memory codes are presented in Table 5.2. The effective length of the RI codes is equal to or less than non-RI unit memory codes presented in Chapter 3. When q is prime, the effective length of the codes is equal to length of the non-RI codes and in some cases they are the same code. The only exception is for $q = 3$. In this case, the RI code has an l_{eff} one less than the non-RI code. This is similar to the case when q is non-prime. The effective length of the code is one less than the maximum effective length of the non-RI code.

In some cases, the RI code has a greater d^2_{free} than the fading codes. However, the squared free distance is upper bounded by the squared free distance of the non-RI AWGN codes that were presented in Tables 3.10 and 3.11.

The above results are to be expected as the RI code search had additional constraints placed on the tap polynomials from the non-RI code search. The codes in the non-RI search were maximized for either the effective length and squared product distance for fading or the squared free distance for the AWGN channel. As such, the squared free distance of the RI-codes is upper bounded by the squared free distance of the non-RI AWGN codes and the effective length is similarly upper bounded by

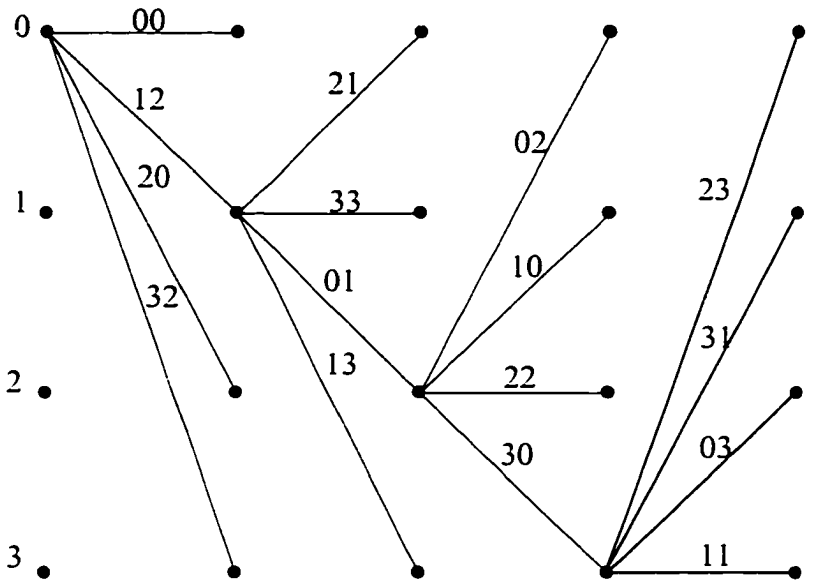


Figure 5.2. *Trellis definition for 12/21 code over* $\mathbb{Z}_4$

the non-RI fading codes effective length.

The results for the constraint length 2 RI-codes are similar to the unit memory code case. When q is prime, the effective length is equal to the non-RI codes, but when non-prime the effective length of the code is one less than the maximum effective length for non-RI codes. These codes are presented in Table 5.3.

The RI codes with maximum squared free distance are presented in Table 5.4. For $q < 7$ the codes in Table 5.3 also had maximum d^2_{free}. For $q = 7$ the AWGN code has an asymptotic coding gain of 0.28 dB and an asymptotic loss of 0.19 dB on the Rayleigh fading channel by Equation 4.1. As can be seen, the performance difference between the fading and the AWGN codes is very small. Similarly, for the case when $q = 9$ the AWGN coding gain is 0.81 dB. However, the l_{eff} is 4 where as the l_{eff} of the fading code is 6. Thus, the fading code is expected to perform better in fading.

TTable 5.5 presents the results for rotationally invariant codes with constraint length 3. These codes have the same effective length and d^2_{free} as the non-RI codes. Also, the codes for $\mathbb{Z}_3$ and $\mathbb{Z}_6$ have the same characteristics as the non-RI codes presented in Table 3.8. The d^2_{free} over $\mathbb{Z}_4$ and $\mathbb{Z}_5$ is less than that for the non-RI codes. This difference results in 0.51 dB and a 0.21 dB loss, respectively, on the AWGN channel.

In Table 5.7 we show the comparison between the non-RI codes from Chapter 3 and the RI codes presented in this chapter. We compare the effective length and the asymptotic gain over BPSK on the AWGN channel. The table presents the codes for constraint lengths of 1, 2, and 3, and codes over $\mathbb{Z}_3$ to $\mathbb{Z}_{16}$. As can be seen the effective length, l_{eff} of the RI codes are upper bounded by the effective lengths of the non-RI codes. In the case when the effective lengths are identical the non-RI code has a greater d^2_{free} than the RI code. For those cases when the effective length of the RI code is less than that of the non-RI code, the d^2_{free} of the RI code is greater than the RI code. For example, for the $\mathbb{Z}_8$ code with constraint length 1, the asymptotic gain over BPSK is 0.37 dB greater for the RI code than the non-RI code although the effective length is one less than the non-RI code.

In comparson with the literature, this search found several codes which were previously presented. For example, the $\mathbb{Z}_{16}$ 16-state code 136/21 was also found in Massey et al. [8]. For $\mathbb{Z}_4$ codes with 4 and 16 states, 12/21 and 311/221, presented in Tables 5.2 and 5.3 are previously known from Baldini et al. [3]. We also present two additional 16-state $\mathbb{Z}_4$ codes, namely, 113/221 and 122/311 which have the same characteristics as the 311/211 code.

The code 311/211 has $d^2_{free} = 12$ and the first four terms are shown in Table 5.6. Also shown in the table is a code that has $d^2_{free} = 16$ and the same d^2_{prod} as the 311/221 code. The 212/311 code does not have an error path of length 5 in the product transfer function like the 311/221 code, thus it is expected that it would perform better than the 311/221 code. However, the number of errors associated with each error path is greater than the 311/221 code and thus may perform worse.

Monte-Carlo simulations were used to confirm the relative performances of the codes. In Figure 5.2, simulation results are shown for the 212/131 and 311/221 codes. The first two paths of the transfer function were used to generate an upper bound on the performance on the Rayleigh fading channel. From the bound the 311/221 code should perform slightly better than the 212/131 code, and the simulation results the confirm that the difference between these two codes is extremely small. For the AWGN channel, the 212/131 code performs slightly better than the 311/221 code, as expected. For the symbol error range presented in the figure, the codes perform practically the same.

Table 5.2. *RI codes with constraint length 1 for Rayleigh fading*

Ring	polynomials	n_{free}	d^2_{free}	n_e	n_{prod}	d^2_{prod}	n_p	l_{eff}	g_∞ dB
3	02/11.20/11	2	9.00	2	2	27	2	3	2.51
4	12/21	1	8.00	1	1	16	1	2	3.01
5	12/21.13/31	4	10.00	2	4	25	2	4	4.63
6	32/41	2	8.00	2	1	16	1	2	4.12
7	23/41.33/51 64/21	2	7.75	2	2	5.2711	2	4	4.35
8	32/41.74/21	2	7.17	2	1	16	1	2	4.30
9	25/61	2	6.77	1	2	27	1	3	4.29
10	3 4/ 6 1 7 2/ 8 1 9 4/ 2 1 (g)	2	5.76	2	1	16	1	2	3.79
11	3 4/ 6 1 8 6/ 2 1	2	6.37	2	2	2.8068	2	4	4.40
12	3 8/10 1	1	5.54	3	1	16	1	2	3.95
13	2 4/ 5 1 6 3/ 8 1	1	5.96	3	2	1.3722	2	4	4.27
14	13 6/ 4 1	2	6.06	2	1	16	1	2	4.60
15	11 8/ 3 1	2	4.43	2	2	27	1	3	3.35
	13 9/ 6 1	2	5.63	2	2	9	1	2	
16	13 6/ 2 1 5 10/14 1 (g)	1	5.33	4	1	16	1	2	4.26

Table 5.3. *RI codes with constraint length 2 for Rayleigh fading*

Ring	polynomials	n_{free}	d^2_{free}	n_e	n_{prod}	d^2_{prod}	n_p	l_{eff}	g_∞ dB
2	001/111 010/111 100/111	1	16.00	2	1	256	2	4	3.01
3	112/211.122/221	6	18.00	3	6	729	3	6	5.52
4	113/221.122/311 311/221	2	12.00	1	2	256	1.5	4	4.77
5	133/241.142/331 432/211. 443/321	4	14.15	3.5	4	125	3	6	6.14
6	232/511.434/551 452/131. 254/131	10	13.00	3.4	2	256	2.5	4	6.23
7	453/651. 423/621 125/341. 435/561 453/651	2	11.10	3	18	49	3	6	5.91
8	335/641. 533/641 342/711.472/751 546/771.614/731 614/731.645/771 243/771	2	11.17	3.25	2	256	1.5	4	6.22
9	125/521. 142/241 415/271.452/841 152/251.175/571 745/871.752/581	2	8.48	4	6	9	3	6	5.26

Table 5.4. *RI codes with constraint length 2 for the AWGN channel*

Ring	polynomials	n_{free}	d^2_{free}	n_e	n_{prod}	d^2_{prod}	n_p	l_{eff}	g_∞ dB
7	134/251.254/461	2	11.85	4	4	15.09	3	6	6.19
	152/431.335/211								
	553/411.633/221								
	432/261.655/441								
8	335/641.533/641	8	11.17	3.2	2	256	1.5	4	6.22
9	154/361.163/451	2	10.24	1	1	81	3	4	6.08
	164/371.173/461								
	346/481.347/761								
	367/781. 373/841								
	613/721								

Table 5.5. *RI codes with constraint length 3 for Rayleigh fading*

Ring	polynomials	n_{free}	d^2_{free}	n_e	n_{prod}	d^2_{prod}	n_p	l_{eff}	g_∞ dB
3	1112/1211	4	21	3.5	4	2187	3.5	7	6.19
	1121/2111								
	2111/1121								
	2122/2221								
4	1121/3231	2	16.00	4	2	4096	6	6	6.02
	1323/1211								
	3231/1121								
	3233/3121								
5	4142/3431	4	16.91	4.5	8	238.7	3.7	8	6.91
6	1145/2351	6	15.00	4.2	5	4096	3	6	6.85

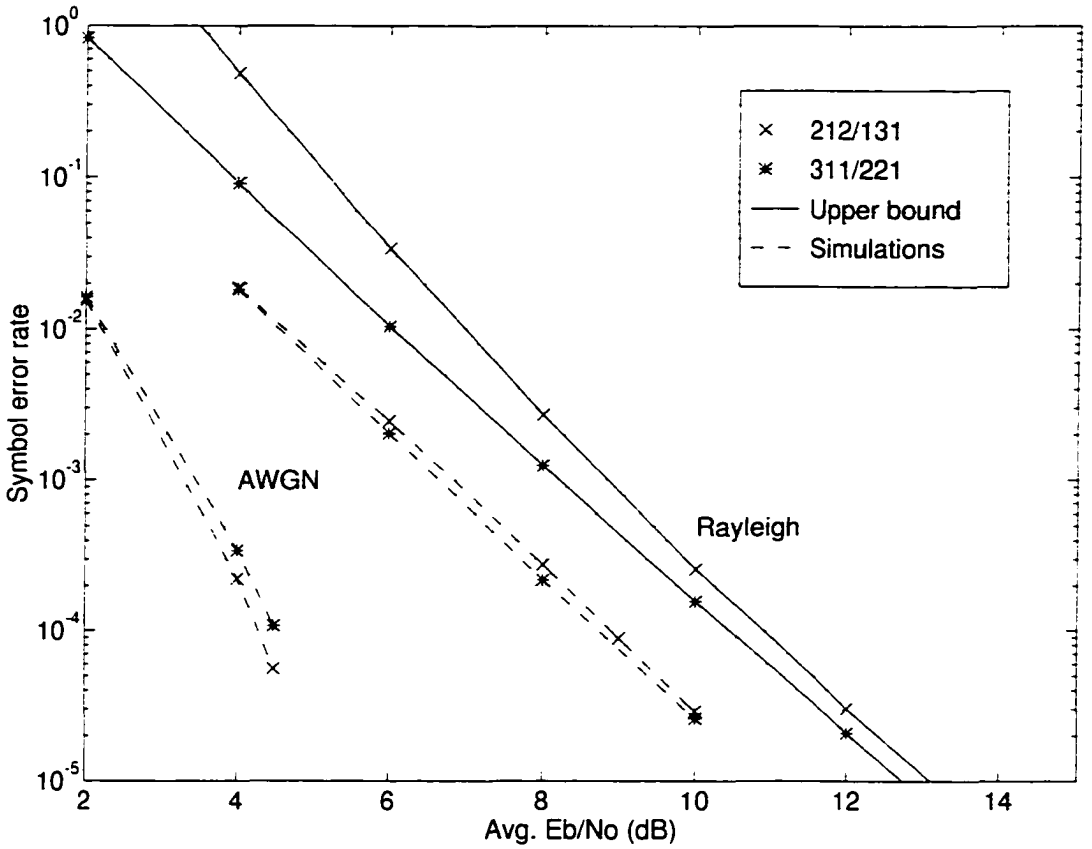


Figure 5.3. *Performance comparison for the 16-state $\mathbb{Z}_4$ codes 212/131 and 311/211 on the Rayleigh and AWGN channel.*

Table 5.6. *Transfer functions for the 212/131 and 311/221 code*

212/131						
Euclidean			Product			
n_e	d_e^2	errors	n_p	d_p^2	errors	l
14	16	3.5	2	256	2.5	4
46	20	4.3	12	256	3.7	6
267	24	5.1	5	4096	3	6
848	28	5.6	20	1024	4	7
311/221 code						
Euclidean			Product			
n_e	d_e^2	errors	n_p	d_p^2	errors	l
2	12	1	2	256	1.5	4
18	16	2.8	2	64	1	5
80	20	3.3	10	256	2.8	6
317	24	4.5	5	4096	3	6

5.3 Summary

The restrictions on the tap polynomials for the rotationally invariant codes were described. The results of an exhaustive search for RI codes were presented. The search found a two codes which have appeared in the literature, namely, 12/21 over $\mathbb{Z}_4$ [3] and 13 6/ 2 1 over $\mathbb{Z}_{16}$ [8]. As we used the results from [8] to restrict the tap polynomials, we would expect to find the same rate 1/2 code over $\mathbb{Z}_{16}$ as they did in their search.

In a few cases, the RI code has the best l_{eff} and d_{prod}^2 for the ring. For example, the constraint length 2 and 3 codes for $\mathbb{Z}_3$ have the same characteristics as the best codes found in Chapter 3.

The effective lengths of the RI codes are upper bounded by the effective lengths of the non-RI codes. However, there were a few cases where the d_{free}^2 was greater than the non-RI code when the effective length was less than the non-RI code. For example, for the constraint length 1 codes, the codes for $\mathbb{Z}_q$ where q is 6, 8, 9, 12, 14 and 16 all have better d_{free}^2 and the l_{eff} is one less than the non-RI codes. This means

Table 5.7. *RI comparison with non-RI ring codes*

Ring	$v = 1$				$v = 2$				$v = 3$			
	Non-RI		RI		Non-RI		RI		Non-RI		RI	
	g_∞	l_{eff}	g_∞	l_{eff}	g_∞	l_{eff}	g_∞	l_{eff}	g_∞	l_{eff}	g_∞	l_{eff}
3	3.76	4	2.51	3	5.52	6	5.52	6	6.19	7	6.19	7
4	3.98	3	3.01	2	5.44	5	4.77	4	6.54	6	6.02	6
5	4.63	4	4.63	4	6.39	6	6.14	6	7.12	8	6.91	8
6	2.88	3	4.12	2	4.12	5	6.23	4	6.85	6	6.85	6
6^*	4.64	3	4.12	2								
7	4.35	4	4.35	4	6.13	6	5.91	6				
8	3.93	3	4.30	2	5.98	5	6.22	4				
9	3.12	4	4.29	3	6.79	4	5.29	3				
10	4.63	3	3.79	2	7.92	5						
11	4.61	4	4.40	4								
12	3.73	3	3.95	2	4.29	5						
13	4.40	4	4.27	4								
14	4.37	3	4.60	2								
15	4.0	4	3.35	3								
16	3.23	3	4.26	2								

that the codes will perform worse asymptotically in fading than the non-RI codes, but are better on the AWGN channel.

In most cases the RI code with the added restrictions are worse on the Rayleigh fading channel and the AWGN channel. However, these codes are still of interest due to the rotational invariance property. If the channel has slow phase rotations then the RI codes will perform much better than the non-RI codes.

Chapter 6

Summary of Results and Suggestions for Future Work

Chapter 1 presented a historical background of ring codes and some of the early work which led up to this dissertation.

In Chapter 2, fundamentals of the digital communication systems were presented. The basic structure of the convolutional codes along with the PSK modulation were presented. As well, the asymptotic estimate of the codes' performance was introduced and defined the characteristics to be used in the search for good codes.

Chapter 3 described the exhaustive search routine and the results of the search. The restrictions on the tap polynomial which led to the reduced search algorithm were developed. The codes found by the reduced search were also presented. Although this dissertation was mainly concerned with maximizing the fading performance by maximizing the effective length and squared product distance, codes which maximized only the squared free distance were also included. These codes are called AWGN codes as their performance is optimized for the AWGN channel. As the performance of the codes on the Rayleigh fading channel at low SNR is affected by the d_{free}, these codes are expected to perform well at low SNR, as well as, on the AWGN channel.

Simulation results for several codes were presented in Chapter 4. The codes were simulated over a Rayleigh channel with ideal interleaving and ideal channel state information. This was to confirm the expected performance and compare the codes with known codes from the literature. In this chapter, we found that several codes have better performance on the Rayleigh fading channel than known codes. Also, in one case the code had better performance on the AWGN channel than codes in the literature. Several comparisons were done between the code optimized for fading and

the codes optimized for the AWGN channel. In many cases, it was found that at low SNR, the AWGN codes performed slightly better than the fading codes. The fading codes were better asymptotically on the fading channel, however, in some cases the error rate was too low to simulate the system performance accurately in a reasonable amount of time.

Rotationally invariant codes were introduced in Chapter 5. Massey et al. [8] developed restrictions on the tap polynomials which could be used to find rotationally invariant (RI) codes. These codes are valuable as a phase rotation which is a multiple of $2\pi/q$ when using q-PSK will produce a small number of errors. A non-RI code would produce errors at the output of the receiver until the rotation was corrected. As the restrictions on the code polynomials allowed the search algorithm to eliminate codes quickly, an exhaustive search was carried out. The results of the search for codes over $\mathbb{Z}_q$, $q \in \{3, \ldots, 16\}$, were presented in Chapter 5. In the case of the 4-state $\mathbb{Z}_4$ and 16-state $\mathbb{Z}_{16}$ codes, the search found codes also found by exhaustive search by Baldini and Farrell [2] and Massey et al. [8], respectively. As in Chapter 2, the fading codes as well as AWGN codes are included in the results of this chapter.

6.1 Future Work

The continuation of this research would include a continuation of the search for good ring codes. Rate 1/2 codes with higher constraint lengths would be of interest. Also, higher rate codes such as rate=2/3 codes and rate=3/4 codes would be of interest due to the increased efficiency. The general structure of a rate 2/3 encoder is shown in Figure 6.1. Baldini and Farrell [3] examined codes of both rate 2/3 over $\mathbb{Z}_8$, and rate 3/4 over $\mathbb{Z}_{16}$ and Massey et al. [8] have examined codes of rate=2/3 over $\mathbb{Z}_8$. In the case of codes that must work in fading environments, these codes should not have parallel transitions. Thus, all input bits must affect the state of the encoder to ensure the code does not have parallel transitions in the trellis.

One of the topics in coding theory that has recently received much interest is Turbo-codes. These codes were introduced in Berrou et al. [47] and are also called Parallel Concatenated Codes (PCC) [49]. A common structure for the Turbo-codes uses multiple Recursive Systematic Convolutional (RSC) codes as component codes.

One structure is presented in Figure 6.2 although several variations appear in the literature. Examples of the structures of turbo encoders are presented in [47, 48, 49]. Another common structure deletes the bit x' from the output to obtained a rate 1/3 code. Significant coding gains have been obtained utilizing this structure.

A continuation of this research would be to use some of the codes found in this dissertation as the component codes in a Turbo-code. It is likely that the ring codes which showed gains over binary trellis codes with the same number of states would also achieve gains when used in a Turbo-code structure. One of the goals when this research started was to develop a turbo-ring-coder over an arbitrary integer ring modulo-q. However, sufficient information on good codes for fading and over several of the integer rings was not available. It is hoped that this research will lead to the use of some of the codes herein as component codes in a Turbo-coded system.

The 4-phase spreading sequences presented in Boztaş et al. [40] and Hammons et al. [45] are applicable to Code Division Multiple Access (CDMA) Spread Spectrum systems. An interesting study would be to combine the codes over $\mathbb{Z}_4$ and the 4-phase spreading sequences in a CDMA system. Another study of interest is to use codes over $\mathbb{Z}_q$ and q-ary spreading sequences similar to those in Boztaş et al. [40] to form a q-ary coding/q-ary spreading communication system. For example, the 12/31 code over $\mathbb{Z}_6$ had gain over BPSK of 4.64 dB could be spread using a 6-ary spreading sequence. A 6-ary spreading sequence could be generated by a shift register utilizing an irreducible polynomial from Appendix A. An investigation to find the polynomials which generate sequences with low cross-correlation would be required.

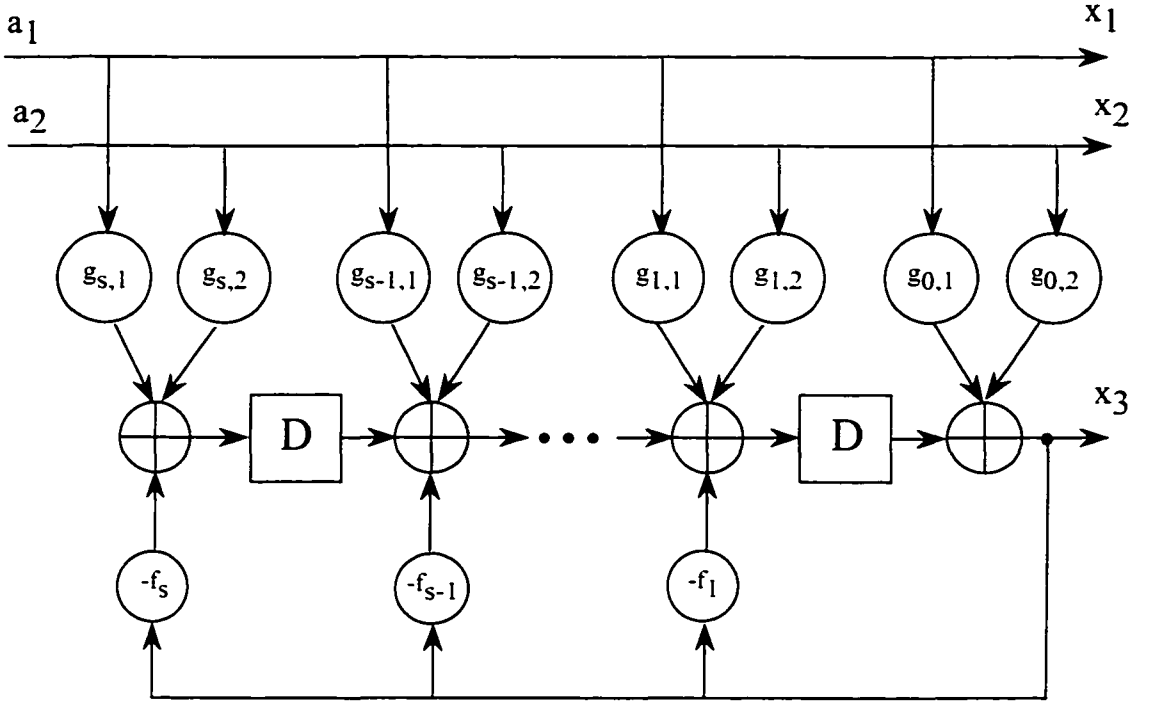


Figure 6.1. *Structure of a rate 2/3*

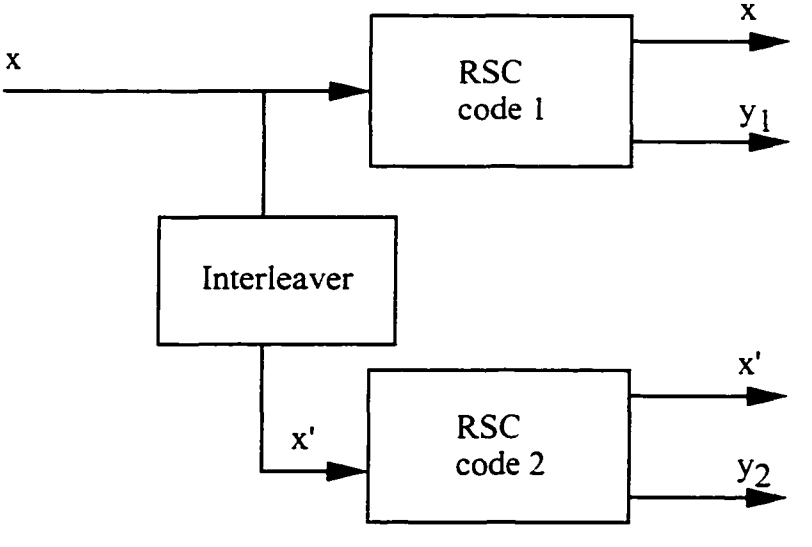


Figure 6.2. *An example of a rate 1/4 Turbo code structure*

Bibliography

[1] C.E. Shannon. "A Mathematical Theory of Communications." *Bell Syst. Tech. J.*, vol. 27, pp. 379-423,623-656, 1948.

[2] R. Baldini F. and P.G. Farrel. "Coded modulation based on rings of integers modulo-q. Part 1: Block codes." *IEE Proc.-Commun.* vol. 141, no. 3, June 1994, pp. 129-136.

[3] R. Baldini F. and P.G. Farrel. "Coded modulation based on rings of integers modulo-q. Part 2: Convolutional codes." *IEE Proc.-Commun.* vol. 141, no. 3, June 1994, pp. 137-142.

[4] R. Baldini F. and P.G. Farrel. "Coded modulation with convolutional codes over rings." Presented at EUROCODE'90, Udine Italy, 1990.

[5] J.L. Massey and T. Mittelhozer. "Convolutional codes over ring." Presented at 4th Joint Swedish-Soviet Int. Workshop on Information Theory, Gotland Sweden, Aug. 1989.

[6] T. Mittelholzer. "Minimal Encoders for Convolutional Codes over Rings." in *Communications Theory and Applications*, B. Honary, M. Darnell, and P.G. Farrell (ed.), pp. 30-36, HW Comm Ltd., 1993.

[7] J.L. Massey, T. Mittelhozer, T. Riedel and M. Vollenweider. "Ring Convolutional Codes for Phase Modulation." Presented at ISIT'90, San Diego, CA, Jan. 1990.

[8] J.L. Massey, T. Mittelhozer and F. Tarköy "New Modulation/Coding Techniques for a One-User Satellite Channel." Final Report, ESTEC Contract no. 8696/89/NL/US Technical Assistance for the CDMA Communications System Analysis, Nov. 1994.

[9] J.L. Massey and T. Mittelhozer. "Systematicity and Rotational invariance of convolutional codes over rings." *Proc. 2nd Int. Workshop on Algebraic Algebraic and Combinatorial Coding Theory*, Sept. 16-22, 1990, Leningrad, pp. 154-158.

[10] J.L. Massey and T. Mittelhozer "Codes over rings - a practical necessity." *Presented at AAECC Symposium*, Toulouse France, June 1989.

[11] T. Mittelholzer, J.L. Massey, "Convolutional Codes and Cohomological Invariants." *Abstracts of papers Workshop on Applications of Algebraic Geometry*, Jan. 8 - 12, 1990, Univ. of Puerto Rico, Puerto Rico.

[12] I.F. Blake. "Codes over Certain Rings." Inf. and Control. vol. 20. no. 4. pp. 396-404. May 1972.

[13] I.F. Blake. "Codes over Integer Residue Rings." Inf. and Control. vol. 29. no. 4. pp. 295-300. Dec. 1975.

[14] B. Rimoldi and Q. Li. "Coded Continuous Phase Modulation Using Ring Convolutional Codes." *IEEE Trans. Commun.*, vol. 43. no. 11. pp. 2714-2720. Nov. 1995.

[15] R.H.-H. Yang and D.P. Taylor. "Trellis coded continuous phase frequency shift keying with ring convolutional codes." *IEEE Trans. Inform. Theory*. vol. 40. pp. 1057-1067. July 1994.

[16] G. Karam. K. Gosse. and K. Maalej. "Trellis coded CPFSK over rings." *Proc. of ICC'95*. Seattle. Wa.. June 18-22. 1995. pp. 673-677.

[17] G. Ungerboeck. "Channel Coding with Multilevel/Phase Signals." *IEEE Trans. Inform. Theory*. vol. IT-28. no. 1. pp. 56-66. Jan. 1982.

[18] G. Ungerboeck. "Trellis Coded Modulation with Redundant Signal Sets Part I: Introduction." *IEEE Comm. Magazine*. vol. 25. no. 2. Februrary 1987. pp. 5-11.

[19] G. Ungerboeck. "Trellis Coded Modulation with Redundant Signal Sets Part II: State of the Art." *IEEE Comm. Magazine*. vol. 25. no. 2. Februrary 1987. pp. 12-21.

[20] D. Divsalar and M.K. Simon. "The Design of Trellis Coded MPSK for Fading Channels: Performance Criteria." *IEEE Trans. Commun.*, vol. 36. no. 8. pp. 1004-1012. Sept. 1988.

[21] D. Divsalar and M.K. Simon. "Trellis Coded Modulation for 4800-9600 bits/s Transmission over a Fading Mobile Satellite Channel." *IEEE J. Select. Areas Commun.*, vol. SAC-5. no. 2. pp. 162-175. Feb. 1987.

[22] A. Lee. "Analysis of Amplitude. Phase and Error Distributions for Shadowed Mobile Satellite Communications Channels." M.Sc. Thesis. Queen's University at Kingston. Jan. 1988.

[23] S. Wilson. Y.S. Leung. "Trellis-Coded Phase Modulation on Rayleigh Channels." *Proc. of ICC'87*. Seattle. Wa.. June 1987. pp. 21.3.1-5.

[24] C. Schlegel and D.J. Costello. "Bandwidth efficient coding for fading channels: code construction and performance analysis." *IEEE J. Select. Areas Commun.*, vol. 7 no. 9. pp. 1356-1368. Dec. 1989.

[25] T. Kasami. T. Takata. T. Fujiwara and S. Lin. "On linear structure and phase rotation invariant properties of block M-PSK modulation codes".*IEEE Trans. Inform. Theory*. vol. IT-37. 1991. pp. 164-166

[26] D.J. Rhee. S. Rajpal and S. Lin. "Some Block- and Trellis-Coded Modulations for the Rayleigh Fading Channel." *IEEE Trans. Commun.*, vol. 44. no. 1. pp. 34-42. Jan. 1996.

[27] L.H. Zetterberg. "A Class of Trellis Codes for Phase Moduation Based on Geometrical Design and Coset Mapping." Presented at 4th Joint Swedish-Soviet Int. Workshop on Information Theory. Gotland Sweden. Aug. 1989. pp. 23-25.

[28] C.-J. Chen. T.-Y. Chen and H.-A. Loeliger. "Constuction of Linear Ring Codes for 6 PSK." *IEEE Trans. Inform. Theory.* vol. 40. no. 2. pp. 563-566. Mar. 1994.

[29] R. De Gaudenzi. F. Giannetti and M. Luise. "Advances in Satellite CDMA Transmission for Mobile and Personal Communications." *Proc. of the IEEE.* vol. 84. no. 1. Jan. 1996. pp. 18-39.

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

[31] J.K. Cavers and P. Ho. "Analysis of the error performance of trellis coded modulations in Rayleigh-fading channels." *IEEE Trans. Commun.*, vol. COM-40. pp. 74-83. Jan. 1992.

[32] S.H. Jamali and T. Le-Ngoc. *Coded-Modulation Techniques for Fading Channels.* Kluwer Academic Publishers. 1994.

[33] C. Tellambura. "Performance Analysis of Trellis Codes Transmitted over Fading Channels.". Ph.D. Thesis. University of Victoria. Victoria. B.C.. Canada. 1993

[34] D.H. Saracino. *Abstract Algebra: A First Course.* Addison-Wesley Publishing Company. Inc. Philippines. 1980.

[35] V.K. Bhargava. D. Haccoun. R. Matyas and P. Nuspl. *Digital Communications by Satellite.* Wiley Interscience. New York. 1981.

[36] S. Wicker. and V.K. Bhargava. *Reed-Solomon Codes and Their Applications.* IEEE Press. New York. 1994.

[37] J.G. Proakis. *Digital Communications. Third Edition* New York: McGraw-Hill. 1995.

[38] G. Clark and J.B. Cain. *Error-Correction coding for Digitial communications.* New York: Plenum. 1981.

[39] M. Vogel. "Performance Analysis of Interleaving." M.A.Sc. Thesis. University of Victoria. British Columbia. Canada. 1993.

[40] S. Boztaş. R. Hammons and P.V. Kumar. "4-Phase Sequences with Near-Optimum Correlation Properties." *IEEE Trans. Inform. Theory.* vol. 38. no. 3. May 1992. pp.1101-1113.

[41] P. Solé. "A Quarternary Cyclic Code and a Family of Quadriphase Sequences

with Low Correlation Properties." *Lecture Notes in Computer Science* **388** (1989), pp. 193-201.

[42] R. Hammons and P.V. Kumar, " On a Recent 4-Phase Sequence Design for CDMA," IEICE Trans. Commun., vol. E76-B, no. 8, Aug. 1993, pp. 804-813.

[43] P.V. Kumar and O. Moreno, "Polyphase sequences with correlation properties better than binary sequences," *IEEE Trans. Inform. Theory*, vol. 37, no. 3, pp. 603-616, May 1991.

[44] A.R. Hammons Jr., P.V. Kumar, A.R. Calderbank, N.J.A. Sloane and P. Solé, "The Z_4-Linearity of Kerdock, Preparata, Goethals and Related Codes," *IEEE Trans. Inform. Theory*, vol. 40, no. 2, pp. 301-319, Mar. 1994.

[45] A.R. Hammons Jr. and P.V. Kumar, "On the Apparent Duality of the Kerdock and Preparata codes," *Proc. of ISIT '93*, San Antonio, Texas, Jan. 1993, pp. 196.

[46] P. Shankar, "On BCH codes over arbitrary integer rings," *IEEE Trans. Inform. Theory*, pp. 480-484, Jul. 1979.

[47] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding: Turbo codes," *Proc. 1993 IEEE International Conference on Communications*, Geneva, Switzerland, May 1993, pp. 1064-1070.

[48] D. Divsalar and F. Pollara, "On the Design of Turbo Codes," *The Telecommunications and Data Acquisition Progress Report 42-123, July-September 1995*, Jet Propulsion Laboratory, Pasadena, California, pp. 99-121, Nov. 15,1995. http://edms-www.jpl.nasa.gov/tda/progress_report/42-123/123D.pdf.

[49] S. Benedetto, and G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 409-428, Mar. 1996.

[50] R. Lidl and H. Niederreiter, *Finite Fields, Encyclopedia of Mathematics and Its Applications*, vol. 20, Reading, MA: Addison Wesley Publishing Company, 1983.

[51] I.F. Blake and J.W. Mark, "A Note on Complex Sequences with Low Correlation," *IEEE Trans. Inform. Theory*, vol. IT-28, no. 5, Sept. 1982.

[52] G.E. Corazza, C. Ferrarelli and F. Vatalaro, "A Rice-Lognormal Terrestrial and Satellite Channel Model," *Proc. of ICUPC'94*, Sept. 94, San Diego, CA, pp. 155-159.

[53] G.E. Corazza, C. Ferrarelli and F. Vatalaro, "A Statistical Model for Land Mobile Satellite Channels," *IEEE Trans. Vehic. Technol.*, vol VT-43, Aug. 1994.

[54] E. Lutz, D. Cygan, M. Dippold, F. Dolainsky and W. Papke, "The Land Mobile Satellite Communications Channel - Recording, Statistics and Channel Model," *IEEE Trans. Vehic. Technol.*, vol. VT-40, pp. 375-385, May 1991.

[55] C. Loo. "A Statistical Model for Land Mobile Satellite Link." *IEEE Trans. Vehic. Technol.*, vol. 34, pp. 122-127, Aug. 1985.

[56] C. Loo and N. Secord. "Computer models for fading channels with applications to digital transmission." *IEEE Trans. Vehic. Technol.*, VT-40, pp. 700-707, Nov. 1991.

[57] R. Pichna, R. Kerr, Q. Wang, V.K. Bhargava and I.F. Blake "CDMA cellular network analysis software.", *Final Report* prepared for the Department of Communications under contract #36-001-2-3560/01-ST, Ottawa ON, 1993.

[58] R.W. Kerr and P.J. McLane, "Coherent Detection of Interleaved Trellis Encoded CPFSK on Shadowed Mobile Satellite Channels. *IEEE Trans. Vehic. Technol.*, vol. 41, no.2, pp. 159-169, May 1992.

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

[60] A.J. Mueller, "Issues in Diversity and Adaptive Error Control Coding for Wireless Communications." M.A.Sc. Thesis, University of Victoria, Victoria, B.C., Canada, 1995.

Appendix A

Tables of Polynomials

In order to reduce the search for good codes over $\mathbb{Z}_q$, we restricted the polynomials $g(x)$ and $f(x)$, which define the code, to have no factors of lesser degree in $\mathbb{Z}_q[x]$. When q is prime, this is equivalent to requiring $g(x)$ and $f(x)$ to be irreducible. As irreducible polynomials over fields are well known and are tabulated in the literature (cf. [50]), we will not list them here. However, we want irreducible polynomials over $\mathbb{Z}_q$ when q is not prime as well. A polynomial $p \in R[x]$ is irreducible over R if p has a positive degree and $p = bc$ with $b, c \in R[x]$ implies that b or c is a constant polynomial [50]. In other words, it is irreducible if it allows only trivial factorization. However, there is the case when working in $\mathbb{Z}_q[x]$, that there may exist a factor of equal or greater degree. For example, let $f(x) = 2x + 3$ and $g(x) = 3x^2 + 2x + 1$ in $\mathbb{Z}_6$, then $f(x)g(x) = x^2 + 2x + 3$. Another example is $f(x) = 3x + 3$ and $g(x) = 2x + 2$ where the product is equal 0.

The polynomials given in the Tables A.1-A.10 are over $\mathbb{Z}_q$ and have no factors of lesser degree. By formal definition they are not necessarily irreducible as there may exist factors of higher degrees. However, the purpose of using these polynomials was to ensure that two polynomials did not have factors of lesser degree. In the tables, the polynomial $a_n x^n + a_{n-1} x^{x-1} + \ldots + a_0$ is abbreviated as $a_n\ a_{n_1}\ \ldots\ a_0$.

The polynomials in the tables were computed using the Erastothenes sieve method. In this method, to find all the polynomials of degree n that have no factors less than degree n, all of the reducible polynomials of degree n are calculated. The calculation is done by multiplying all possible factors (less than degree n) which produce a degree n polynomial and removing it from the set of all polynomials of degree n. After all of the computation is completed the polynomials remaining in the set have no factors less than degree n. When q is prime the polynomials are called irreducible. This method

Table A.1. *Polynomials over $\mathbb{Z}_4[x]$ with no factors of lesser degree for degrees 2 to 4*

$\mathbb{Z}_q = 4$ Degree 2									
101	102	111	113	122	123	131	133	201	221
$\mathbb{Z}_q = 4$ Degree 3									
1002	1011	1013	1022	1031	1033	1101	1103	1113	1121
1123	1131	1202	1211	1213	1222	1231	1233	1301	1303
1311	1321	1323	1333	2001	2021	2201	2221		
$\mathbb{Z}_q = 4$ Degree 4									
10001	10002	10011	10013	10022	10023	10031	10033	10101	10103
10121	10122	10123	10202	10203	10211	10213	10221	10222	10231
10233	10301	10302	10303	10321	10323	11001	11003	11013	11021
11023	11031	11102	11111	11113	11122	11131	11133	11201	11203
11211	11221	11223	11233	11302	11311	11313	11322	11331	11333
12002	12003	12011	12013	12021	12022	12031	12033	12101	12102
12103	12121	12123	12201	12202	12211	12213	12222	12223	12231
12233	12301	12303	12321	12322	12323	13001	13003	13011	13021
13023	13033	13102	13111	13113	13122	13131	13133	13201	13203
13213	13221	13223	13231	13302	13311	13313	13322	13331	13333
20001	20021	20201	20221	22001	22021	22201	22221		

found the irreducible polynomials that are presented [50]. These tables are omitted here as they are presented elsewhere. The sieve method is impractical for large degree polynomials, however only polynomials with small degrees were of interest.

The tables present the results of the sieve method. They are not the product of any polynomial of lesser degree. Constant multiples of the polynomials presented in the following tables were omitted.

Table A.2. *Polynomials over $\mathbb{Z}_4[x]$ with no factors of lesser degree 5*

$\mathbb{Z}_q = 4$ Degree 5							
100002	100022	100101	100103	100121	100123	100202	100222
100301	100303	100321	100323	101001	101003	101021	101023
101111	101113	101131	101133	101201	101203	101221	101223
101311	101313	101331	101333	102002	102022	102101	102103
102121	102123	102202	102222	102301	102303	102321	102323
103001	103003	103021	103023	103111	103113	103131	103133
103201	103203	103221	103223	103311	103313	103331	103333
110013	110031	110111	110113	110131	110133	110211	110233
110311	110313	110331	110333	111011	111013	111031	111033
111101	111103	111121	111123	111211	111213	111231	111233
111301	111303	111321	111323	112011	112033	112111	112113
112131	112133	112213	112231	112311	112313	112331	112333
113011	113013	113031	113033	113101	113103	113121	113123
113211	113213	113231	113233	113301	113303	113321	113323
120002	120022	120101	120103	120121	120123	120202	120222
120301	120303	120321	120323	121001	121003	121021	121023
121111	121113	121131	121133	121201	121203	121221	121223
121311	121313	121331	121333	122002	122022	122101	122103
122121	122123	122202	122222	122301	122303	122321	122323
123001	123003	123021	123023	123111	123113	123131	123133
123201	123203	123221	123223	123311	123313	123331	123333
130011	130033	130111	130113	130131	130133	130213	130231
130311	130313	130331	130333	131011	131013	131031	131033
131101	131103	131121	131123	131211	131213	131231	131233
131301	131303	131321	131323	132013	132031	132111	132113
132131	132133	132211	132233	132311	132313	132331	132333
133011	133013	133031	133033	133101	133103	133121	133123
133211	133213	133231	133233	133301	133303	133321	133323
200001	200021	200201	200221	202001	202021	202201	202221
220001	220021	220201	220221	222001	222021	222201	222221

Table A.3. *Polynomials over $\mathbb{Z}_6[x]$ with no factors of lesser degree for degrees 2 and 3*

$\mathbb{Z}_q = 6$ Degree 2											
101	104	111	112	113	115	122	125	131	133	134	135
142	145	151	152	153	155	205	211	214	221	232	235
241	251	254	311	313	315	331					

$\mathbb{Z}_q = 6$ Degree 3											
1011	1013	1015	1021	1022	1024	1025	1031	1033	1035	1051	1052
1053	1054	1055	1101	1102	1103	1105	1112	1115	1121	1123	1124
1125	1132	1135	1141	1142	1143	1145	1151	1154	1201	1204	1211
1213	1214	1215	1222	1225	1231	1233	1234	1235	1241	1244	1251
1252	1253	1255	1301	1303	1305	1321	1322	1323	1324	1325	1341
1343	1345	1351	1352	1354	1355	1402	1405	1411	1412	1413	1415
1421	1424	1431	1432	1433	1435	1442	1445	1451	1453	1454	1455
1501	1503	1504	1505	1511	1514	1521	1522	1523	1525	1531	1534
1541	1543	1544	1545	1552	1555	2011	2012	2014	2015	2041	2045
2102	2105	2111	2114	2122	2125	2132	2135	2141	2144	2152	2155
2201	2212	2215	2221	2231	2234	2245	2251	2254	2311	2312	2314
2315	2341	2342	2344	2345	2405	2411	2414	2425	2432	2435	2441
2452	2455	2501	2504	2512	2515	2521	2524	2531	2534	2542	2545
2551	2554	3011	3013	3015	3031	3101	3103	3105	3121	3123	3125
3141	3143	3145	3211	3213	3215	3231	3233	3235	3251	3253	3255
3301	3321	3323	3325								

Table A.4. *Polynomials of degree 4 over $\mathbb{Z}_6[x]$ with no factors of lesser degree*

$\mathbb{Z}_q = 6$ Degree 4									
10011	10012	10013	10015	10022	10025	10031	10033	10035	10042
10045	10051	10052	10053	10055	10102	10105	10111	10114	10121
10124	10125	10132	10135	10141	10144	10145	10151	10154	10202
10205	10211	10213	10215	10231	10232	10233	10235	10251	10253
10255	10312	10315	10321	10322	10325	10331	10334	10341	10342
10345	10352	10355	10402	10405	10411	10413	10414	10415	10421
10424	10431	10432	10433	10435	10441	10444	10451	10453	10454
10455	10502	10505	10521	10523	10531	10532	10534	10535	10541
10543	11001	11002	11003	11005	11012	11021	11023	11024	11025
11032	11035	11041	11043	11045	11051	11054	11101	11104	11111
11113	11114	11115	11122	11125	11131	11133	11134	11135	11141
11144	11151	11152	11153	11155	11201	11203	11205	11221	11222
11223	11225	11241	11243	11245	11252	11254	11255	11302	11305
11311	11313	11315	11321	11324	11331	11332	11333	11335	11345
11351	11353	11354	11355	11401	11403	11404	11405	11411	11414
11421	11422	11423	11425	11431	11434	11441	11443	11444	11445
11452	11455	11511	11513	11515	11521	11522	11525	11531	11533
11535	11551	11552	11553	11555	12002	12005	12011	12013	12014
12015	12031	12032	12033	12035	12041	12044	12051	12053	12055
12101	12104	12112	12115	12121	12124	12125	12131	12134	12142
12143	12145	12151	12154	12211	12212	12213	12215	12231	12233
12235	12242	12245	12251	12253	12255	12301	12302	12305	12311
12314	12332	12335	12341	12343	12344	12352	12355	12401	12404
12411	12412	12413	12415	12421	12424	12431	12433	12434	12435
12442	12445	12451	12453	12454	12455	12501	12505	12511	12512
12514	12515	12523	12525	12542	12545	13001	13003	13005	13012
13015	13021	13022	13023	13025	13034	13041	13042	13043	13045
13052	13055	13102	13105	13111	13113	13114	13115	13121	13124
13131	13132	13133	13135	13141	13144	13151	13153	13154	13155
13201	13202	13203	13205	13221	13223	13225	13232	13234	13235

continued on next page

$\mathbb{Z}_q = 6$ Degree 4 *Table A.4 (continued)*									
13241	13243	13245	13301	13311	13312	13313	13315	13322	13325
13331	13333	13335	13342	13345	13351	13352	13353	13355	13401
13402	13403	13405	13411	13414	13421	13423	13424	13425	13432
13435	13441	13443	13444	13445	13451	13454	13501	13502	13505
13511	13513	13515	13531	13532	13533	13535	13551	13553	13555
14002	14005	14011	14013	14015	14021	14024	14031	14032	14033
14035	14051	14053	14054	14055	14101	14104	14111	14114	14122
14123	14125	14131	14134	14141	14144	14145	14152	14155	14211
14213	14215	14222	14225	14231	14233	14235	14251	14252	14253
14255	14301	14302	14305	14312	14315	14321	14323	14324	14332
14335	14351	14354	14401	14404	14411	14413	14414	14415	14422
14425	14431	14433	14434	14435	14441	14444	14451	14452	14453
14455	14501	14505	14522	14525	14543	14545	14551	14552	14554
14555	15001	15002	15003	15005	15011	15014	15021	15023	15025
15032	15035	15041	15043	15044	15045	15052	15101	15104	15111
15112	15113	15115	15121	15124	15131	15133	15134	15135	15142
15145	15151	15153	15154	15155	15201	15203	15205	15212	15214
15215	15221	15223	15225	15241	15242	15243	15245	15302	15305
15311	15313	15314	15315	15325	15331	15332	15333	15335	15341
15344	15351	15353	15355	15401	15403	15404	15405	15412	15415
15421	15423	15424	15425	15431	15434	15441	15442	15443	15445
15451	15454	15511	15512	15513	15515	15531	15533	15535	15541
15542	15545	15551	15553	15555	20011	20014	20021	20041	20051
20054	20101	20104	20131	20134	20201	20212	20215	20225	20231
20234	20245	20252	20255	20311	20314	20321	20324	20341	20344
20351	20354	20401	20431	20434	20501	20504	20512	20515	20522
20525	20531	20534	20542	20545	20552	20555	21001	21004	21011
21022	21025	21031	21034	21052	21055	21121	21124	21125	21151
21154	21202	21205	21212	21215	21221	21224	21232	21235	21242
21245	21251	21254	21301	21304	21322	21325	21331	21334	21341
21352	21355	21421	21424	21451	21454	21455	21502	21505	21512

continued on next page

$\mathbb{Z}_q = 6$ Degree 4 *Table A.4 (continued)*									
21515	21521	21524	21532	21535	21542	21545	21551	21554	22001
22012	22015	22031	22034	22045	22111	22114	22141	22144	22205
22211	22214	22225	22232	22235	22241	22252	22255	22301	22304
22312	22315	22331	22334	22342	22345	22411	22414	22441	22502
22505	22511	22514	22522	22525	22532	22535	22541	22544	22552
22555	23011	23014	23021	23024	23035	23041	23044	23051	23054
23101	23104	23105	23131	23134	23201	23204	23212	23215	23222
23225	23231	23234	23242	23245	23252	23255	23305	23311	23314
23321	23324	23341	23344	23351	23354	23401	23404	23431	23434
23435	23501	23504	23512	23515	23522	23525	23531	23534	23542
23545	23552	23555	24001	24025	24031	24034	24052	24055	24121
24124	24151	24154	24205	24212	24215	24221	24232	24235	24245
24251	24254	24301	24304	24322	24325	24331	24334	24352	24355
24421	24451	24454	24502	24505	24512	24515	24521	24524	24532
24535	24542	24545	24551	24554	25001	25004	25012	25015	25031
25034	25042	25045	25051	25111	25114	25141	25144	25145	25202
25205	25211	25214	25222	25225	25232	25235	25241	25244	25252
25255	25301	25304	25312	25315	25321	25331	25334	25342	25345
25411	25414	25415	25441	25444	25502	25505	25511	25514	25522
25525	25532	25535	25541	25544	25552	25555	30011	30013	30015
30031	30211	30213	30215	30231	30233	30235	30251	30253	30255
31001	31003	31005	31021	31023	31025	31041	31043	31045	31111
31113	31115	31131	31133	31135	31151	31153	31155	31201	31203
31205	31221	31223	31225	31241	31243	31245	31311	31313	31315
31331	31333	31335	31351	31353	31355	31401	31403	31405	31421
31423	31425	31441	31443	31445	31511	31513	31515	31531	31533
31535	31551	31553	31555	32011	32013	32015	32031	32033	32035
32051	32053	32055	32105	32125	32141	32211	32213	32215	32231
32233	32235	32251	32253	32255	32341	32345	32411	32413	32415
32431	32433	32435	32451	32453	32455	32501	32521	32545	33001
33021	33023	33025	33111	33113	33115	33131	33133	33135	33151
33153	33155	33201	33203	33205	33221	33223	33225	33241	33243
33245	33311	33313	33315	33331					

Table A.5. *Polynomials over $\mathbb{Z}_8[x]$ with no factors of lesser degree for degrees 2 and 3*

$\mathbb{Z}_q = 8$ Degree 2											
101	102	103	105	106	111	113	115	117	122	123	124
126	127	131	133	135	137	141	142	145	146	147	151
153	155	157	162	163	164	166	167	171	173	175	177
201	203	221	223	241	243	261	263	421	423		

$\mathbb{Z}_q = 8$ Degree 3											
1002	1004	1006	1011	1013	1015	1017	1022	1026	1031	1033	1035
1037	1042	1044	1046	1051	1053	1055	1057	1062	1066	1071	1073
1075	1077	1101	1103	1105	1107	1113	1117	1121	1123	1125	1127
1131	1135	1137	1141	1143	1145	1147	1153	1157	1161	1163	1165
1167	1171	1173	1175	1202	1204	1206	1211	1213	1215	1217	1222
1226	1231	1233	1235	1237	1242	1244	1246	1251	1253	1255	1257
1262	1266	1271	1273	1275	1277	1301	1303	1305	1307	1311	1315
1321	1323	1325	1327	1333	1335	1337	1341	1343	1345	1347	1351
1355	1361	1363	1365	1367	1371	1373	1377	1402	1404	1406	1411
1413	1415	1417	1422	1426	1431	1433	1435	1437	1442	1444	1446
1451	1453	1455	1457	1462	1466	1471	1473	1475	1477	1501	1503
1505	1507	1513	1517	1521	1523	1525	1527	1531	1533	1535	1541
1543	1545	1547	1553	1557	1561	1563	1565	1567	1571	1575	1577
1602	1604	1606	1611	1613	1615	1617	1622	1626	1631	1633	1635
1637	1642	1644	1646	1651	1653	1655	1657	1662	1666	1671	1673
1675	1677	1701	1703	1705	1707	1711	1715	1721	1723	1725	1727
1731	1733	1737	1741	1743	1745	1747	1751	1755	1761	1763	1765
1767	1773	1775	1777	2001	2003	2021	2023	2041	2043	2061	2063
2201	2203	2221	2223	2241	2243	2261	2263	2401	2403	2421	2423
2441	2443	2461	2463	2601	2603	2621	2623	2641	2643	2661	2663
4001	4021	4023	4041	4101	4102	4105	4106	4107	4111	4113	4115
4117	4122	4123	4124	4126	4127	4131	4133	4135	4137	4141	4142
4143	4145	4146	4151	4153	4155	4157	4162	4163	4164	4166	4167
4171	4173	4175	4177	4401	4421	4423	4441				

Table A.6. *Polynomials over* $\mathbb{Z}_9[x]$ *and* $\mathbb{Z}_{10}[x]$ *with no factors of lesser degree for degree 2*

$\mathbb{Z}_q = 9$ Degree 2											
101	103	104	106	107	111	112	114	115	118	122	124
125	127	128	131	133	134	136	137	141	142	145	147
148	151	152	155	157	158	161	163	164	166	167	172
174	175	177	178	181	182	184	185	188	301	302	331
332	361	362									

$\mathbb{Z}_q = 10$ Degree 2											
102	103	107	108	111	112	113	115	116	117	119	123
124	128	129	131	133	134	135	137	138	139	141	142
146	147	151	152	153	155	157	158	159	161	162	166
167	171	173	174	175	177	178	179	183	184	188	189
191	192	193	195	196	197	199	201	209	211	213	216
218	227	229	232	234	237	239	241	243	251	254	256
259	261	263	272	274	277	279	287	289	291	293	296
298	511	513	515	517	519	551					

Table A.7. *Polynomials over $\mathbb{Z}_{12}[x]$ with no factors of lesser degree for degree 2*

$\mathbb{Z}_q = 12$ Degree 2								
1 0 1	1 0 2	1 0 4	1 0 5	1 0 6	1 0 7	1 0 9	1 0 10	1 1 1
1 1 2	1 1 3	1 1 5	1 1 7	1 1 8	1 1 9	1 1 11	1 2 2	1 2 3
1 2 5	1 2 6	1 2 7	1 2 8	1 2 10	1 2 11	1 3 1	1 3 3	1 3 4
1 3 5	1 3 7	1 3 9	1 3 10	1 3 11	1 4 1	1 4 2	1 4 5	1 4 6
1 4 8	1 4 9	1 4 10	1 4 11	1 5 1	1 5 2	1 5 3	1 5 5	1 5 7
1 5 8	1 5 9	1 5 11	1 6 1	1 6 2	1 6 3	1 6 4	1 6 6	1 6 7
1 6 10	1 6 11	1 7 1	1 7 2	1 7 3	1 7 5	1 7 7	1 7 8	1 7 9
1 7 11	1 8 1	1 8 2	1 8 5	1 8 6	1 8 8	1 8 9	1 8 10	1 8 11
1 9 1	1 9 3	1 9 4	1 9 5	1 9 7	1 9 9	1 9 10	1 9 11	1 10 2
1 10 3	1 10 5	1 10 6	1 10 7	1 10 8	1 10 10	1 10 11	1 11 1	1 11 2
1 11 3	1 11 5	1 11 7	1 11 8	1 11 9	1 11 11	2 0 1	2 0 3	2 0 5
2 1 1	2 1 4	2 1 7	2 1 10	2 2 1	2 2 3	2 2 5	2 3 2	2 3 5
2 3 8	2 3 11	2 4 1	2 4 3	2 4 5	2 5 1	2 5 4	2 5 7	2 5 10
2 6 1	2 6 3	2 6 5	2 8 1	2 8 3	2 8 5	2 10 1	2 10 3	2 10 5
3 0 2	3 0 7	3 1 1	3 1 3	3 1 5	3 1 7	3 1 9	3 1 11	3 2 1
3 2 2	3 2 5	3 2 6	3 2 9	3 2 10	3 3 1	3 3 7	3 4 2	3 4 3
3 4 6	3 4 7	3 4 10	3 4 11	3 6 1	3 6 2	3 7 1	3 7 3	3 7 5
3 7 7	3 7 9	3 7 11	3 9 1	3 9 7	4 0 1	4 1 2	4 1 5	4 1 8
4 1 11	4 2 5	4 3 1	4 3 4	4 3 7	4 3 10	4 4 5	4 5 2	4 5 5
4 5 8	4 5 11	4 6 1	4 8 5	4 10 5	6 0 1	6 2 1	6 2 3	6 2 5
6 4 1	6 4 3	6 4 5	6 6 1					

Table A.8. *Polynomials over $\mathbb{Z}_{14}[x]$ with no factors of lesser degree for degree 2*

$\mathbb{Z}_q = 14$ Degree 2								
1 0 1	1 0 2	1 0 4	1 0 8	1 0 9	1 0 11	1 1 1	1 1 3	1 1 4
1 1 5	1 1 6	1 1 7	1 1 9	1 1 10	1 1 11	1 1 13	1 2 2	1 2 3
1 2 5	1 2 9	1 2 10	1 2 12	1 3 1	1 3 3	1 3 5	1 3 6	1 3 7
1 3 8	1 3 9	1 3 11	1 3 12	1 3 13	1 4 1	1 4 5	1 4 6	1 4 8
1 4 12	1 4 13	1 5 1	1 5 2	1 5 3	1 5 5	1 5 7	1 5 9	1 5 10
1 5 11	1 5 12	1 5 13	1 6 3	1 6 4	1 6 6	1 6 10	1 6 11	1 6 13
1 7 1	1 7 2	1 7 3	1 7 4	1 7 5	1 7 7	1 7 8	1 7 9	1 7 11
1 7 13	1 8 3	1 8 4	1 8 6	1 8 10	1 8 11	1 8 13	1 9 1	1 9 2
1 9 3	1 9 5	1 9 7	1 9 9	1 9 10	1 9 11	1 9 12	1 9 13	1 10 1
1 10 5	1 10 6	1 10 8	1 10 12	1 10 13	1 11 1	1 11 3	1 11 5	1 11 6
1 11 7	1 11 8	1 11 9	1 11 11	1 11 12	1 11 13	1 12 2	1 12 3	1 12 5
1 12 9	1 12 10	1 12 12	1 13 1	1 13 3	1 13 4	1 13 5	1 13 6	1 13 7
1 13 9	1 13 10	1 13 11	1 13 13	2 0 1	2 0 9	2 0 11	2 1 2	2 1 3
2 1 5	2 1 9	2 1 10	2 1 12	2 2 1	2 2 5	2 2 13	2 3 3	2 3 4
2 3 6	2 3 10	2 3 11	2 3 13	2 4 3	2 4 11	2 4 13	2 5 1	2 5 5
2 5 6	2 5 8	2 5 12	2 5 13	2 6 3	2 6 5	2 6 9	2 7 1	2 7 2
2 7 4	2 7 8	2 7 9	2 7 11	2 8 3	2 8 5	2 8 9	2 9 1	2 9 5
2 9 6	2 9 8	2 9 12	2 9 13	2 10 3	2 10 11	2 10 13	2 11 3	2 11 4
2 11 6	2 11 10	2 11 11	2 11 13	2 12 1	2 12 5	2 12 13	2 13 2	2 13 3
2 13 5	2 13 9	2 13 10	2 13 12	7 1 1	7 1 3	7 1 5	7 1 7	7 1 9
7 1 11	7 1 13	7 7 1						

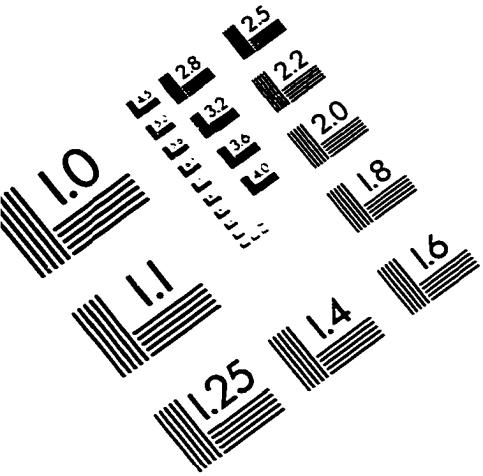
Table A.9. *Polynomials over $\mathbb{Z}_{15}[x]$ with no factors of lesser degree for degree 2*

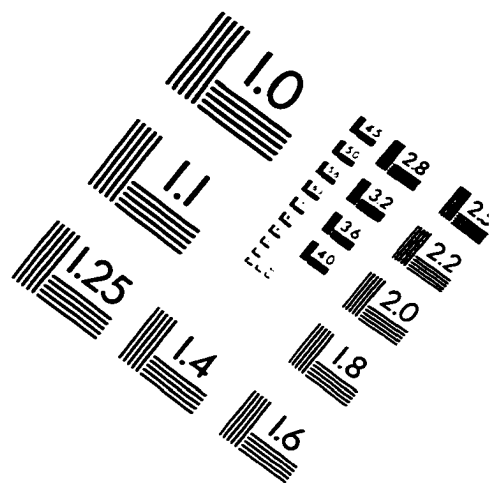
$\mathbb{Z}_7 = 15$ Degree 2								
1 0 1	1 0 2	1 0 3	1 0 4	1 0 7	1 0 8	1 0 10	1 0 12	1 0 13
1 1 1	1 1 2	1 1 5	1 1 6	1 1 7	1 1 8	1 1 11	1 1 12	1 1 14
1 2 2	1 2 3	1 2 4	1 2 5	1 2 8	1 2 9	1 2 11	1 2 13	1 2 14
1 3 1	1 3 3	1 3 4	1 3 7	1 3 8	1 3 9	1 3 10	1 3 13	1 3 14
1 4 1	1 4 2	1 4 5	1 4 6	1 4 7	1 4 8	1 4 11	1 4 12	1 4 14
1 5 2	1 5 3	1 5 5	1 5 7	1 5 8	1 5 11	1 5 12	1 5 13	1 5 14
1 6 1	1 6 2	1 6 4	1 6 6	1 6 7	1 6 10	1 6 11	1 6 12	1 6 13
1 7 2	1 7 3	1 7 4	1 7 5	1 7 8	1 7 9	1 7 11	1 7 13	1 7 14
1 8 2	1 8 3	1 8 4	1 8 5	1 8 8	1 8 9	1 8 11	1 8 13	1 8 14
1 9 1	1 9 2	1 9 4	1 9 6	1 9 7	1 9 10	1 9 11	1 9 12	1 9 13
1 10 2	1 10 3	1 10 5	1 10 7	1 10 8	1 10 11	1 10 12	1 10 13	1 10 14
1 11 1	1 11 2	1 11 5	1 11 6	1 11 7	1 11 8	1 11 11	1 11 12	1 11 14
1 12 1	1 12 3	1 12 4	1 12 7	1 12 8	1 12 9	1 12 10	1 12 13	1 12 14
1 13 2	1 13 3	1 13 4	1 13 5	1 13 8	1 13 9	1 13 11	1 13 13	1 13 14
1 14 1	1 14 2	1 14 5	1 14 6	1 14 7	1 14 8	1 14 11	1 14 12	1 14 14
3 0 1	3 0 4	3 1 2	3 1 4	3 1 7	3 1 9	3 1 12	3 1 14	3 2 1
3 2 3	3 2 6	3 2 8	3 2 11	3 2 13	3 3 1	3 3 8	3 4 2	3 4 4
3 4 7	3 4 9	3 4 12	3 4 14	3 5 1	3 5 4	3 5 6	3 5 9	3 5 11
3 5 14	3 6 2	3 6 4	3 8 1	3 8 3	3 8 6	3 8 8	3 8 11	3 8 13
3 9 2	3 9 4	3 12 1	3 12 8	5 0 2	5 1 1	5 1 4	5 1 7	5 1 10
5 1 13	5 2 1	5 2 4	5 2 7	5 2 10	5 2 13	5 3 2	5 3 5	5 3 8
5 3 11	5 3 14	5 5 1	5 10 1					

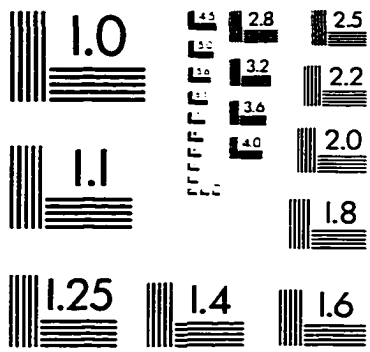
Table A.10. *Polynomials over $\mathbb{Z}_{16}[x]$ with no factors of lesser degree for degree 2*

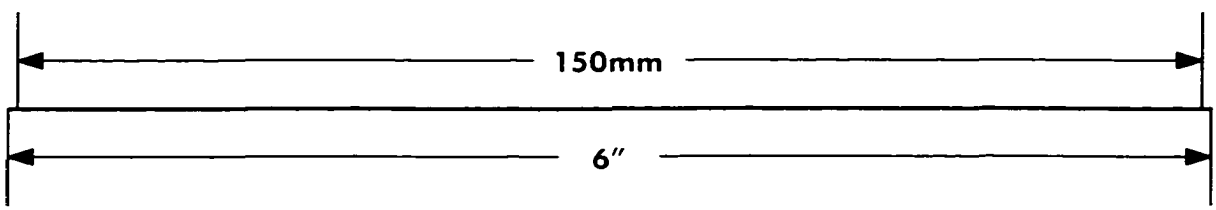
$\mathbb{Z}_q = 16$ Degree 2								
1 0 1	1 0 2	1 0 3	1 0 4	1 0 5	1 0 6	1 0 8	1 0 9	1 0 10
1 0 11	1 0 13	1 0 14	1 1 1	1 1 3	1 1 5	1 1 7	1 1 9	1 1 11
1 1 13	1 1 15	1 2 2	1 2 3	1 2 4	1 2 5	1 2 6	1 2 7	1 2 9
1 2 10	1 2 11	1 2 12	1 2 14	1 2 15	1 3 1	1 3 3	1 3 5	1 3 7
1 3 9	1 3 11	1 3 13	1 3 15	1 4 1	1 4 2	1 4 5	1 4 6	1 4 7
1 4 8	1 4 9	1 4 10	1 4 12	1 4 13	1 4 14	1 4 15	1 5 1	1 5 3
1 5 5	1 5 7	1 5 9	1 5 11	1 5 13	1 5 15	1 6 1	1 6 2	1 6 3
1 6 4	1 6 6	1 6 7	1 6 10	1 6 11	1 6 12	1 6 13	1 6 14	1 6 15
1 7 1	1 7 3	1 7 5	1 7 7	1 7 9	1 7 11	1 7 13	1 7 15	1 8 1
1 8 2	1 8 3	1 8 4	1 8 5	1 8 6	1 8 8	1 8 9	1 8 10	1 8 11
1 8 13	1 8 14	1 9 1	1 9 3	1 9 5	1 9 7	1 9 9	1 9 11	1 9 13
1 9 15	1 10 1	1 10 2	1 10 3	1 10 4	1 10 6	1 10 7	1 10 10	1 10 11
1 10 12	1 10 13	1 10 14	1 10 15	1 11 1	1 11 3	1 11 5	1 11 7	1 11 9
1 11 11	1 11 13	1 11 15	1 12 1	1 12 2	1 12 5	1 12 6	1 12 7	1 12 8
1 12 9	1 12 10	1 12 12	1 12 13	1 12 14	1 12 15	1 13 1	1 13 3	1 13 5
1 13 7	1 13 9	1 13 11	1 13 13	1 13 15	1 14 2	1 14 3	1 14 4	1 14 5
1 14 6	1 14 7	1 14 9	1 14 10	1 14 11	1 14 12	1 14 14	1 14 15	1 15 1
1 15 3	1 15 5	1 15 7	1 15 9	1 15 11	1 15 13	1 15 15	2 0 1	2 0 3
2 0 5	2 0 7	2 2 1	2 2 3	2 2 5	2 2 7	2 4 1	2 4 3	2 4 5
2 4 7	2 6 1	2 6 3	2 6 5	2 6 7	2 8 1	2 8 3	2 8 5	2 8 7
2 10 1	2 10 3	2 10 5	2 10 7	2 12 1	2 12 3	2 12 5	2 12 7	2 14 1
2 14 3	2 14 5	2 14 7	4 0 1	4 2 1	4 2 3	4 2 5	4 2 7	4 4 3
4 6 1	4 6 3	4 6 5	4 6 7	4 8 1	4 12 3	8 0 1	8 4 1	8 4 3
8 8 1								

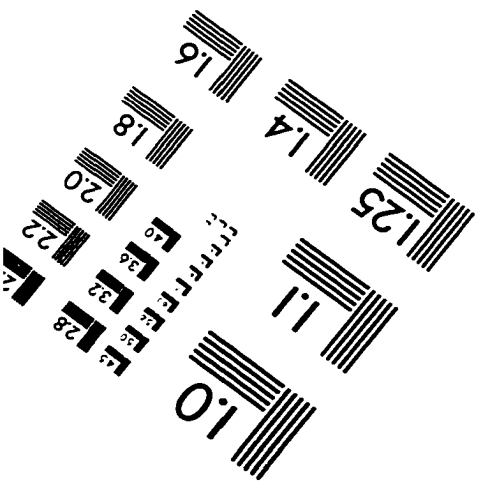
IMAGE EVALUATION
TEST TARGET (QA-3)

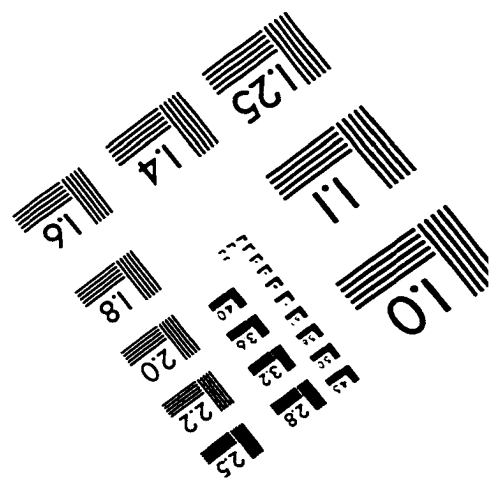












APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989