Air Force Institute of Technology

AFIT Scholar

3-11-2004

Multiple Access Interference Characterization for Direct-Sequence Spread-Spectrum Communications Using Chip Waveform Shaping

Matthew G. Glen

Follow this and additional works at: https://scholar.afit.edu/etd

Part of the Electrical and Electronics Commons

Recommended Citation

MULTIPLE ACCESS INTERFERENCE
CHARACTERIZATION FOR DIRECT-SEQUENCE
SPREAD-SPECTRUM COMMUNICATIONS USING
CHIP WAVEFORM SHAPING

THESIS

Matthew G. Glen, Captain, USAF

AFIT/GE/ENG/04-10

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

MULTIPLE ACCESS INTERFERENCE CHARACTERIZATION FOR
DIRECT-SEQUENCE SPREAD-SPECTRUM COMMUNICATIONS
USING CHIP WAVEFORM SHAPING

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

Matthew G. Glen, BS

Captain, USAF

March 2004

AFIT/GE/ENG/04-10

MULTIPLE ACCESS INTERFERENCE CHARACTERIZATION FOR
DIRECT-SEQUENCE SPREAD-SPECTRUM COMMUNICATIONS
USING CHIP WAVEFORM SHAPING

Matthew G. Glen, BS
Captain, USAF

Approved:

//signed//	11 March 2004
Michael A. Temple, Ph.D.	Date
Thesis Advisor	
//signed//	11 March 2004
Major Matthew E. Goda, Ph.D.	Date
Committee Member	
//signed//	11 March 2004
Major Todd B. Hale, Ph.D.	Date
Committee Member	

Acknowledgments

       I would first like to thank my wonderful wife for all of her patience in dealing with me and loving care in raising our son. Second, I must thank my advisor Dr. Michael Temple. Without his academic guidance and attention to detail this thesis would not be what it is. Finally, I wish to thank my classmates in GE-04M for their camaraderie and all of the good times mixed in with the AFIT mess.

Matthew G. Glen

iv

Table of Contents

List of Figures

List of Tables

AFIT/GE/ENG/04-10

Abstract

The modern world has an increasing demand for wireless multiple access communications; direct-sequence spread-spectrum multiple access (DS/SSMA) systems comprise many of these communication systems. A better understanding of multiple access interference (MAI) effects on DS/SSMA system performance, specifically their impact on overall system bit error rate (BER), enables system designers to minimize MAI degradation and produce greater DS/SSMA system capacity.

This research characterizes MAI effects on DS/SSMA system performance through simulation in Matlab®, and explores the impact of multiple access code selection, chip waveform shaping, and multiple access code length on BER for both *synchronous* and *asynchronous* multiple access networks. In addition, the simulated DS/SSMA model permits rapid research into the effects of additional factors on BER.

Prior to experimental testing, model validation is conducted through single user trials and by comparison with existing research for similar system designs. For *synchronous* and *asynchronous* networks, Gold coding improves BER by 7.5 and 4.0 dB, respectively, relative to aperiodic random spreading codes. *Synchronous* network results show that chip waveform shaping provides no significant BER improvement for the Blackman or Lanczos shapes. However, *asynchronous* network results show a potential BER improvement for Blackman and Lanczos shapes. Increasing code length from 31 to 511 resulted in a 7.5 dB BER improvement. Collectively, these results directly relate changes in BER to waveform cross-correlation statistics.

xi

MULTIPLE ACCESS INTERFERENCE CHARACTERIZATION FOR

DIRECT-SEQUENCE SPREAD-SPECTRUM COMMUNICATIONS

USING CHIP WAVEFORM SHAPING

I. Introduction

1.1   Motivation for Multiple Access Interference Characterization

The ever increasing demand for world-wide multiple access wireless communications drives the need to maximize the current system capability and transmission capacity. Many current systems employ direct-sequence spread-spectrum (DSSS) techniques to enable multiple access capability. A clearer understanding of how multiple users effect overall system performance, through characterization of the multiple access interference (MAI), enables more efficient use of current systems and better designs for future systems.

Although a large body of research exists on how MAI impacts direct-sequence spread-spectrum multiple access (DS/SSMA) performance, most of this work relies on analytical approximations. Additionally, there are numerous factors that can affect MAI contributions in a system between workload and environment (e.g., number of simultaneous transmitters, type of multiple access coding, and code length). Current approximations only account for a limited number of these factors and can require extensive recalculation when factors are changed or added. The state of existing research into MAI in DS/SSMA systems leaves open a need for a representative system model that is easily modified to account for different factors.

This need lends itself to characterization of MAI in DS/SSMA systems through simulation. Simulation of DS/SSMA system performance provides two main benefits. First, simulation allows for rapid testing of the effects that numerous factors can have on system performance and such factors can be changed or others added relatively easily. Second, simulation enables verification of future approximations through a vehicle that more closely represents the actual, physical communication system.

1.2   *Problem Statement and Scope*

To increase the capacity and capability of DS/SSMA systems, a greater understanding of MAI effects (as a function of multiple factors) on system performance is required. Simulation of a DS/SSMA system model provides the ability to characterize the effects of various design factors and permits verification of existing and future analytical approximations. This work provides modeling and simulation results for a representative DS/SSMA system and increases the understanding of particular factors' impact on MAI. Specifically, this work simulates the effect of multiple access code selection, chip waveform shaping, and multiple access code length on bit error rate (BER) in a DS/SSMA system. Furthermore, this research provides a simulation base capable of relatively easy modification to evaluate the impact of additional factors on BER.

1.3   *Methodology*

This research simulates the transmitter, channel, and receiver of a DS/SSMA system using Matlab®. The simulation contains all elements of random binary data

generation, data modulation, multiple access coding, transmission, reception, despreading, and communication symbol detection and estimation. All environmental and system level factors that potentially impact BER are contained within the simulation and easily modified to support additional research. Theoretical performance models and previous research provide the basis for system model verification. Simulation is used to estimate the effect of multiple access code selection, chip waveform shaping, and multiple access code length on BER.

1.4   *Thesis Organization*

This thesis contains five chapters. Chapter 1 provides an introduction on the need for MAI characterization for DS/SSMA systems. Chapter 2 outlines relevant background information on existing BER approximations and the factors to be explored in research. Research methodology is outlined in Chapter 3. Verification and experimental testing results are provided in Chapter 4. Chapter 5 contains a summary of contributions and findings and outlines possible future research. One appendix is provided containing the Matlab® code associated with system simulation.

II. Background

2.1 Direct Sequence Spread Spectrum Communications

Modern digital communication techniques have kindled widespread appeal and demand for improving existing and future communications systems. This increased demand has forced regulating organizations to establish limits on what portions of the spectrum specific applications can use. The result is limited bandwidth for all communications applications. The desire to get more use out of the available bandwidth has driven various techniques to increase the number of users or applications simultaneously occupying a specific spectral region. One such technique is direct-sequence, spread-spectrum multiple access (DS/SSMA).

2.1.1 Multiple Access Interference.

Direct-sequence spread-spectrum (DSSS) techniques are commonly used to implement multiple access communications. Existing systems using DSSS techniques include the Global Positioning System (GPS) and IS-95 digital cellular phone system [Peterson, Ziemer, and Borth, 1995]. DSSS coding modulates each communication symbol with a waveform consisting of multiple chip intervals. The chip modulating waveform has a particular pulse shape with either a positive or negative orientation as determined by the DSSS code. The undesired signals make up the multiple access interference (MAI) that increases with an increasing number of users. For perfectly orthogonal DSSS coding the MAI term is identically zero and the system functions as if there were only a single user present, i.e., the desired user. Although the orthogonal coding is optimal, the number of required users, mathematical code limitations, the computational complexity required to generate large numbers of orthogonal codes, and

the desire to analytically model systems typically necessitates the selection of non-orthogonal codes in either the design or analysis of DS/SSMA systems [Gerantiotis and Ghaffari, 1991]. DS/SSMA systems rely on code cross-correlation characteristics to reject undesired signals and minimize MAI. The ability to reject undesired signals is known as multiple access interference suppression. By reducing MAI in a given system, more users can simultaneously access the same link.

2.1.2   Signal and System Model.

The works of Yao and Pursley present a standard DS/SSMA system design. The system consists of K users with each using BPSK modulation for their respective data. The data modulated waveform $b_i(t)$ for each user can be represented by [Yao, 1977]

$$b_i(t) = \sum_{n=-\infty}^{\infty} b_{i,n} P_{T_s}(t - nT_s)$$

(2.1)

where $b_{i,n}$ is a sequence of elements, $b_{i,n} \ \varepsilon \ [-1,1]$, and $P_T(\bullet)$ is a unit height rectangular pulse of duration T_s, the symbol duration. Spreading code modulation $a_k(t)$ is defined as [Pursley, 1977]

$$a_k(t) = \sum_{j=-\infty}^{\infty} a_j^{(k)} p_{T_c}(t - jT_c)$$

(2.2)

where $a_j^{(k)}$ is a sequence of elements for the kth user, $a_j \ \varepsilon \ [-1,1]$, each chip interval has duration T_c, and the spreading code period is $N \bullet T_c$. The most common pulse shape used over each chip interval is rectangular, as represented by $p_{T_c}(t - jT_c)$ in (2.2). One of the most common spreading techniques involves multiplying each communication symbol by N coded replications of the chip waveform over one full code period per symbol interval. This particular technique is the basis for DSSS systems considered in

this research. Using the data and spreading code modulation of (2.1) and (2.2), respectively, the transmitted signal for the kth user is defined as

$$s_k(t) = \sqrt{2P}\, a_k(t) b_k(t) \cos(\omega_c t + \theta_k) \qquad (2.3)$$

For the equal power cases considered here, $\sqrt{2P}$ represents the transmitted power of all signals, $\cos(\omega_c t + \theta_k)$ is the phase modulated carrier of the kth user, ω_c is the modulation frequency, and θ_k is the phase delay of the kth user [Pursley, 1977].

The total received multiple access signal is the sum of K transmitted signals and the channel noise and may be expressed as

$$r(t) = \sum_{k=1}^{K} \sqrt{2P}\, a_k(t - \tau_k) b_k(t - \tau_k) \cos(\omega_c t + \phi_k) + n(t) \qquad (2.4)$$

where τ_k and ϕ_k are the time delay and received phase, respectively, of the kth user. The function $n(t)$ represents thermal channel noise and is assumed to be additive white Gaussian noise (AWGN) having two-sided spectral density of $N_o/2$. All subsequent derivations presented in this work assume the desired user's signal has zero time delay and zero received phase. All interfering users have time delays, relative to the desired user, in the interval [0, T_s] and relative phases in [0, 2π]. In the special case of a *synchronous* network all interfering users have zero time delay.

2.2   *Analytical Bit Error Rate (BER) Approximation*

The work of [Pursley, 1977] provides an expression for bit error rate (BER) in terms of the number of network users (K) and code length (N). Using a conventional matched filter design, the received signal is despread and demodulated via correlation which generates the test statistic Z_i given as [Pursley, 1977]

$$Z_i = \int_0^T r(t) a_i(t) \cos(\omega_c t) dt \,. \tag{2.5}$$

Assuming $\omega_c \gg T_s^{-1}$, the double frequency component can be ignored and Z_i can be rewritten as

$$Z_i = \sqrt{\frac{P}{2}} \cdot \left\{ b_{i,0}\, T + \sum_{\substack{k=1 \\ k \neq i}}^K [b_{k,-1}\, R_{k,i}(\tau_k) + b_{k,0}\, \hat{R}_{k,i}(\tau_k)] \cdot \cos \phi_k \right\} \tag{2.6}$$
$$+ \int_0^T n(t) a_i(t) \cos(\omega_c t) dt$$

where $R_{k,i}(\tau)$ and $\hat{R}_{k,i}(\tau)$ are defined as continuous-time, partial cross-correlation functions and expressed as [Pursley, 1977]

$$R_{k,i}(\tau) = \int_0^\tau a_k(t - \tau) a_i(t) dt \tag{2.7}$$
$$\hat{R}_{k,i}(\tau) = \int_\tau^T a_k(t - \tau) a_i(t) dt$$

The partial cross-correlations $R_{k,i}(\tau)$ and $\hat{R}_{k,i}(\tau)$ of (7) account for mismatch between symbol transition boundaries of the desired user and the kth user resulting from the kth user's time delay, and the kth user is an interferer. $R_{k,i}(\tau)$ correlates the end of the first symbol of the kth user that falls within the desired user's symbol interval, and $\hat{R}_{k,i}(\tau)$ correlates the beginning of the next subsequent symbol of the kth user which completes the rest of the desired user's symbol interval.

$R_{k,i}(\tau)$ and $\hat{R}_{k,i}(\tau)$ can be written in terms of the discrete aperiodic cross-correlation function $C_{k,i}(l)$ defined as [Pursley, 1977]

$$C_{k,i}(l) = \begin{cases} \displaystyle\sum_{j=0}^{N-1-l} a_j^{(k)} a_{j+l}^i & 0 \le l \le N-1 \\[2ex] \displaystyle\sum_{j=0}^{N-1+l} a_{j-l}^{(k)} a_j^i & 1-N \le l \le 0 \\[2ex] 0, & |l| \ge N. \end{cases} \qquad (2.8)$$

Substituting $C_{k,i}(l)$ of (2.8) into (2.7) results in the following alternate expressions for

$R_{k,i}(\tau)$ and $\hat{R}_{k,i}(\tau)$:

$$R_{k,i}(\tau) = C_{k,i}(l-N)T_c + [C_{k,i}(l+1-N) - C_{k,i}(l-N)] \cdot (\tau - lT_c)$$
$$\hat{R}_{k,i}(\tau) = C_{k,i}(l)T_c + [C_{k,i}(l+1) - C_{k,i}(l)] \cdot (\tau - lT_c) \qquad (2.9)$$

To establish the received signal-to-interference-plus-noise ratio (SINR), the

power of the noise plus interfering signals is calculated as the variance of Z_i defined in

(2.6) over one symbol interval and is given by [Pursley, 1977]

$$Var\{Z_i\} = \left(\frac{P}{4T}\right) \sum_{\substack{k=1 \\ k \ne i}}^{K} \int_o^T \left(R_{k,i}^2(\tau) + \hat{R}_{k,i}^2(\tau)\right) d\tau + N_o T/4$$
$$= \left(\frac{P}{4T}\right) \sum_{\substack{k=1 \\ k \ne i}}^{K} \sum_{l=0}^{N-1} \int_{lT_c}^{(l+1)T_c} \left(R_{k,i}^2(\tau) + \hat{R}_{k,i}^2(\tau)\right) d\tau + N_o T/4. \qquad (2.10)$$

In this case, the statistical expectation is taken with respect to variables ϕ_k, τ_k, and b_k.

Given the desired user's data remains constant over the symbol interval under

consideration, (2.10) only contains contributions from channel noise and MAI.

Substituting the $R_{k,i}(\tau)$ and $\hat{R}_{k,i}(\tau)$ expressions from (2.9) into (2.10) results in a

simplified variance given by

$$Var\{Z_i\} = \left(\frac{PT^2}{12N^3}\right) \left(\sum_{\substack{k=1 \\ k \ne i}}^{K} r_{k,i}\right) + N_o T/4 \qquad (2.11)$$

where [Pursley, 1977]

$$r_{k,i} = \sum_{l=0}^{N-1} \left\{ C_{k,i}{}^2 (l-N) + C_{k,i}(l-N)C_{k,i}(l-N+1) + C_{k,i}{}^2 (l-N+1) \right. \tag{2.12}$$
$$\left. + C_{k,i}{}^2 (l) + C_{k,i}(l)C_{k,i}(l+1) + C_{k,i}{}^2 (l+1) \right\}$$

The SINR is then determined by dividing $\sqrt{P/2T}$ by the rms noise of $\sqrt{Var\{Z_i\}}$

[Pursley, 1977], yielding:

$$SINR = \left\{ (6N^3)^{-1} \sum_{\substack{k=1 \\ k \neq i}}^{K} r_{k,i} + \frac{N_o}{2E_b} \right\}^{-1/2} . \tag{2.13}$$

To facilitate preliminary system design, it is shown in [Pursley, 1977] that for random

spreading sequences (2.13) simplifies to

$$SINR = \left\{ \frac{K-1}{3N} + \frac{N_o}{2E_b} \right\}^{-1/2} \tag{2.14}$$

where the *(K-1)/(3N)* term results from taking the expected value of the summation in

(2.13) given random sequences are employed. The equation for probability of bit error

(P_B) for a BPSK system without interferers is [Peterson, Ziemer, and Borth, 1995]

$$P_B = Q\left(\sqrt{\frac{2E_b}{N_o}} \right). \tag{2.15}$$

Using (2.14) an approximate P_B value can be calculated for a DS/SSMA system using

(2.15) as

$$P_B = Q(SINR). \tag{2.16}$$

It is useful to note that if only one user is present ($K = 1$), (2.14) identically simplifies to

the *Q*-function argument of (2.15), the correct result for the single user case.

2.2.1 BER for Asynchronous and Synchronous Systems.

Using characteristic functions for DSSS signals, Gerantiotis and Ghaffari have

verified the BER approximation results of Pursley [Gerantiotis and Ghaffari, 1991;

Pursley, 1977] and redefined (2.15) via derivation as

$$SINR = \left\{ \frac{K-1}{N} m_\psi + \frac{N_o}{2E_b} \right\}^{-1/2} \tag{2.17}$$

where $m_\psi = 1/3$ for rectangular pulses and is defined as

$$m_\psi = T_c^{-3} \int_0^{T_c} R_\psi^2(\tau) d\tau = T_c^{-3} \int_0^{T_c} \hat{R}_\psi^2(\tau) d\tau . \tag{2.18}$$

In (2.18) R_ψ and $\hat{R}_\psi$ are partial autocorrelation functions of the pulse shape defined as

[Gerantiotis and Ghaffari, 1991]

$$\hat{R}_\psi(\tau) = \int_\tau^{T_c} \psi(t)\psi(t-\tau) dt$$

$$R_\psi(\tau) = \hat{R}_\psi(T_c - \tau). \tag{2.19}$$

This derivation clarifies that (2.14) is specifically applicable to the SINR approximation

for *asynchronous* signals using rectangular shaped chip waveforms [Gerantiotis and

Ghaffari, 1991]. Continuing with this development a BER approximation for

synchronous DS/SSMA systems using rectangular shaped chip waveforms is found to be

[Gerantiotis and Ghaffari, 1991]

$$SINR = \left\{ \frac{K-1}{2N} + \frac{N_o}{2E_b} \right\}^{-1/2} . \tag{2.20}$$

It has been shown that BER can be approximated for *asynchronous* and *synchronous*

systems by substituting (2.14) and (2.20), respectively, into the *Q*-function argument of

(2.16) [Gerantiotis and Ghaffari, 1991].

2.2.2   Accuracy of BER Approximations.

Using the SINR approximations of (2.14) and (2.20), and their own characteristic function approximations for SINR, Gerantiotis and Ghaffari calculated the BER for both the synchronous and asynchronous cases using both random aperiodic spreading codes and deterministic *m*-sequence codes [Gerantiotis and Ghaffari, 1991]. For all cases considered, parameter values of $K = 3$ and $N = 31$ were used with rectangular-shaped chip waveforms. Their results for PSK modulation are shown in Table 2.1. All values with superscript G are approximations resulting from (2.14) and (2.20), and those without a superscript result from Gerantiotis and Ghafferi's characteristic function approximation.

Table 2.1. Error Probabilities for DS/SSMA Systems ($K = 3$, $N = 31$)
[Gerantiotis and Ghaffari, 1991]

| $\dfrac{E_b}{N_o}$ | Random Sequences | | | | m-Sequences | | | |
| | Synchronous | | Asynchronous | | Synchronous | | Asynchronous | |
	$\bar{P}_e^G$	$\bar{P}_e$	$\bar{P}_e^G$	$\bar{P}_e$	$P_{e,1}^G$	$P_{e,1}$	$P_{e,1}^G$	$P_{e,1}$
8	1.37×10^{-3}	1.57×10^{-3}	8.15×10^{-4}	9.13×10^{-4}	1.03×10^{-3}	8.04×10^{-4}	8.13×10^{-4}	8.49×10^{-4}
10	2.45×10^{-4}	3.71×10^{-4}	9.21×10^{-5}	$1.44 \times 10^{-}$	1.43×10^{-4}	5.99×10^{-5}	9.17×10^{-5}	1.08×10^{-4}
12	3.77×10^{-5}	9.73×10^{-5}	7.07×10^{-6}	2.55×10^{-5}	1.54×10^{-5}	1.28×10^{-6}	7.02×10^{-6}	1.08×10^{-5}
14	5.98×10^{-6}	3.19×10^{-5}	4.46×10^{-7}	7.69×10^{-6}	153×10^{-8}	3.96×10^{-9}	4.40×10^{-7}	1.05×10^{-6}
16	1.16×10^{-6}	1.35×10^{-6}	3.01×10^{-8}	44.1×10^{-7}	1.78×10^{-7}	5.98×10^{-12}	2.95×10^{-8}	1.06×10^{-7}

Four observations from Table 2.1. are important for this research. First, the analytical BER approximations based on (2.14) and (2.20) were found to be optimistic for random codes in the synchronous and asynchronous cases, and for deterministic codes in the asynchronous case. However, the BER approximations based on (2.14) and (2.20) are very conservative for synchronous systems with deterministic spreading codes. Second, for random sequences the asynchronous case outperforms the synchronous case

due to the increased randomization and averaging provided by the asynchronous case [Gerantiotis and Ghaffari, 1991]. Third, for deterministic sequences the synchronous case outperforms the asynchronous case, reflecting the fact that deterministic codes are typically designed to maximize the ratio of autocorrelation to cross-correlation for the synchronous case [Peterson, Ziemer, and Borth, 1995]. Finally, in both synchronous and asynchronous cases the deterministic codes outperformed the random codes. This result is more pronounced in the synchronous case [Gerantiotis and Ghaffari, 1991].

2.3   Chip Waveform Shaping

To reduce MAI effects in DS/SSMA systems, the impact of using chip waveform shapes other than rectangular is explored. This approach to reducing MAI is based on decreasing the cross-correlation statistics between spreading codes as a result of variation in the chip waveform shape. The pulse shapes considered in this research include rectangular, half-sine, raised-cosine, Blackman, Kaiser, and Lanczos. As defined in [Kok and Do, 1997] each of these pulse shapes can be analytically represented as follows:

a) Blackman

$$\psi(t) = k\left[0.42 - 0.5\cos\left(\frac{2\pi t}{T_c}\right)\right]u(t) \tag{2.21}$$

b) Half-sine

$$\psi(t) = \sqrt{2}\sin\left(\frac{\pi t}{T_c}\right)u(t) \tag{2.22}$$

c) Kaiser

$$\psi(t) = k \, \frac{I_0\left(\beta\pi \sqrt{1 - \left(\frac{t - T_c/2}{T_c} \right)} \right)}{I_0(\beta\pi)} \, u(t) \tag{2.23}$$

d) Lanczos

$$\psi(t) = k \left[\frac{\sin(2\pi t)}{2\pi t} \right]^2 u(t) \tag{2.24}$$

e) Raised-cosine

$$\psi(t) = \sqrt{\frac{2}{3}} \left[1 - \cos\left(\frac{2\pi t}{T_c} \right) \right] u(t) \tag{2.25}$$

where T_c is the chip duration, $u(t)$ is the unit step function, I_0 is a zero-order modified Bessel function, and k is a real-valued scaling constant.

2.3.1   Random Coding with Chip Waveform Shaping

Using the system model from (2.1-2.4) and chip waveform definitions of (2.21), (2.22), and (2.25), Lehnert and Cho investigated the impact of waveform shaping on DS/SSMA system performance using aperiodic random spreading sequences [Lehnert, 2002; Cho and Lehnert, 1999]. All results are for systems with $K = 9$ total users and $N = 31$ for the processing gain. Each user has equal energy, and the time delays of the 8 interfering users are normalized to one symbol interval relative to the desired user and modeled as uniformly distributed random variables.

The average BERs are computed using (2.14), where a conditional Gaussian approximation (CGA) is used to estimate the interference term when estimating SINR, which is the argument of (2.16). The CGA is used here in place of standard Gaussian approximation instrumental in the derivation of (2.14) and (2.20) because the CGA more

accurately approximates the MAI effect on communication performance in a DS/SSMA

system [Cho and Lehnert, 1999]. Based on the CGA assumption, the Blackman

waveform shape was predicted to provide the best performance relative to the half-sine,

raised-cosine, and rectangular shapes. Results of this estimation process are reproduced

in Fig. 2.1. for half-sine, raised-cosine, and Blackman pulse shapes at E_b/N_o values

ranging from 0 to 30 dB. The data clearly illustrates the Blackman chip waveform shape

outperforms all other waveform shapes considered.
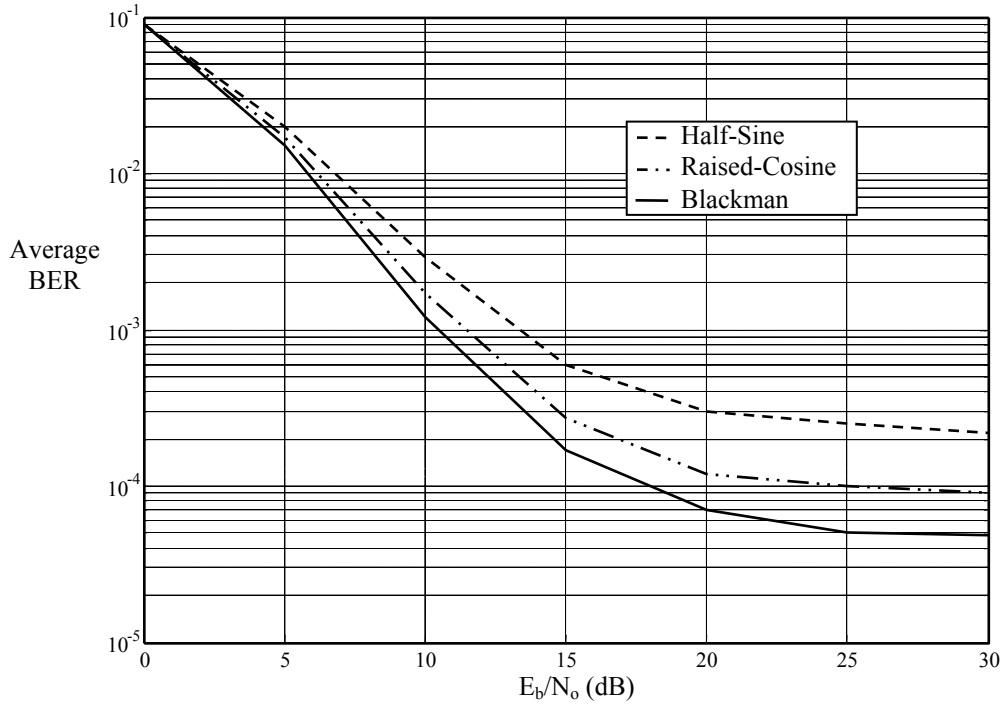
Figure 2.1. Random Coding with Chip Waveform Shaping: Average BER for half-sine, raised-cosine and Blackman shapes [Cho and Lehnert, 1999].

2.3.2   Gold-Coding with Chip Waveform Shaping.

The work of Kok and Do analytically explores the impact of using chip waveform

shapes on DS/SSMA BER performance. All shapes evaluated are time-limited and

normalized for the analysis. As in the random coding case, the chip waveform shapes

evaluated were rectangular, Blackman, half-sine, Kaiser, Lanczos, and raised-cosine.

Using Gold codes of length 127 and 511, while increasing number of users from 5 to 70, the SINR is calculated for each pulse shape case based on aperiodic autocorrelation functions of (2.8) [Kok and Do, 1997]. This SINR value is then used as the argument of (2.16) to estimate BER for each case, where the Q-function is evaluated with the algorithm described in [Parl, 1980]. Results show that all non-rectangular chip waveform shapes outperform the rectangular case. Specifically, the system employing the Lanczos shape achieves the best BER, followed in order by Kaiser, Blackman, raised-cosine, half-sine, and rectangular [Kok and Do, 1997].

2.4   Summary

The goal of this research is to characterize the effects of pulse shaping on DS/SSMA through simulation. This chapter provides a description of the DSSS system design and the MAI term caused by multiple simultaneous transmitters. An approximation for expected BER for both synchronous and asynchronous systems is derived. Discussion of the accuracy of this BER approximation enables a more informed comparison to the results of this research. Finally, the basics of pulse shaping as a means of BER improvement is discussed for random and Gold codes, and examples of approximated performance improvement are provided as a baseline for comparison to this research.

III. Methodology

3.1   *Problem Definition*

This research characterizes the multiple access interference (MAI) for direct-sequence, spread-spectrum multiple access (DS/SSMA) systems through simulation, and investigates the impact of code selection, pulse shaping, and code length on bit error rate (BER). Multiple access capabilities are measured in terms of BER for given signal-to-noise ratios as related to the average energy per bit of the desired user (E_b) divided by the background noise power spectral density (N_o). The effects that parametric changes have on multiple access BER are also considered. Previous analytical work characterizes BER in terms of E_b/N_o, the number of users, and the ability of pulse shaping over a chip interval to reduce MAI levels. This research develops and uses a Matlab$^{®}$ simulation to model DS/SSMA system performance and to investigate the impact code selection and pulse shaping has on BER. Additionally, the simulation provides a means for predicting the impact of increasing code length in systems employing random and Gold codes in conjunction with the pulse shaping. Simulation results are verified against the single user communications baseline and research results outlined in Chapter 2. Once verified, the code is used to characterize the impact of synchronization, code selection, code length, and pulse shaping on BER for a DS/SSMA system.

3.2   *Approach*

Simulation results are reported as BER vs. E_b/N_o curves for both synchronous and asynchronous systems with multiple users, using random binary and Gold coding techniques, and implementing chip waveform shaping. Each system user transmits

communication waveforms using binary phase shift keyed (BPSK) baseband modulation and coded spreading waveforms consisting of the desired pulse shape over the chip intervals. All signals are assumed to be received with equal power and additive white Gaussian noise (AWGN) power levels are adjusted to achieve the desired E_b/N_o value. Each chip interval has duration T_c and spreading codes are periodic with length N. Thus, a collection of coded chip waveforms comprises one symbol duration T_s and equals NT_c. Simulation validation consists of two stages, including 1) single user results are generated and compared with (2-15) and 2) results are compared to the analytical findings of Gerantiotis and Ghaffari [Gerantiotis and Ghaffari, 1991]. Following code validation, simulation results are used to explore the impact of pulse shaping, code selection, and code length on BER for DS/SSMA systems.

3.3   *System Boundaries*

The system of interest is a direct-sequence spread-spectrum (DSSS) receiver using BPSK modulation (data and spreading) for multiple access communications. Figure 3.1 shows a diagram of the physical system which consists of a receive antenna, a radio frequency (RF) bandpass filter, a despreading mixer, an intermediate frequency (IF) bandpass filter, and the data phase demodulator. This research simulates DSSS receiver performance in a multiple access environment using Matlab[®]. Each user transmits communication waveforms consisting of 1) BPSK data modulation and 2) BPSK spreading modulation having the desired pulse shape over each chip interval. All signals are received with equal power and AWGN power levels are adjusted to achieve the desired E_b/N_o value. Each chip interval has duration T_c and spreading codes are periodic

with length N such that the symbol duration T_s equals NT_c. The ability of the

despreading mixer, IF filter and demodulator to correctly estimate communication

symbols (which are subsequently mapped to bits) from the desired user, in the presence

of other system users, is the focus of this research.
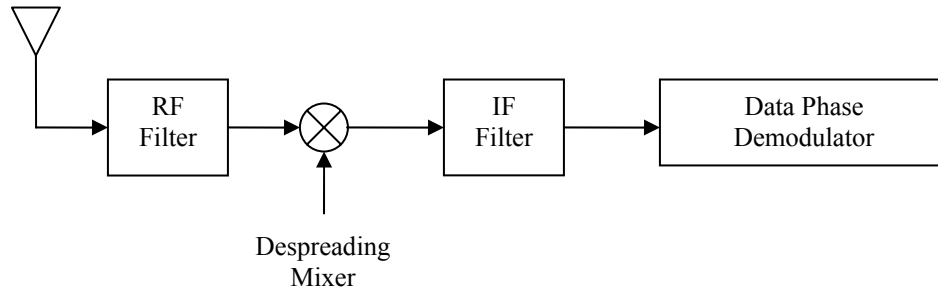
Figure 3.1. BPSK DSSS Receiver [Peterson, Ziemer, and Borth, 1995].

3.4   System Services

DSSS receivers are designed to receive signals and estimate bits sent from the

desired user while rejecting interfering signals of all undesired users. By doing so, DSSS

receivers enable multiple users to simultaneously communicate within the same spectral

region. Over each received symbol interval, the receiver estimate results in one of two

possible outcomes, either 1) a symbol (bit) is received and estimated correctly (correct

condition), or 2) a symbol (bit) is received and estimated incorrectly (error condition).

The source of making errors per outcome 2 can be either background noise or

interference from multiple users (MAI). Although AWGN background noise is present

for all simulations and analyses conducted in this research, the primary focus of this work

is on errors due to MAI.

3.5   Performance Metrics

The introduction of errors in signal detection and estimation for signals received through a physical medium is inevitable and these errors are even more likely to occur with MAI present. The percentage of errors occurring in DSSS receiver processing is quantified using bit error rate (BER or P_B), defined as the number of demodulated bits that are *incorrectly* estimated at the receiver divided by the total number of bits transmitted. The required system BER establishes a lower bound on the percentage of errors a receiver can make and still correctly function. This research uses BER as the primary metric to measure system performance. Fundamental to BER determination is the ratio of available signal energy E_b to total interfering energy. In this case, the total interfering energy consists of channel noise energy, as established by N_o, and interference due to multiple users (N_I). The resultant energy-to-noise ratio can be expressed as

$$\frac{E_b}{N_o + N_I}.$$

(3.1)

For a system using BPSK data modulation, BER is directly related to this ratio and is

$$P_B = Q\left(\sqrt{\frac{2E_b}{N_T}}\right)$$

(3.2)

where Q is the complimentary error function and $N_T = N_o + N_I$ [Lehnert, 2002]. The existing analytical approximations estimate E_b/N_T values which are converted to BER for comparison with the simulated results using (3.2).

3.6   Parameters

3.6.1   System

System parameters which affect the demodulator's ability to correctly estimate symbols from the desired user include:

- Channel propagation characteristics

- Receiver antenna gain (or loss)

- Spectral width and center frequencies of the RF and IF filters

- User of interest (one of total system users)

- Type of spreading code used

These system parameters reflect the physical receiver as simulated. As indicated earlier, the channel is assumed to be an ideal AWGN channel with a two-sided constant power spectral density of $N_o/2$. Receive antenna gain affects the total amount of received power (signal, noise, and interference). For this work, it is assumed that received antenna gain affects all signal, noise, and interference terms equally and is therefore ignored. The spectral width and center frequencies of the RF and IF bandpass filters impact the amount of signal power processed by the system. Filter center frequency is determined by which user is designated as the desired user, and the filter's spectral width and shape are determined by the spreading code employed. For all simulations, it is assumed that the center frequency and spreading code are perfectly selected for the desired user. The type of spreading codes employed impact the level of MAI due to varying autocorrelation and cross-correlation characteristics of different codes. Random codes are employed for both single user and multiple access validation tests. This is appropriate for the single user validation because single user performance is independent of code selection. Random

codes are employed for multiple access validation to permit direct comparison with the system setup used by Gerantiotis and Ghaffari [Gerantiotis and Ghaffari, 1991]. Both random and Gold codes are used to expand the research through simulation and analysis of results obtained with chip waveform shaping and varying code lengths.

3.6.2   Workload.

Many workload parameters exist and potentially impact receiver BER. These parameters include:

- Signal structure (i.e., transmitted symbol shape)

- Signal power

- Thermal channel noise

- Time delay of interferers relative to the desired user

- Number of chip intervals (T_c) per communication symbol interval (T_s)

- Number of code periods per symbol interval

- Number of system users

- Shape of chip waveforms

The transmitted signals for this research are baseband waveforms consisting of uniformly distributed random strings of positive and negative ones that represent random binary waveforms. For a given simulation, all users have the same signal power. Thermal channel noise is modeled as AWGN. Relative time delays for interfering users impact BER by changing the cross-correlation levels between the interfering codes and the desired user's code. The number of chips per symbol interval is commonly related to processing gain, G, and BER improves (decreases) as G increases [Peterson, Ziemer, and Borth, 1995]. For all simulations exactly one code period is contained within each

symbol interval. In DS/SSMA systems MAI levels increase as the number of users increases and BER increases. Finally, chip waveform shape impacts code cross-correlation values such that the dispreading mixer output increases (or decreases) and results in a corresponding increase (or decrease) in BER.

3.7   Factors

Of the parameters listed above, only the following factors are varied for this research: thermal noise power, time delay of interferers relative to the desired user, number of users, spreading code length, and chip waveform shape. Signal power is accounted for in E_b/N_o with desired values achieved by holding E_b constant and varying the noise power spectral density N_o. E_b/N_o values of 0 to 10 dB are simulated for all cases considered unless otherwise specified. The time delay of interferers relative to the desired user is varied between two states, 1) time delay equals zero for all users in a *synchronous* system, or 2) time delays are uniformly distributed random variables in the range of $[0, T_s]$ for an *asynchronous* system. The number of system users directly impacts MAI levels and 1) varies based on availability of analytic approximations for comparison, or 2) is held constant at $K = 9$ to permit comparison of simulated results with those presented by Cho and Lehnert [Cho and Lehnert, 1999]. Spreading code length, N, varies from 31 to 511 to illustrate the impact of code length variation on BER. Finally, chip waveform shape varies based on the analytical approximations presented by Lehnert, and Kok and Do. Chip waveform shapes considered include, rectangular, Blackman, and Lanczos as defined in (2.22) through (2.25) [Lehnert, 2002; Kok and Do, 1997]. Figure 3.2 illustrates the normalized Blackman and Lanczos pulses.
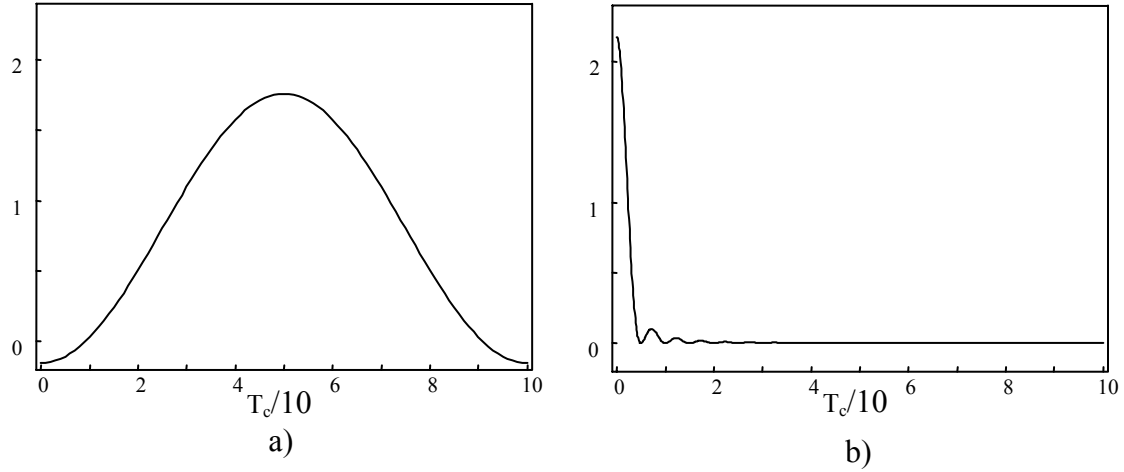
Figure 3.2. Normalized Chip Waveform Shapes. a) Blackman and b) Lanczos
[Kok and Do, 1997].

3.8   Evaluation Technique

All evaluations are based on comparison of BER results. The BERs used are

either calculated based on the BPSK single-user BER expression of (2.15), approximated

based on analytical expressions using (2.14) and (2.20) in (2.16), observed from previous

research as outlined in Chapter 2, or estimated through simulation from this research.

Simulation of the physical system provides all experimental data used for evaluation.

Simulations are conducted in two phases. First, the code is validated against known

analytical equations for a single user and results of previous research for the case of

multiple access. Additional simulations establish BER variation resulting from random

codes verses Gold codes, multiple chip waveform shapes, and spreading code length.

The impact of factors is determined by comparing BER curves for systems employing

different factors in the setup or workload. Using two simulation phases in this research

enables code verification before exploring the impact of additional factors on BER. This

verification supports the validity of the new simulated cases.

3.9   Workload

The workload varies appropriately for the setup of each simulation phase.

Workloads in all phases include background thermal noise. Signal power levels used

provide adequate resolution to create BER plots for comparison with approximations,

previous research, and new simulation results. Selection of time delay and processing

gain reflect previous research into BERs for DS/SSMA systems. Selection of three chip

waveform shapes enables quality comparison with previous research while limiting the

computational complexity.

3.10   Experimental Design

3.10.1   System Design.

Matlab® simulations are used to produce BER vs. E_b/N_o results for both

synchronous and *asynchronous* systems with multiple users, using random and Gold

codes, and implementing chip waveform shaping. Each user transmits communication

waveforms having BPSK baseband data modulation and coded spreading modulation

using the desired pulse shape over the chip intervals. All signals are transmitted

(received) with equal power and noise power is varied to achieve desired E_b/N_o values.

Each chip interval has duration T_c and codes are periodic with length N such that symbol

duration T_s equals NT_c. All spreading pulse shapes are scaled to normalize transmitted

power to that of a system using rectangular chip waveforms (rectangular waveform

shaping is the baseline for comparing other waveform shapes). Random spreading codes are generated as aperiodic, uniformly distributed sequences of positive and negative ones. Gold codes are generated as outlined in [Peterson, Ziemer, and Borth, 1995] using an initial register state of all ones; generator polynomials $[45]_8$ and $[75]_8$ are used to generate $N = 31$-length codes and $[1021]_8$ and $[1461]_8$ are used to generate $N = 511$-length codes.

3.10.2 Code Validation.

Code validation is conducted in two stages. First, using only rectangular chip waveforms, simulation results for $K = 1$ transmitter and $N = 31$ length codes (processing gain) are compared to the single user baseline calculated using (2.15) to validate code performance for the simplest case. This comparison is repeated using both random and Gold codes to validate both code implementations in the simulation code. Second, using only random codes and rectangular chip waveforms, the trends of simulations using $K = 3$ and $N = 31$ are compared to findings of Gerantiotis and Ghaffari relative to the analytical approximations for signal-to-interference plus noise (SINR) in (2.15) and (2.19).

3.10.3 Waveform Shaping and Code Length Tests.

Following validation, simulated BER results are compared for chip waveforms having rectangular, Blackman, and Lanczos shapes as defined in (2.21) through (2.25). Using the verified simulation code, BER is estimated for spreading codes comprised of random codes and one full-period of the 31-length Gold codes. A collection of "best-case" and "worst-case" Gold codes are used. Here, "best-case" Gold codes are selected as the combination of Gold codes having the *lowest* cross correlation values with the desired signal. Likewise, "worst-case" Gold codes are selected as the combination of Gold codes having the *highest* possible cross correlation values with the desired signal.

Finally, to illustrate the impact of code length on MAI, BER is simulated for systems using $N = 511$-length Gold codes and compared to aperiodic random spreading codes having $N = 511$ chips per symbol interval.

3.10.4 Simulation Length.

Each bit of data that is coded, transmitted, despread, and estimated in the receiver represents an independent test of the system's ability to accurately demodulate the desired user's data. Subsequent bit transmissions and estimations represent experimental repetitions. Each simulation runs until 300 bits are estimated in error. The total number of bits transmitted, n, represents the number of experimental trials in the simulation. Based on BER values estimated in this research, using a 95% confidence interval and running simulations until 300 bit errors are detected, the sample mean observed through simulation will vary from the actual population mean due to variation by an amount r that is at most approximately $\pm 11\%$ of the actual mean of the simulation [Canadeo, 2003]. Variation r is defined as

$$r = z\sqrt{\frac{P_B(1-P_B)}{n}} \tag{3.3}$$

where $z = 1.96$ for the 95% confidence interval and P_B is the effective BER defined as the ratio of the number of bits in error to n [Jain, 1991].

The above analysis is based on assuming errors are independent and normally distributed. The properties of an AWGN channel and the central limit theorem indicate that these errors should be independent and normally distributed. To test these assumptions, control tests are run with the number of users set at $K = 5$ and $K = 10$ using E_b/N_o values of 0 and 5 dB. Each test is repeated fifteen times, providing fifteen independent BER values for each number of users considered at a given energy profile.

The experimental error, e_j, is calculated as the difference between measured BER and the trial average, and this error is plotted against the normal quantiles, defined as

$$x_i = 4.91 \left[q_i^{0.14} - \left(1 - q_i\right)^{0.14} \right] \qquad (3.4)$$

where

$$q_i = \frac{i - 0.5}{n} \qquad (3.5)$$

i is the number of the trial being plotted, and n is the total number of trials. The linearity of this plot verifies the assumption of normally distributed errors. Additionally, the error, e_j, is plotted against the average BER results. The lack of a trend in the errors verifies the independence of errors [Jain, 1991].

3.11   Analyze and Interpret Results

Simulation results are used to compile BER vs. E_b/N_o curves for both phases of the experiment. Relative benefits of tested factors are not easily determined by direct comparison of BER plots alone due to variation caused by randomness in some of the factors, e.g. the particular noise realization. To make statistically significant comparisons, error bars are included in the BER curves. Error bar plots based on the variation values calculated by (3.3), where the error bars are plotted an amount r above and below the observed BER, provide bounds for actual BER mean values to a 95% confidence level and allow more meaningful results through direct comparison [Jain, 1991]. For BER values to be "different" in a statistically significant manner, the error bars of two BER cases being compared cannot overlap. Taking this into consideration,

direct comparison of BER curves yields meaningful information about the relative performance enhancements of factors under consideration.

Both phases of testing utilize direct comparison with error bars present to determine if results are either significantly similar for validation or significantly different for identifying performance enhancements.

3.12   Summary

This chapter outlines the experimental setup required for 1) validating the DS/SSMA receiver simulation and 2) comparing the relative performance enhancements of selected factors. The system's service is the transmission of data bits and metrics are the BER vs. E_b/N_o curves including error bars.

All system components are simulated, with simulations conducted in two phases. The first phase validates the analytical models of the DS/SSMA receiver and the second phase illustrates the relative impact of chip waveform shaping, code selection, and code length on BER.

The analysis of results compares the BER curves to expected theoretical BER, previously reported BER results, or other experimental results of this research. Relative impact on BER is the expected result of these comparisons.

IV. Results and Analysis

4.1   Validation Testing

Simulation code is validated in two stages. First, simulated bit error rate (BER)

for a single user in the presence of noise is compared to theoretical BER calculations

from (2.14). Second, the trend of simulated BER for multiple users is compared to

results of Geraniotis and Ghaffari [Geraniotis and Ghaffari, 1991].
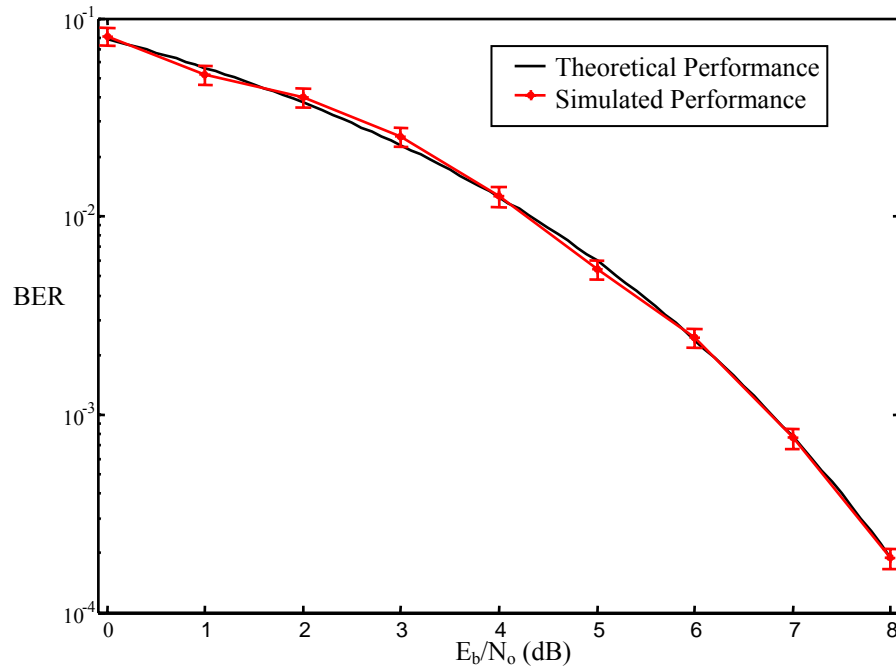
Figure 4.1. Validation of Single User with Random Code Sequence

4.1.1   Single User Validation.

Using only rectangular chip waveforms, BER for a single user with random

spreading codes is simulated for an additive white Gaussian noise (AWGN) channel.

Error bars are generated as outlined in Chapter 3 and reflect the 95% confidence interval

for simulated results. Simulation results for this single user case are shown in Fig. 4.1

along with theoretical BER calculated per (2.14). As shown, the theoretical BER curve

lies within simulated error bar bounds, indicating the simulated results are consistent with theoretical performance. Figure 4.2 shows the same relationship for the case where Gold code spreading sequences are used in place of the random sequences. Once again, the theoretical result falls within simulated error bar bounds indicating simulated BER with Gold coded sequences is consistent with theoretical performance.
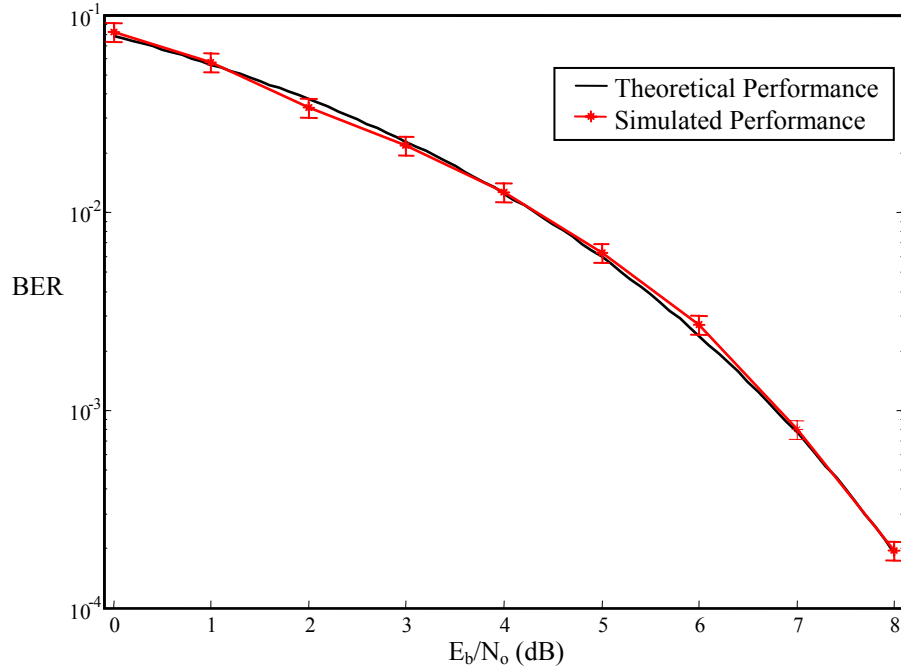
Figure 4.2. Validation of Single User with Gold Code Sequences

4.1.2   Multiple User Validation.

Following the system setup of Geraniotis and Ghaffari, the trend of simulated BER results are compared to the data from Table 2.1 [Geraniotis and Ghaffari, 1991]. The system includes $K = 3$ users, a spreading code of length $N = 31$ chips, rectangular shaped chip waveforms, and aperiodic random spreading codes. Simulations are run for both *synchronous* and *asynchronous* networks and results are compared with analytical approximations of (2.19) and (2.15), respectively.

Figure 4.3 shows simulated results for *synchronous* users using E_b/N_o values of 8, 10, 12, 14, and 16 (dB). Analytic approximations from (2.19) are also shown for the same energy profiles. Simulation results indicate performance which is poorer (higher BER) than predicted by the analytical approximations. The results presented here are consistent with the findings of Geraniotis and Ghaffari [Geraniotis and Ghaffari, 1991]. Given the analytic approximation falls well outside the error bar bounds for all values of E_b/N_o considered, these results are statistically significant at the 95% confidence level.
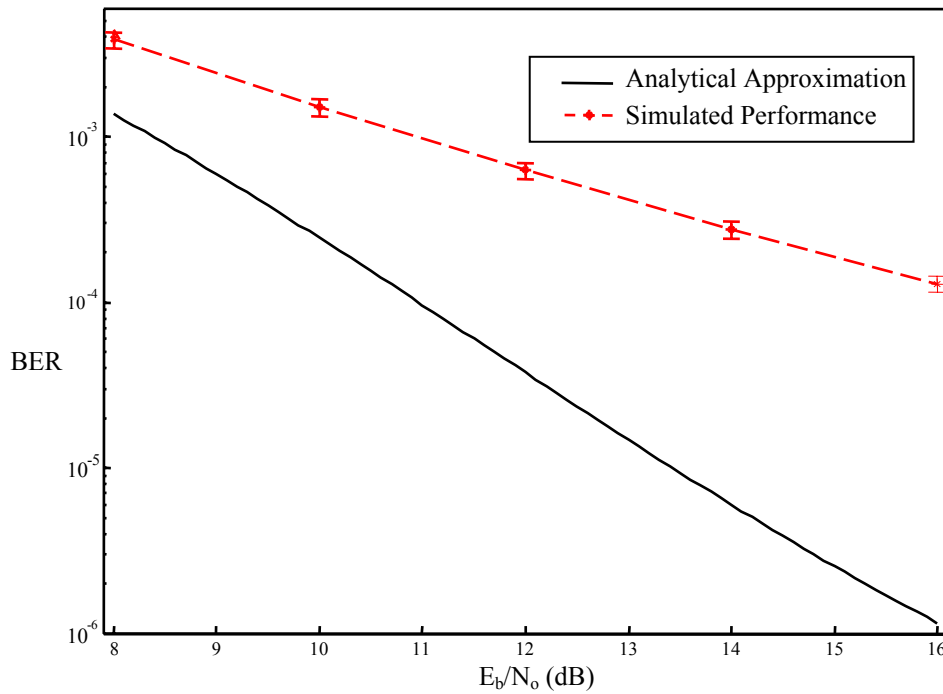
Figure 4.3. Simulated vs. Analytic BER for *Synchronous* DS/SSMA Systems Using Random Code Sequences.

A comparison of calculated BER from (2.15) with simulation results of an *asynchronous* system, using E_b/N_o values of 8, 10, 12, 14, and 16 (dB), is shown in Fig. 4.4. Once again, the data indicates that simulated BER under performs (higher BER) the approximations and are consistent with Geraniotis and Ghaffari's findings that the analytical approximation is "optimistic" for the asynchronous case [Geraniotis and

Ghaffari, 1991]. Again, the error bars add statistical significance to this result at the 95% confidence level.
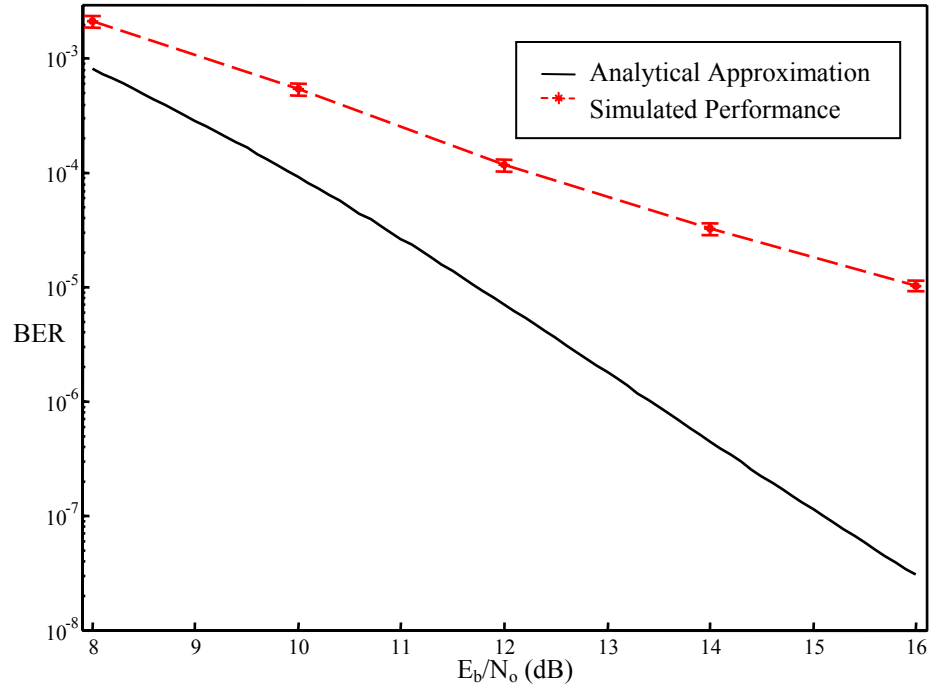
Figure 4.4. Simulated vs. Analytical BER for *Asynchronous* DS/SSMA Systems Using Random Code Sequences.

4.1.3 Error Analysis

Error analysis was conducted to verify the accuracy of the synchronous simulation model. Using only rectangular pulse shapes and random code sequences, simulations for $K = 5$ and 10 users at E_b/N_o values of 0 and 5 (dB) are repeated fifteen times. Experimental error, e_j, is calculated as the difference between the observed BER for each trial and the average of fifteen trials. Values of e_j are plotted against the normal quantiles as determined by (3.4) and (3.5) in Fig. 4.5, Fig. 4.6, Fig. 4.7, and Fig. 4.8. The nearly linear nature of these plots verifies the errors are normally distributed [Jain, 1991].
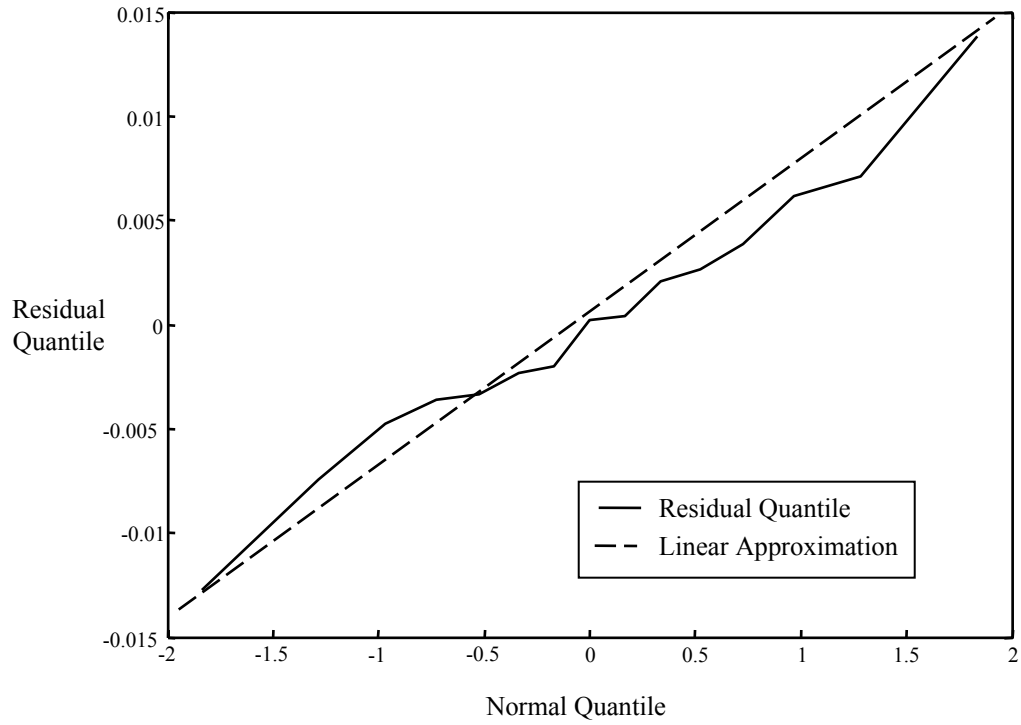
Figure 4.5. Quantile-Quantile Plot for $K = 5$ Users and $E_b/N_o = 0$ (dB).
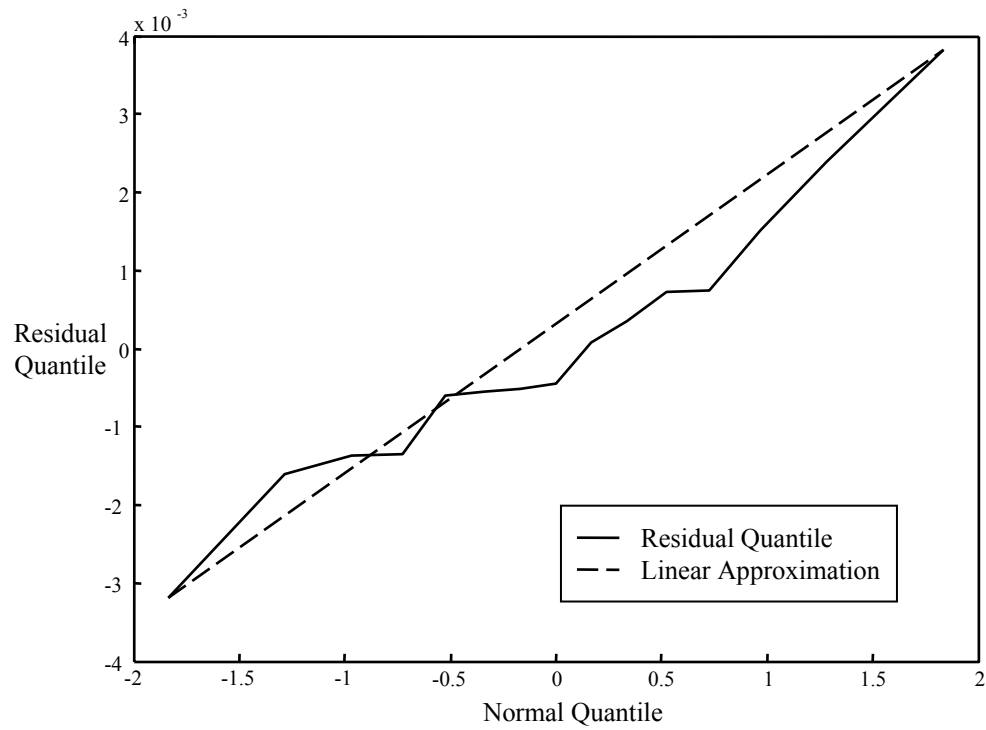
Figure 4.6. Quantile-Quantile Plot for $K = 5$ Users and $E_b/N_o = 5$ (dB).
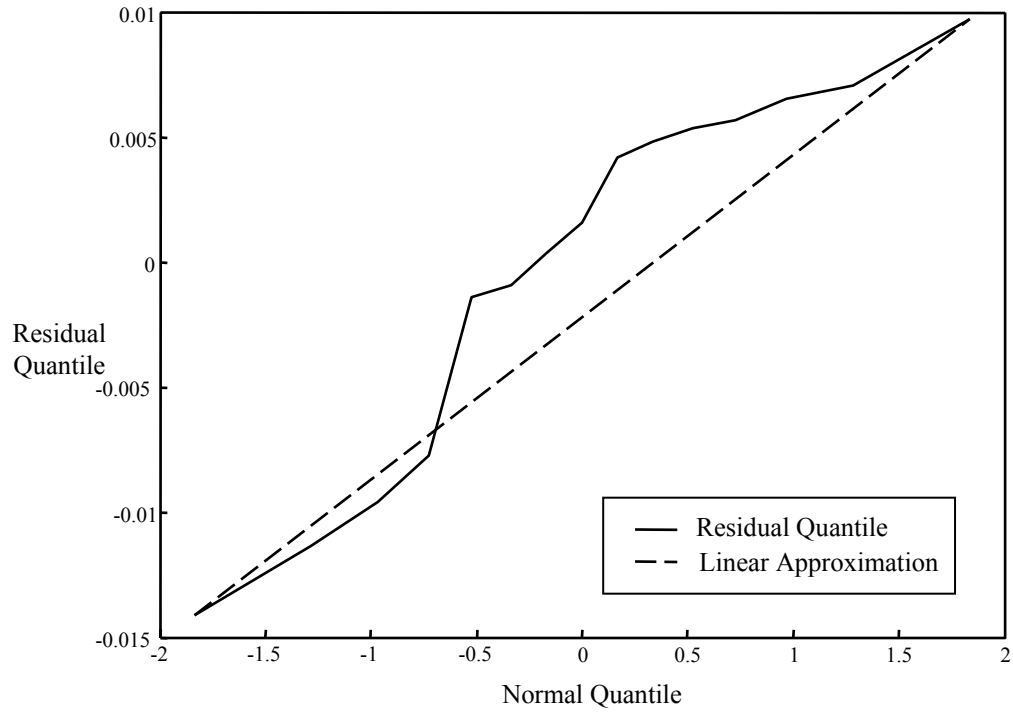
Figure 4.7. Quantile-Quantile Plot for $K = 10$ Users and $E_b/N_o = 0$ (dB).
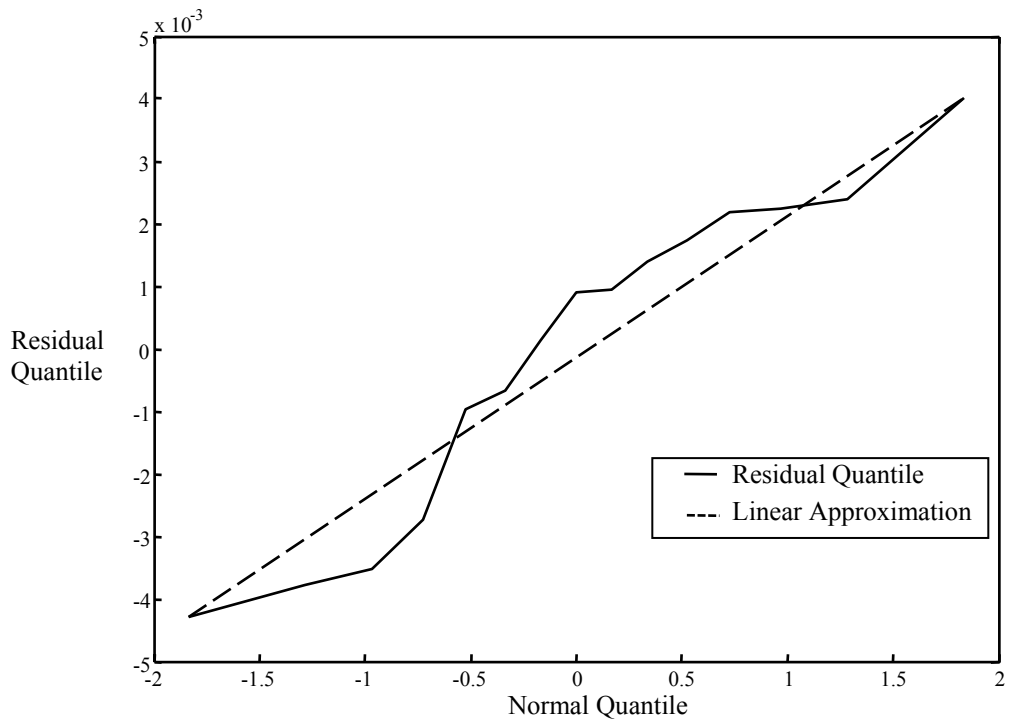
Figure 4.8. Quantile-Quantile Plot for $K = 10$ Users and $E_b/N_o = 5$ (dB).

Figure 4.9 and Fig. 4.10 show the residual quantity, e_j, plotted about zero, which represents the average BER from the trials. All residual errors are uniformly distributed about the trial mean indicating a lack of bias. Note: The first fifteen points are for $E_b/N_o = 0$, and the next fifteen points are for $E_b/N_o = 5.0$ in Fig. 4.9 and Fig. 4.10. As E_b/N_o increases BER decreases and residual variation from the mean also decreases. This is consistent with the prediction that the errors would fall within $\pm 11\%$ of the mean BER [Jain, 1991].
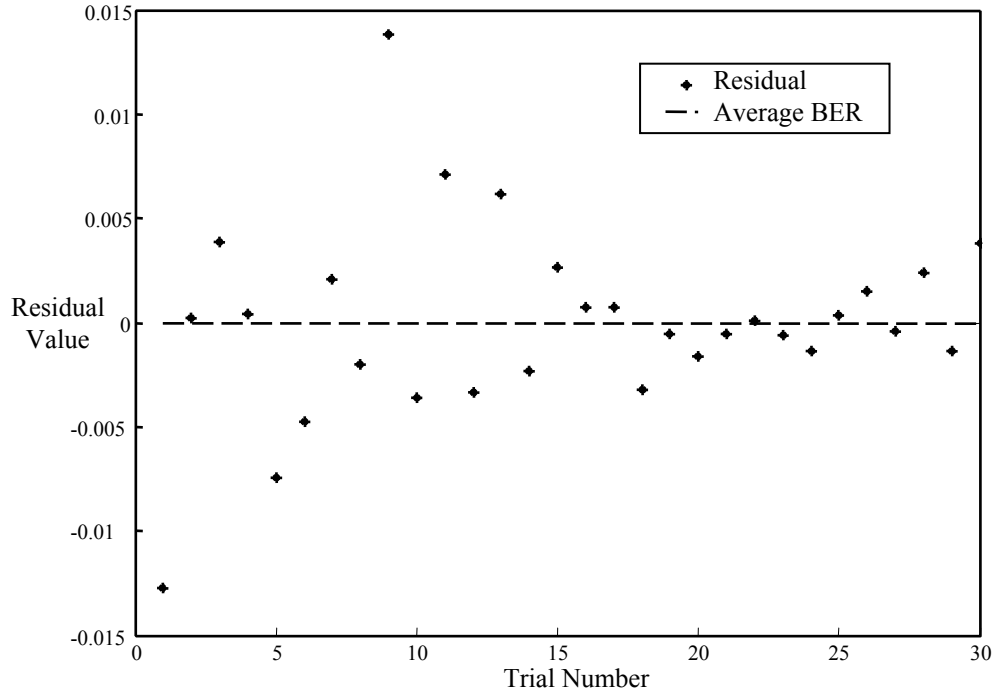
Figure 4.9. Residual Plot for $K = 5$ Users.
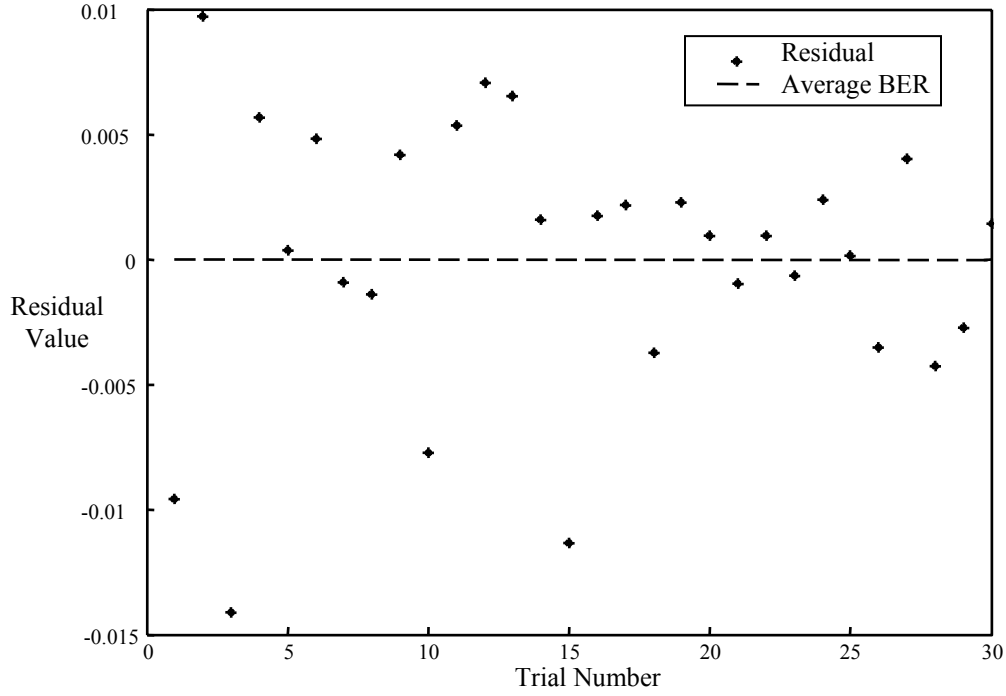
Figure 4.10. Residual Plot for $K = 10$ Users.

4.1.4 Conclusions.

The two stage validation tests illustrate that the fundamental system model and coding in the Matlab® simulation are sufficiently accurate to perform comparative analyses. Error analysis plots indicate the system model as simulated is free from undesired bias. The results of these validation and error analysis tests provide a solid basis for establishing model accuracy and support the experimental results of this research.

4.2 Impact of Gold Codes

Previous research into the impact of multiple access interference (MAI) on DS/SSMA system BER has focused primarily on systems using aperiodic random spreading code sequences, or *m*-Sequences [Cho and Lehnert, 1999; Geraniotis and

Ghaffari, 1991]. Results of simulation for a DS/SSMA system with $K = 9$ users, $N = 31$ length codes, using rectangular chip waveforms and employing Gold code sequences is compared to BER results for aperiodic random spreading codes for both *synchronous* and *asynchronous* systems.

Figure 4.11 shows BER curves for both best-case and worst-case Gold codes (relative to random codes) for the *synchronous* system. The best-case Gold coded system far outperforms the randomly coded system, while the worst-case Gold coded system under performs the randomly coded system. Error bars are included in Fig. 4.11 and indicate the differences noted are statistically significant for all E_b/N_o values tested.

Figure 4.12 compares BER results for best-case and worst-case Gold codes (relative to random codes) for the *asynchronous* system. Again, the best-case Gold coded system outperforms the randomly coded system and the worst-case Gold coded system under performs the randomly coded system. The error bars indicate this performance improvement is significant for E_b/N_o values greater than 1.0 (dB). However, the performance differences due to Gold coding noted for the *asynchronous* system are less than observed for the *synchronous* system. These results illustrate that Gold code characteristics, i.e., the cross-correlation and autocorrelation values for a given family, are well suited for the *synchronous* case considered [Peterson, Ziemer, and Borth, 1995].
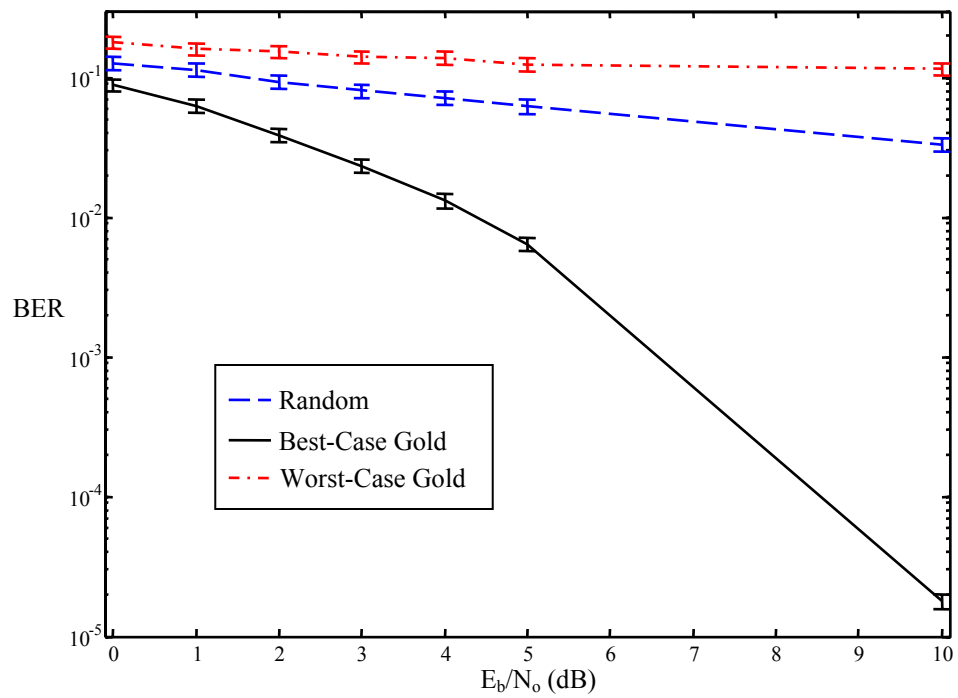
Figure 4.11. *Synchronous* Network: Gold vs. Randomly Coded System
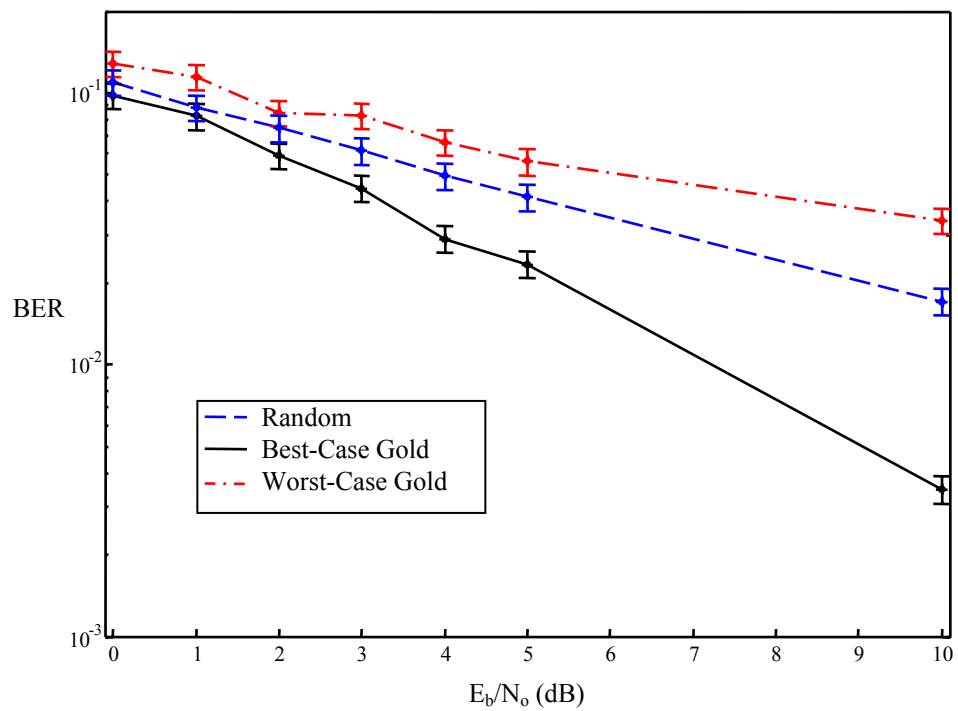
Figure 4.12. *Asynchronous* Network: Gold vs. Randomly Coded System

4.3   Pulse Shaping Results

The effects of chip waveform shaping on *synchronous* and *asynchronous* DS/SSMA system performance are characterized via simulation. All simulations contained $K = 9$ users with $N = 31$-length spreading codes to permit comparison of results obtained by Cho and Lehnert [Cho and Lehnert, 1999]. BER results are obtained using randomly coded sequences, best-case and worst-case Gold coded sequences and rectangular, Blackman, and Lanczos shaped chip waveforms.

4.3.1   Synchronous System Performance

As shown in Fig. 4.13, Fig. 4.14, and Fig. 4.15, simulation results for *synchronous* DS/SSMA systems show very little improvement in BER performance as the chip waveform shape changes; the resultant error bars overlap which indicates the differences shown are not statistically significant.
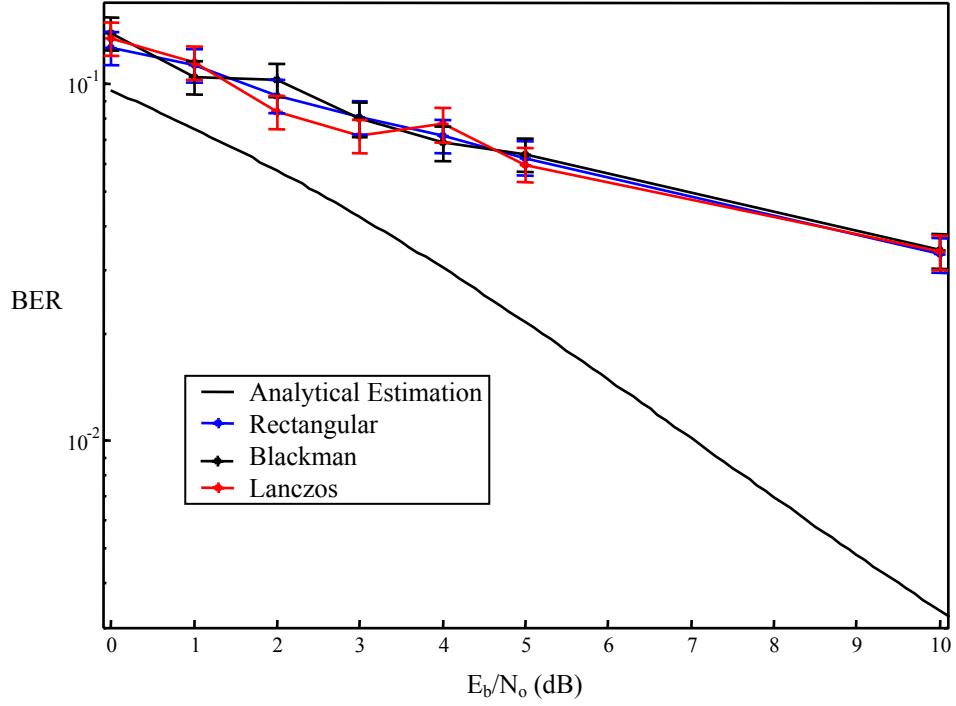
Figure 4.13. BER for *Synchronous* Randomly Coded DS/SSMA System with $K = 9$ Users, $N = 31$ Length Codes and Chip Waveform Shaping.
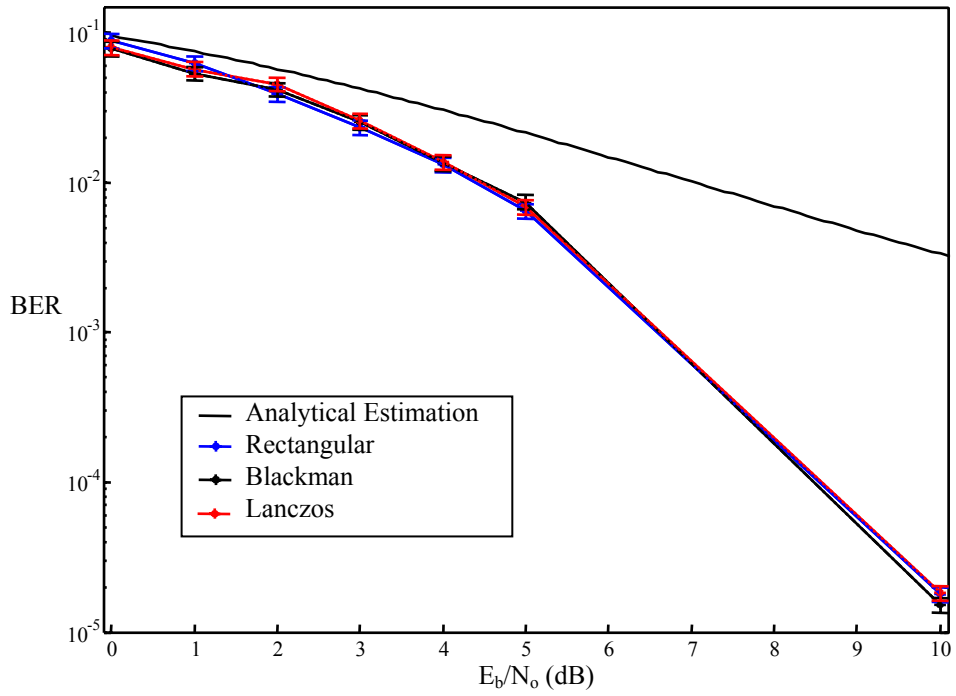
Figure 4.14. BER for *Synchronous* Best-Case Gold Coded DS/SSMA System with $K = 9$ Users, $N = 31$-Length Codes and Chip Waveform Shaping.
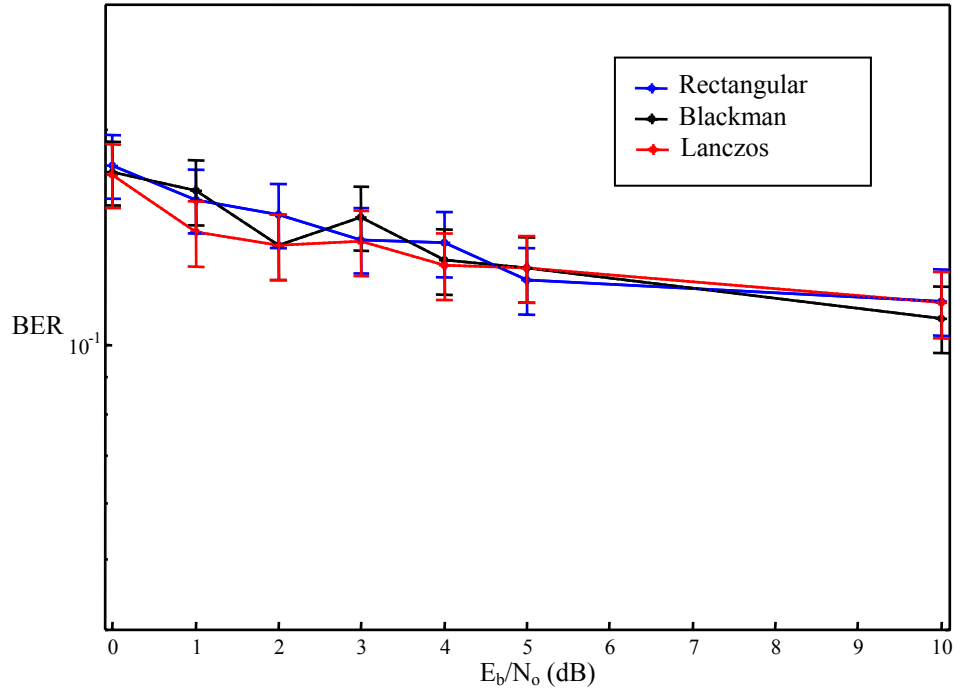
Figure 4.15. BER for *Synchronous* Worst-Case Gold Coded DS/SSMA System with
$K = 9$ Users, $N = 31$ Length Codes and Chip Waveform Shaping.

4.3.2   Asynchronous System Performance.

Results for the *asynchronous* system using randomly coded sequences are shown

in Fig. 4.16. Clearly, the BER resulting from the Lanczos waveform shape is improved

relative to the rectangular waveform shape; an improvement of approximately 6.0 dB is

indicated. Results presented in Fig. 4.17 for the best-case Gold coded *asynchronous*

system also indicate improvement when the Lanczos waveform shape is introduced; an

improvement of approximately 4.5 dB is indicated. As seen in Fig. 4.18, when the worst-

case Gold coded sequences are used in an *asynchronous* system, an improvement for

Lanczos waveform shapes of approximately 7.5 (dB) is realized.
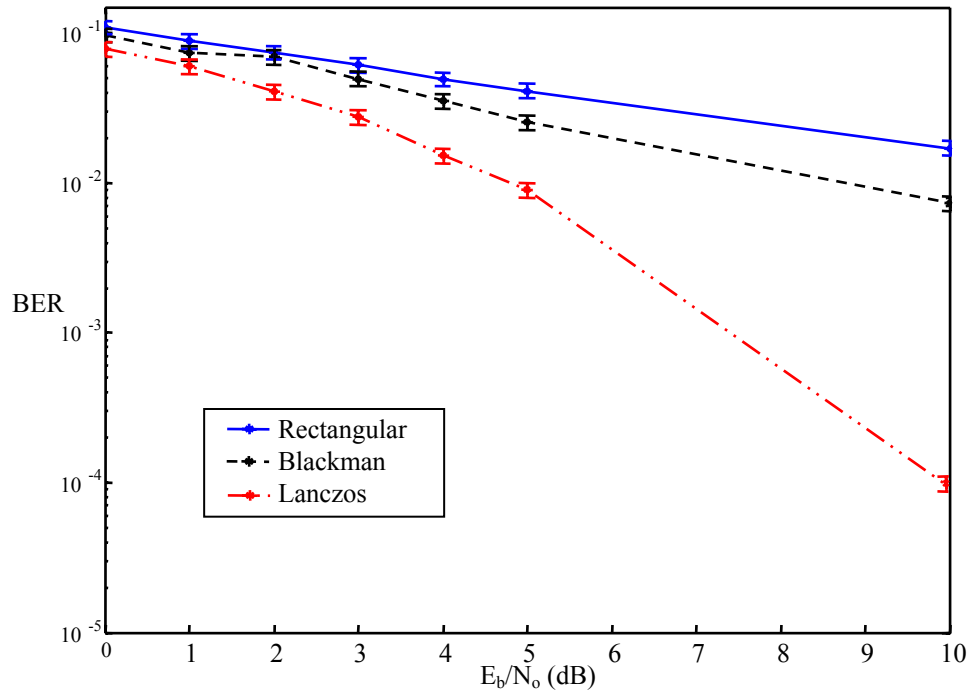
Figure 4.16. BER for *Asynchronous* Randomly Coded DS/SSMA System with $K = 9$ Users, $N = 31$ Length Codes and Chip Waveform Shaping.
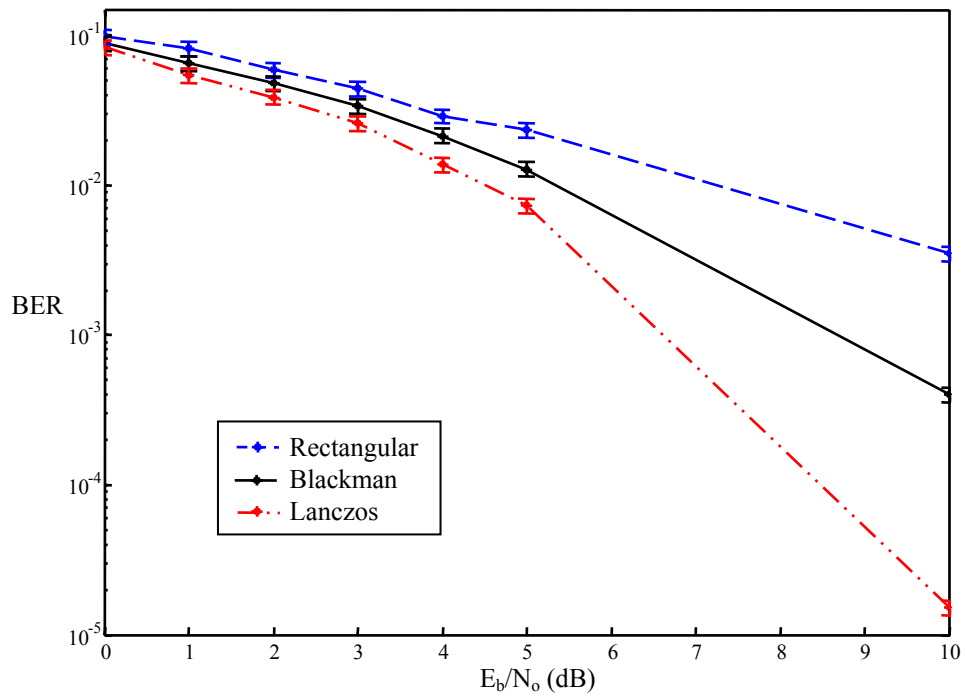
Figure 4.17. BER for *Asynchronous* Best-Case Gold Coded DS/SSMA System with $K = 9$ Users, $N = 31$ Length Codes and Chip Waveform Shaping.
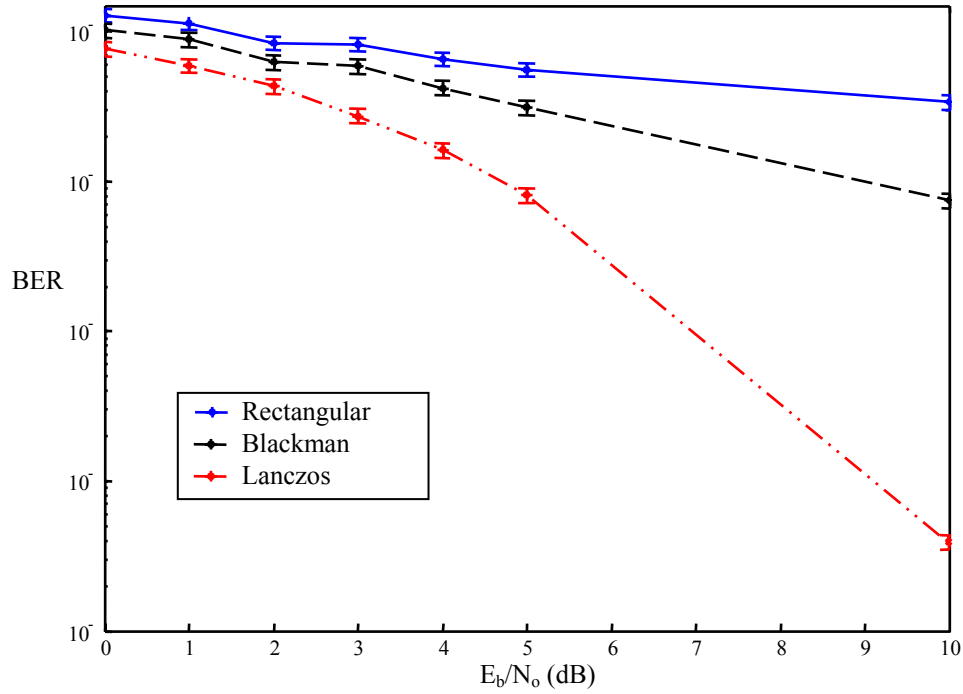
Figure 4.18. BER for *Asynchronous* Worst-Case Gold Coded DS/SSMA System with
$K = 9$ Users, $N = 31$ Length Codes and Chip Waveform Shaping.

4.4   *Cross-Correlation Analysis*

The particular fixed propagation delays associated with asynchronous results

presented in Section 4.3 are 201, 263, 273, 279, 309, 310, 171, and 339 samples for users

2 through 9, respectively. These delays represent Delay Profile #1. Cross-correlation

values between delayed interferers and the desired signal are calculated for each chip

waveform shape considered and are listed in Table 4.1 along with the sum of the

interferer cross-correlations. The sum of cross-correlation values for the best-case Gold

coded case progressively decreases when going from rectangular, to Blackman, to

Lanczos pulse shaping. Cumulative cross-correlation values obtained from Lanczos

pulse shaping is a factor of 6.25 lower than that of the rectangular pulse shape. This can

be mapped to a corresponding performance gain of 7.96 dB as obtained from:

$$Processing\ Gain = 10 \times log_{10}\left(\frac{Correlation\ of\ Interest}{Reference\ Correlation}\right). \tag{4.1}$$

This calculated processing gain is greater than that reflected in BER curves; Lanczos

waveform shaping provides an approximate 4.5 dB improvement over rectangular

waveform shaping. Systems using worst-case Gold coded sequences show a factor of 2.1

decrease in cumulative cross-correlation values between rectangular and Lanczos

waveform shapes, corresponding to an approximate 3.2 dB improvement; BER curves of

Fig. 4.18 indicate much greater improvement.

Table 4.1. Cross-Correlation Between Desired User and Interfering Users Employing
Best-Case Gold Codes and Chip Waveform Shaping for Delay Profile #1.

User #	2	3	4	5	6	7	8	9	
Delay	201	263	273	279	309	310	171	339	Sum
Cross-Correlation Value Multiplied by 341									
Rectangular	-11	-11	61	-11	77	5	29	5	144
Blackman	-4.83	-10.13	54.82	-2.12	70.91	-7.89	-1.78	-7.89	91.09
Lanczos	-4.69	-2.27	11.04	-7.20	15.86	3.38	3.54	3.38	23.05

Each multiple access interferer is modeled as having $N_s \times N_c$ (number of samples

per symbol times the number of chips per symbol) possible delay values; for parameters

used here there are 341 possible delay values and $(K - 1) = 8$ total multiple access

interferers in the system. Therefore, there are $341^8 = 1.8 \times 10^{20}$ possible combinations of

delay values for the system under consideration [Leon-Garcia, 1994]. To account for the

large number of possible delay profiles without conducting exhaustive Monte Carlo

simulations, the average cross-correlation value for each interfering code using

rectangular waveform shapes, and the cumulative of those averages, as listed in

Table 4.2, are considered.

Table 4.2 Average Cross-Correlation Between Desired User and Interfering Users
Employing Best-Case Gold Codes for All Possible Delays.

User #	2	3	4	5	6	7	8	9	Total
	-1.243	0.1775	0.1775	-1.243	-1.243	0.1775	-1.2453	0.1775	-4.2643

For simulation purposes, specific fixed propagation delay values for the

interferers were selected such that the cumulative cross-correlation values obtained in the

simulation closely matched the magnitude of the values listed in Table 4.2. The resulting

delays for users 2 through 9 are listed in Table 4.3 (Delay Profile #2) and produce a

cumulative cross-correlation value of 8/341 for rectangular chip waveforms. The BER

performance is again simulated using Delay Profile #2 with results shown in Fig. 4.19,

Fig. 4.20, and Fig. 4.21 for random, best-case Gold and worst-case Gold coded

sequences, respectively.

Table 4.3 Delay Profile #2 Based on Achieving Near Average Cross-Correlation
Between Desired User and Interfering Users Employing Best-Case Gold Codes and
Chip Waveform Shaping.

User #	2	3	4	5	6	7	8	9	
Delay	17	134	266	93	329	207	155	13	Sum
Cross-Correlation Value Multiplied by 341									
Rectangular	-3	5	5	-3	-3	5	-3	5	8
Blackman	6.11	-7.89	-7.89	6.11	-9.94	-7.89	-9.94	-7.89	-39.25
Lanczos	3.62	3.38	3.38	3.62	-1.97	3.38	-1.97	3.38	16.83
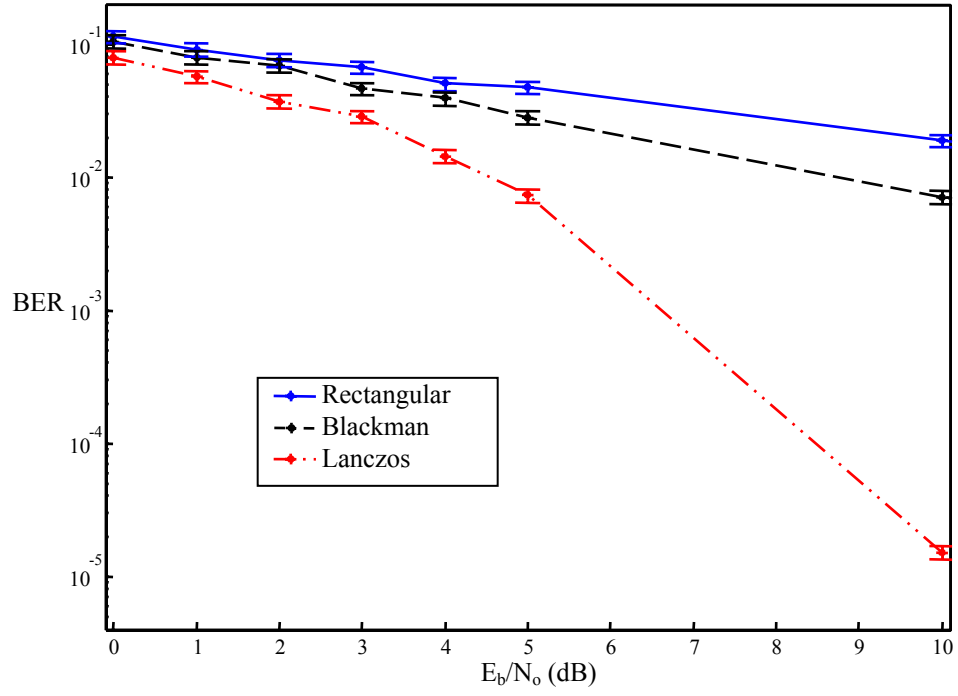
Figure 4.19. BER for *Randomly Coded* DS/SSMA System with Selected Delays:
$K = 9$ Users and $N = 31$ Length Codes.
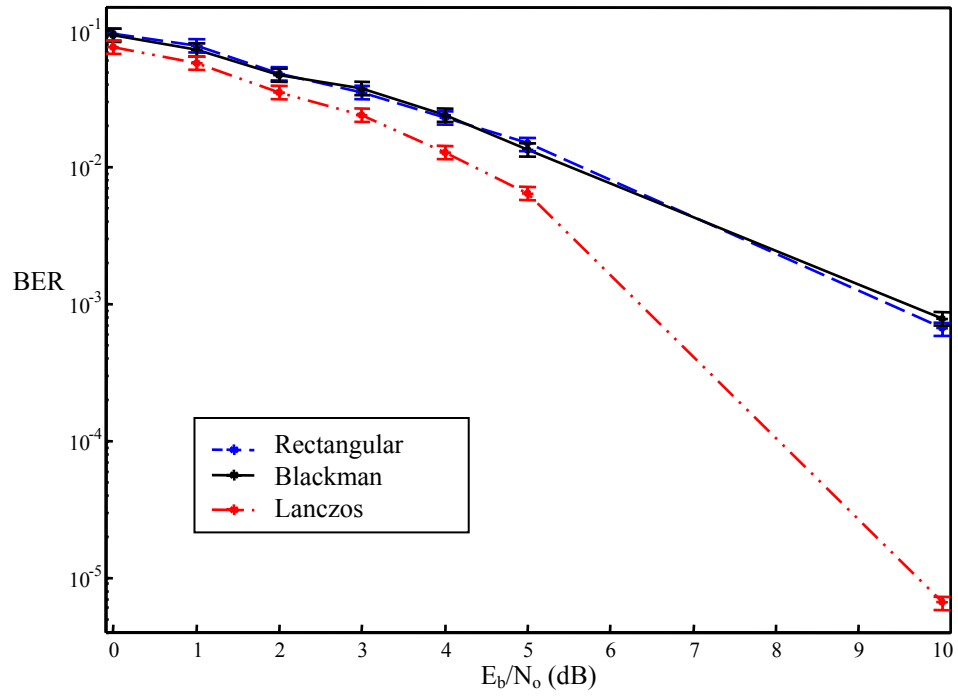
Figure 4.20. BER for *Best-Case Gold Coded* DS/SSMA System with Selected Delays:
$K = 9$ Users and $N = 31$ Length Codes.
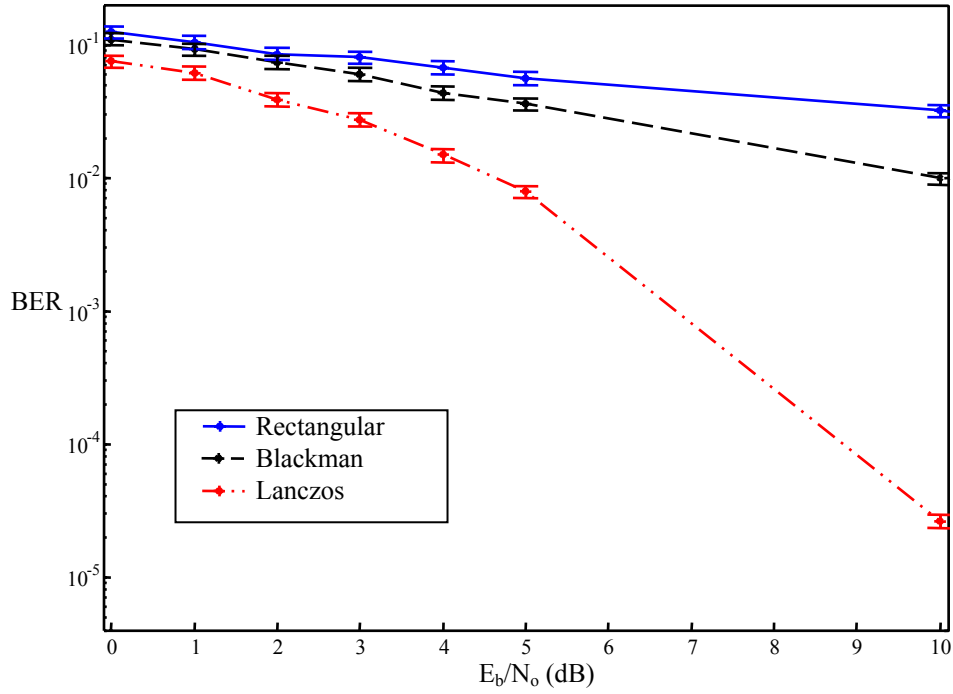
Figure 4.21. BER for *Worst-Case Gold Coded* DS/SSMA System with Selected
Delays: $K = 9$ Users and $N = 31$ Length Codes.

Using the fixed Delay Profile #2 described by values in Table 4.3, the BER of a

best-case Gold coded system shows BER improvement of 0.0 and 3.5 dB for Blackman

and Lanczos pulse shaping, respectively. These improvements are *not consistent* with the

processing gain of -6.91 and -3.22 dB obtained by using the cumulative average cross-

correlation values in Table 4.3, indicating an error in either the prediction or the

simulation.

To isolate the potential source of error, a more detailed analysis of the test statistic

involved in this process is conducted. The total test statistic z (the correlator output for

signal, MAI and noise present) is defined as:

$$z = z_{11} + \sum_{j=2}^{9} z_{1j} + z_n \qquad (4.2)$$

where z_{11} is the autocorrelation response between user one's transmitted (spread) symbol with user one's spreading code (the desired user), the z_{1j} terms in the summation are the cross-correlation response between user one's spreading code and the jth user's transmitted symbol, and z_n is the noise component out of the correlator. Table 4.4 lists the mean and variance of individual test statistics for an E_b/N_o value of 3.0 dB using Delay Profile #2, where z_{1i} is the summation term from (4.2) representing the summation of cross-correlations of each interfering user with user one. The z_{1i} term is the determining factor in predicting MAI effects. The variances of the test statistic components indicate the amount of power associated with each term.

Table 4.4. Test Statistic Mean and Variance Values

	z_{11}		z_{1i}		z_n	
	μ	σ^2 (x10^5)	μ	σ^2 (x10^3)	μ	σ^2 (x10^4)
Rectangular	6.019	1.163	-0.177	6.325	0.005	2.880
Blackman	0.597	1.163	-0.300	6.227	1.651	2.890
Lanczos	0.424	1.163	-0.036	0.292	-0.098	2.898

For the case where Blackman shaped chip waveforms are used, the variance in the MAI term (z_{1i}) is not significantly different from the rectangular shaped chip waveforms. This trend is consistent with BER data plotted in Fig. 4.20 which shows little to no variation between Blackman and rectangular pulse shaping. As shown in Table 4.4, the variance in the MAI term for the Lanczos shaped chip waveforms is approximately one order-of-magnitude lower than both the rectangular and Blackman case, indicating significantly better BER performance should be realized for Lanczos shaped waveforms relative to either rectangular or Blackman shaped chip waveforms. This trend is also consistent with the BER results in Fig. 4.20.

Due to the temporal overlap of the desired user's symbol interval with two portions of each interfering users' symbol intervals, there exist four possible data value combinations for each *asynchronous* interfering user in the despreading process; the interferer can have a value of either one or zero for both symbols, a one then a zero, or a zero then a one. These data combinations effect the cross-correlation between the desired user's spreading code and the delayed interfering user's spreading code. Table 4.5 and Table 4.6 list the mean and variance of the cross-correlations for users 2 through 9 when these data effects are included. The summation of the variances of the cross-correlations with data effects results in 6.330×10^3, 6.195×10^3, and 0.294×10^3 for rectangular, Blackman, and Lanczos chip waveforms, respectively. These values indicate that for the specific delays listed in Table 4.3 the systems with rectangular and Blackman chip waveforms should perform similarly, while the system with Lanczos chip waveforms should significantly outperform the other two. The simulation results in Fig. 4.20 reflect this prediction.

Table 4.5. Mean and Variance of Cross-Correlation with Data Effects for
Users 2 through 5 using Delay Profile #2

User #	2		3		4		5	
Delay	17		134		266		93	
	μ	$\sigma^2 \times 10^3$	μ	$\sigma^2 \times 10^3$	μ	$\sigma^2 \times 10^3$	μ	$\sigma^2 \times 10^3$
Rectangular	0.019	0.029	0.053	0.192	0.074	3.815	-0.026	0.009
Blackman	0.222	0.035	-0.050	0.800	0.063	2.502	-0.042	0.022
Lanczos	-0.024	0.019	0.042	0.006	-0.008	0.157	0.009	0.009

Table 4.6. Mean and Variance of Cross-Correlation with Data Effects for
Users 6 through 9 using Delay Profile #2

User #	6		7		8		9	
Delay	329		207		155		13	
	μ	σ^2 x10^3	μ	σ^2 x10^3	μ	σ^2 x10^3	μ	σ^2 x10^3
Rectangular	0.011	0.362	0.029	1.638	-0.013	0.186	-0.026	0.098
Blackman	0.016	0.515	-0.034	1.505	0.177	0.505	0.026	0.310
Lanczos	-0.003	0.024	0.025	0.052	0.006	0.022	0.074	0.006

The cross-correlation statistics for the system design resulting in Fig. 4.17 BER curves are listed in Table 4.7. and Table 4.8. The summation of cross-correlation variances including data effects results in 11.185 x 10^3, 5.537 x 10^3 and 0.953 x 10^3 for rectangular, Blackman, and Lanczos chip shapes, respectively. These relative variance values are consistent with the significant improvement displayed in Fig. 4.17. for Blackman and Lanczos chip waveforms over rectangular chip waveforms for Delay Profile #1.

Table 4.7. Mean and Variance of Cross Correlation with Data Effects
for Users 2 through 5 using Delay Profile #1

User #	2		3		4		5	
Delay	201		263		273		279	
	μ	σ^2 x10^3	μ	σ^2 x10^3	μ	σ^2 x10^3	μ	σ^2 x10^3
Rectangular	0.162	0.071	-0.746	0.441	-7.018	2.890	0.027	0.536
Blackman	0.306	0.025	0.214	0.535	-0.495	1.742	0.045	0.082
Lanczos	0.000	0.012	-0.049	0.024	0.238	0.128	-0.067	0.525

Table 4.8. Mean and Variance of Cross Correlation with Data Effects
for Users 2 through 5 using Delay Profile #1

User #	6		7		8		9	
Delay	309		310		171		339	
	μ	σ^2 x10^3	μ	σ^2 x10^3	μ	σ^2 x10^3	μ	σ^2 x10^3
Rectangular	-3.673	2.985	0.198	0.096	-0.251	4.111	0.440	0.055
Blackman	-1.495	2.791	-0.142	0.296	-0.081	0.003	-0.309	0.062
Lanczos	0.371	0.128	0.026	0.006	-0.105	0.112	0.165	0.017

4.5   Code Length Analysis

The BER performance of a system using $N = 31$-length spreading codes is compared to the BER performance of a system using $N = 511$-length spreading codes as shown in Fig. 4.22. In this case, data is presented for *synchronous* systems with $K = 9$ users, random and best-case Gold coded sequences, and rectangular waveform shapes. All cases use a fixed symbol duration and one full period of the spreading code per symbol duration. Data in Fig. 4.22 reflects an improvement of approximately 7.5 dB and 0.5 dB for random and Gold codes, respectively, as code length increases from 31 to 511. For the random case, although appreciable, this improvement is less than the theoretical processing gain improvement of 12.7 dB expected by increasing N from 31 to 511. The Gold coded case shows some improvement, but falls far short of the expected 12.7 dB gain, and is discussed below.

BER results for 511-length randomly and Gold coded results for $K = 9$ are compared to the single user communications performance calculated from (2.15) as shown in Fig. 4.23. Implementing 511-length spreading codes increases the 9 user system performance (decreases BER) to a level that is almost equivalent to the single user case for randomly coded waveforms. For Gold coded systems, BER improves to a level that is not statistically different than the single user communications performance (no MAI present). Due to the presence of AWGN, system BER performance is limited (cannot decrease below) the single-user communications performance. Marginal performance gain is observed when increasing the Gold code length from 31 to 511 given performance cannot be any better than the case where there is no MAI present.
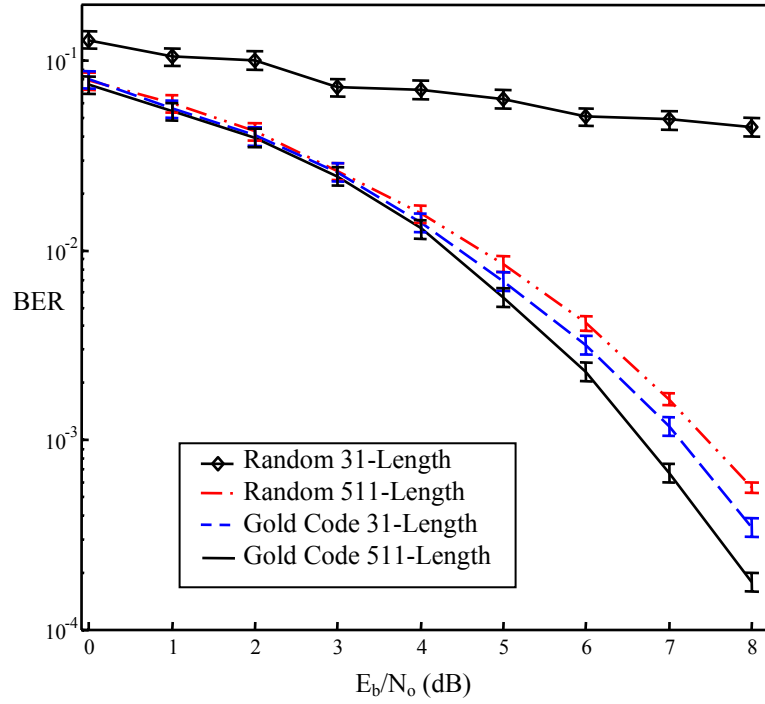
Figure 4.22. BER for Randomly and Gold Coded *Synchronous* DS/SSMA Systems
with Fixed Symbol Duration: $K = 9$ Users and $N = 31$ and 511 Length Codes using
One Full Period of the Spreading Code per Symbol Duration.
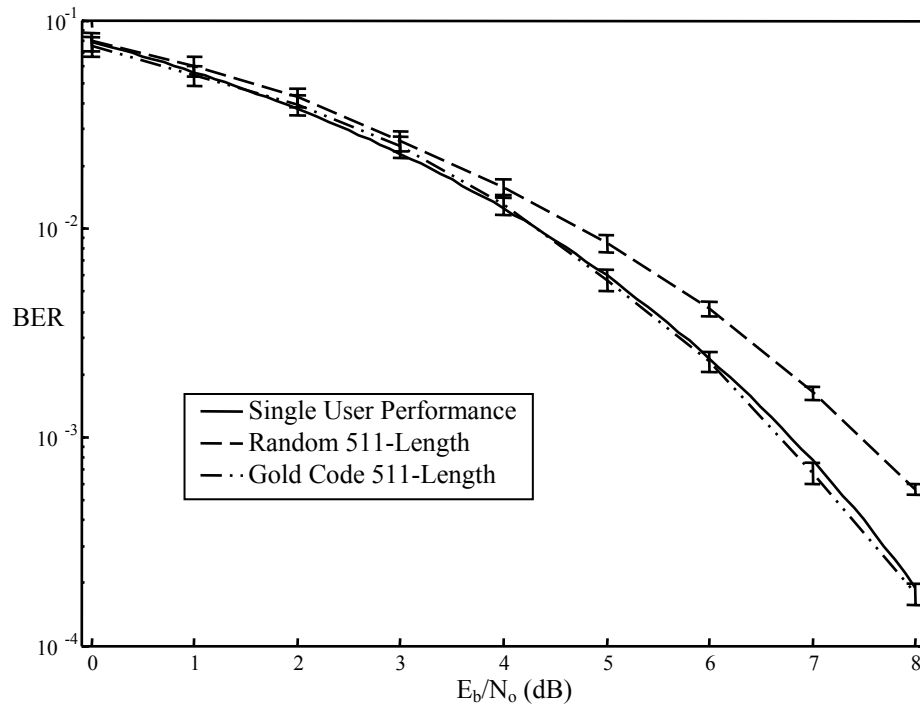
Figure 4.23. Comparison of Single User Performance to Randomly and Gold Coded
Synchronous DS/SSMA Systems: $K = 9$ Users and $N = 511$ Length Codes.

V. Conclusions

5.1   Research Contributions

This research provides characterization of multiple access interference (MAI) for direct-sequence spread-spectrum multiple access (DS/SSMA) systems through variation of several factors, including the use of random versus Gold spreading codes, chip waveform shape selection, and variation in spreading code length. Results provide insight into the applicability of previously published approximations for bit error rate (BER) and the impact that each system factor has on overall performance. This research also provides a modeling and simulation tool capable of supporting future research into the factors effecting DS/SSMA system performance. This simulated model enables a more realistic implementation of potential system configurations to verify future analytical approximations.

5.2   Summary of Findings

5.2.1   Random versus Gold Spreading Codes

For DS/SSMA systems employing Gold spreading codes, performance is shown to be greater than similar systems employing random spreading codes. This improvement is most noticeable at E_b/N_o values greater than 1.0 dB, the point at which the effect of MAI on BER is large enough to dominate the effect of additive white Gaussian noise (AWGN) on BER. Additionally, the improvement due to Gold coding is more pronounced in *synchronous* systems, indicating that the power of Gold coding is based on the *synchronous* characteristics of the codes.

5.2.2 Chip Waveform Shaping

Improvement in BER due to chip waveform shaping is shown to be mostly limited to *asynchronous* systems; changes in spreading code cross-correlation statistics (relative to the synchronous case) occur due to symbol boundary misalignment in the *asynchronous* case. For the case of fixed, predetermined propagation delay profiles for the synchronous users, the BER improves (or degrades) by an amount that is in part dictated by the improvement (or degradation) in variance of symbol cross-correlation which occurs when introducing an alternate chip waveform shape; improved cross-correlation variance here implies lowers MAI levels and corresponding lower BERs as the chip waveform shape is changed. The demonstrated improvement in this work is an effective processing gain of between 3.5 and 4.5 dB for a system using Lanczos chip waveforms, relative to identical systems using rectangular chip waveforms in an *asynchronous* network with best-case Gold codes. This improvement was found to be dependent on the variance of the cross-correlation between spreading codes when considering data effects.

5.2.3 Spreading Code Length

An increase in spreading code length from 31 chips to 511 chips provides significant improvement in BER for a *synchronous* DS/SSMA system employing random spreading codes. For a given BER, a reduction in required E_b/N_o of approximately 7.5 dB is demonstrated, falling below the expected performance improvement (reduction) of $10 \times \log_{10}(L = 511/31) \approx 12.7$ dB. In cases where total interfering power is dominated by MAI and not by AWGN, increasing code length by a factor of L effectively reduces the dominant MAI power by an amount proportional to $1/L$ and decreases (improves) overall

BER. For Gold coded systems the performance improvement was only 0.5 dB for the

same code length increase. This improvement was shown to decrease BER to the level of

single user performance, which is the theoretical best performance in the presence of

AWGN.

5.3   Recommendations for Future Research

5.3.1   M-Ary Data and/or Spreading Modulation

This research only simulated binary modulation and spreading techniques, i.e.,

binary phase shift keyed (BPSK) data and spreading modulations. DS/SSMA systems

can be implemented using other M-ary modulations, e.g., BPSK data modulation can be

combined with quadrature phase shift keyed (QPSK) spreading modulation as done in the

cellular IS-95 communication system. The impact on BER performance using alternate

modulation techniques could be easily simulated with only minor modification of

simulation code generated under this research.

5.3.2   Full Monte Carlo Simulations

Given the randomness associated with several factors in a DS/SSMA system, i.e.,

random multiple access coding, asynchronous user propagation delays, etc., several

variables were fixed for simulations in this work such that they provided "average"

operating characteristics for the simulation. A full Monte Carlo simulation of the system

variables could be used to more fully characterize BER performance in the presence of

MAI.

5.3.3 Near-Far Comparisons

Scenarios could be considered where interfering users are transmitting at locations either "nearer" to or "farther" from the receiver of interest. Simulation of these scenarios would address the so called "near-far" problem which was not addressed as part of this research (in this work all interfering signals were assumed to be received at the same power level as the desired user). In near-far cases, the MAI power received varies from user-to-user relative to the desired user and this variation influences the impact of other factors on BER.

5.3.4 Characterization Using Delay Profile Variation

For this research, variances in cross-correlation (which directly correlate to MAI power levels) were used to induce the interaction of various spreading codes on expected performance improvement. For the asynchronous network case, these MAI power levels were obtained by using a specific propagation delay profile (collection of pre-assigned delay values) for the simulations. As revealed throughout the preliminary stages of this research, there exist specific propagation delay profiles whereby a change in chip waveform shape on interfering users did not improve BER performance relative to the rectangular chip waveforms. While average performance improvement provides valuable academic insight, accurately characterizing performance for delay profile variation (which can occur with changes in system topology) may be important, especially for actual communication systems.

Matlab® simulation code is presented in this appendix. The primary function used for 31-

length codes was ber_shape_async_all31_2.m and for 511-length codes was

ber_shape_sync_BR511.m.

A.1 ber_shape_async_all31_2.m

```
function [berb, berr, berw] = ber_shape_async_all31_norm2(eb,numbsig,samps,delay,shape)
%
% [berb berr berw] = ber_shape_async_rand31_2(eb,numbsig,samps,delay,shape)
%
%   Change the random spreading codes to be aperiodic.
%
% Experimentally determine the BER for
% numbsig users at each of the eb values
% of Eb/No (dB).  All signals are asynchronous.
%
% Inputs:   eb = Eb/No (dB)
%         numbsig = number of signals (including desired user)
%         samps = samples per chip interval
%         srand = random codes
%         delay = number of smaples time delay for each user (from 0 -
%             total samples per symbol interval)
%         shape = desired shape for the chip interval
%             1 = rectangular
%             2 = half sine
%             3 = raised cosine
%             4 = blackman
%             5 = raised cosine + 1
%
% Returns
%   berb = BER using best case Gold codes
%   berw = BER using worst case Gold codes
%   berr = BER using random codes defined by srand
%
%
% Written by Matthew Glen 5 Dec 03

%%%% Spread Spectrum MAI characterization project
%%%%% Create m-seq needed for Gold codes

bits = 1;
slen = 31;
nsamps = 100;
t = 0:samps-1;

if shape == 1
   wf = ones(1,samps);
elseif shape == 2
   wf = sin(pi*t/(length(t)-1));
```

```
elseif shape == 3
    wf = sqrt(2/3)*(1 - cos(2*pi*t/(length(t)-1)));

elseif shape == 4
    wf = 0.42 - 0.5*cos(2*pi*t/(length(t)-1));

elseif shape == 5
    t2 = t+10;
    wf = (sin(2*pi*t2)./(2*pi*t2)).^2;
end

% gpoly1 = [1 0 0 0 0 1 0 0 0 1];    %%% 1021
% gpoly2 = [1 1 0 0 1 1 0 0 0 1];    %%% 1461
% init =   [1 1 1 1 1 1 1 1 1];      %%% initial register state
% mseq = [zeros(2,511)];       %%% create mseq vector of zeros
% mseq(1,:) = mSeqGen(init,gpoly1);  %%% generate the first mseq
% mseq(2,:) = mSeqGen(init,gpoly2);  %%% generate the 2nd mseq

gpoly1 = [1 0 0 1 0 1];    %%% 45
gpoly2 = [1 1 1 1 0 1];    %%% 75
init =   [1 1 1 1 1];         %%% initial register state
mseq = [zeros(2,31)];       %%% create mseq vector of zeros
mseq(1,:) = mSeqGen(init,gpoly1);   %%% generate the first mseq
mseq(2,:) = mSeqGen(init,gpoly2);   %%% generate the 2nd mseq

gold = gold_gen(mseq(1,:), mseq(2,:)).*(-2) +1;  %% see which direction they are lined up
[row col] = size(gold);

%%% Run test until endbit number of errors is reached
endbit = 300;

% Define the best and worst combinations of 511-length gold codes
bestorder = [1 3:33]; % best codes
worstorder = [2 33 30 5 7 10 17 31 29 27 26 24 22 21 13 4 1];  % worst codes

%%% Spreading codes
best(1:numbsig,:) = gold(bestorder(1:numbsig),:);
worst(1:numbsig,:) = gold(worstorder(1:numbsig),:);
% Sample the spreading codes
cbest = kron(best,wf);
cworst = kron(worst,wf);

% Calculate the signal power
% The rbw in the signal has unit power multiple effect
% thus the power is just the power of the spreading code
% All three spreading codes have the same power
sigpower = cbest(1,:)*cbest(1,:)'/(samps*slen);

%%%%%% Normalize the energy in the signals
cbest = cbest./sqrt(sigpower);
cworst = cworst./sqrt(sigpower);

sigpower_n = cbest(1,:)*cbest(1,:)'/(samps*slen);
%%%% Calc noise power required for Eb/No
%  noise coeef = 0.5*samps*slen*(10^(eb/10))^(-1)
```

```
npower = 0.5*sigpower_n*samps*slen*(10.^(eb/10)).^(-1);

for ii = 1:length(eb)
    errorsb = 0;              %%%% Zero out number of errors
    errorsw = 0;
    errorsr = 0;
    totaltested = 0;          %%% Zero out number of total bits tested
    totnoise = randn(1,slen*samps*nsamps);  %%% Gen noise nsamps number of bits long
    nbit = 1;                             %%% Counts which noise bit to use
    srand = sign(randn(1,slen*samps*nsamps));
    trand= kron(srand,wf)./sqrt(sigpower);
    coden = 1;
    %%% generate the first bit for each user
    data1 = sign(rand(numbsig,bits)-0.5);
    rbw1(:,1:slen*samps) = data1(:,1)*ones(1,slen*samps);

    for user = 1:numbsig
        crand1(user,:) = trand((coden-1)*slen*samps+1:coden*slen*samps);
        coden = coden + 1;
    end
    while errorsr < endbit

        %%%% Generate new random spreading codes for each bit
        for user = 1:numbsig
            crand2(user,:) = trand((coden-1)*slen*samps+1:coden*slen*samps);
            coden = coden + 1;
            if coden > nsamps
                srand = sign(randn(1,slen*samps*nsamps));
                trand= kron(srand,wf)./sqrt(sigpower);
                coden = 1;
            end
        end

        crand = [crand1 crand2];

        %%% Generate the second bit for each user
        data2 = sign(rand(numbsig,bits)-0.5);
        rbw2(:,1:slen*samps) = data2(:,1)*ones(1,slen*samps);

        data = [data1 data2];
        rbw = [rbw1 rbw2];

        %% Number of bits in the signal
        %       numbit = length(data(1))/slen;

        %%%% Noise
        %% Create AGWN with power of value in sqrt (var = value in sqrt)
        %% eb/no ~ SNR*slen/2
        %
        % Eb/No = 0 dB noise has 15.5
        %       = 1 dB noise has 12.312
        %       = 2 dB noise has 9.775
        %       = 4 dB noise has 6.169
        %       = 6 dB noise has 3.8941
        %       = 8 dB noise has 2.4564
        %       = 10dB noise has 1.5499
```

```
    if nbit > nsamps
        totnoise = randn(1,slen*samps*nsamps);  %%% Gen noise nsamps number of bits long
        nbit = 1;                    %%% Counts which noise bit to use
    end
    noise = sqrt(npower(ii))*totnoise(slen*samps*(nbit-1)+1:slen*samps*nbit);
    nbit = nbit + 1;

    clear rbest rworst rrand;
    for j = 1:numbsig
        rbest(j,:) = rbw(j,delay(j)+1:delay(j)+slen*samps).*cbest(1,:).*[cbest(j,delay(j)+1:slen*samps)
cbest(j,1:delay(j))];    %%% spread and despread symbols
        rworst(j,:) = rbw(j,delay(j)+1:delay(j)+slen*samps).*cworst(1,:).*[cworst(j,delay(j)+1:slen*samps)
cworst(j,1:delay(j))];   %%% spread and despread symbols
        rrand(j,:) =
rbw(j,delay(j)+1:delay(j)+slen*samps).*crand(1,1:341).*crand(j,delay(j)+1:delay(j)+slen*samps);    %%%
spread and despread symbols

    end
    rbest(numbsig+1,:) = noise.*cbest(1,:);
    rbest(numbsig+2,:) = sum(rbest);

    rworst(numbsig+1,:) = noise.*cworst(1,:);
    rworst(numbsig+2,:) = sum(rworst);

    rrand(numbsig+1,:) = noise.*crand(1,1:341);
    rrand(numbsig+2,:) = sum(rrand);

    rxbitb = sign(sum(rbest(numbsig+2,1:slen*samps),2));

    rxbitw = sign(sum(rworst(numbsig+2,1:slen*samps),2));

    rxbitr = sign(sum(rrand(numbsig+2,1:slen*samps),2));

    totaltested = totaltested + bits;

    errorsb = errorsb+abs((data(1,1)-rxbitb)/2);
    errorsw = errorsw+abs((data(1,1)-rxbitw)/2);
    errorsr = errorsr+abs((data(1,1)-rxbitr)/2);

    if errorsr >= endbit
        break
    end
    data1 = data2;
    rbw1 = rbw2;
    crand1 = crand2;

    end
    berb(ii) = errorsb/totaltested;
    berw(ii) = errorsw/totaltested;
    berr(ii) = errorsr/totaltested;

end
```

A.2   *ber_shape_sync_BR511.m*

```
function [berb,berr] = ber_shape_sync_BR511(eb,numbsig,samps,shape)
%
% [berb berr] = ber_shape_sync_BR511(eb,numbsig,samps,shape)
%
% Experimentally determine the BER for
% numbsig users at each of the eb values
% of Eb/No (dB).  All signals are synchronous.
%
% Inputs:   eb = Eb/No (dB)
%        numbsig = number of signals (including desired user)
%        samps = samples per chip interval
%        srand = random codes
%        shape = desired shape for the chip interval
%              1 = rectangular
%              2 = half sine
%              3 = raised cosine
%              4 = Blackman
%              5 = Lanczos
%
% Returns
%   berb = BER using best case Gold codes
%   berr = BER using random codes defined by srand
%
%
% Written by Matthew Glen 5 Dec 03


%%%% Thesis Code

bits = 1;
slen = 511;
nsamps = 50;
t = 0:samps-1;

if shape == 1
   wf = ones(1,samps);
elseif shape == 2
   wf = sin(pi*t/(length(t)-1));

elseif shape == 3
   wf = sqrt(2/3)*(1 - cos(2*pi*t/(length(t)-1)));

elseif shape == 4
   wf = 0.42 - 0.5*cos(2*pi*t/(length(t)-1));

elseif shape == 5
   t2 = t+10;
   wf = (sin(2*pi*t2)./(2*pi*t2)).^2;
end

%%%%% Create m-seq needed for Gold codes
gpoly1 = [1 0 0 0 0 1 0 0 0 1];    %%% 1021 octal polynomial
gpoly2 = [1 1 0 0 1 1 0 0 0 1];    %%% 1461 octal polynomial
```

```
init =   [1 1 1 1 1 1 1 1 1];        %%% initial register state
mseq = [zeros(2,511)];        %%% create mseq vector of zeros
mseq(1,:) = mSeqGen(init,gpoly1);   %%% generate the first mseq
mseq(2,:) = mSeqGen(init,gpoly2);   %%% generate the 2nd mseq


gold = zeros(slen+2,slen);
gold = gold_gen(mseq(1,:), mseq(2,:)).*(-2) +1;  %% Generate the Gold Code Family
[row col] = size(gold);




%% Run test until endbit number of errors is reached
endbit = 300;

% Define the best and worst combinations of 511-length gold codes
bestorder = [1 3:511]; % best codes
% worstorder = [2 8 11 12 20 23 25 27 29 31 32 34 36 37 41 42 43 45 46 47 48 49 51
%          52 57 60 61 62 68 72 79 80 81 84 94 96 98 100 101];  % worst codes

%%% Spreading codes
best(1:numbsig,:) = gold(bestorder(1:numbsig),:);


% Sample the spreading codes
cbest = kron(best,wf);

% Make 2 bits worth of the spreading code
% cbest = [cbest1 cbest1];
% cworst = [cworst1 cworst1];
% crand = [crand1 crand1];


% Calculate the signal power
% The rbw in the signal has unit power multiple effect
% thus the power is just the power of the spreading code
% All three spreading codes have the same power
sigpower = cbest(1,:)*cbest(1,:)'/(samps*slen);

%%%%%% Normalize the energy in the signals
cbest = cbest./sqrt(sigpower);

%% Calculate new Normalized signal power
%% This value remains the same as sigpower for rectangular shapes
sigpower_n = cbest(1,:)*cbest(1,:)'/(samps*slen);

%%%% Calc noise power required for Eb/No
%  noise coeef = 0.5*samps*slen*(10^(eb/10))^(-1)
npower = 0.5*sigpower_n*samps*slen*(10.^(eb/10)).^(-1);

%% Creates a vector of zeros to be filled with the random spreading codes
crand = zeros(9,slen*samps);


for ii = 1:length(eb)
    errorsb = 0;                %%% Zero out number of errors
```

A-6

```
errorsr = 0;
totaltested = 0;          %%% Zero out number of total bits tested
%%% Generate nsamps number of bits worth of noise
totnoise = randn(1,slen*samps*nsamps);


nbit = 1;                         %%% Counts which noise bit to use


%%% Generate a string of random 1's and -1's to be used as the random
%%% spreading codes
srand = sign(randn(1,slen*samps*nsamps));


%% Track the number of bits worth of random spreading code that has
%% been used
coden = 1;


while errorsb < endbit  %% Run until 'endbit' number of errors

   %%%% Generate new random spreading codes for each bit

   %%  Create the random spreading codes for each transmitting user
   for user = 1:numbsig
      crand(user,:) = kron(srand((coden-1)*slen+1:coden*slen),wf)./sqrt(sigpower);
      coden = coden + 1;
      %% Generate new random 1's and -1's when the previous srand is
      %% all used up
      if coden > nsamps
         srand = sign(randn(1,slen*samps*nsamps));
         coden = 1;
      end
   end


   %% Generate the current data bit values for each user
   data = sign(rand(numbsig,bits)-0.5);


   %% Sample the current data bit values for each user
   rbw(:,1:slen*samps) = data(:,1)*ones(1,slen*samps);


   %%% Noise
   %% Create AGWN with power of value in sqrt (var = value in sqrt)
   %% eb/no ~ SNR*slen/2
   %
   % Eb/No = 0 dB noise has 15.5 variance
   %       = 1 dB noise has 12.312 variance
   %       = 2 dB noise has 9.775 variance
   %       = 4 dB noise has 6.169 variance
   %       = 6 dB noise has 3.8941 variance
   %       = 8 dB noise has 2.4564 variance
   %       = 10dB noise has 1.5499 variance
   %% Group a symbol's worth of noise samples to be added to received
   %% signal from nsamps long noise string 'totnoise'.  Generate new
   %% long noise string if nsamps number of bits have already been
   %% tested.
   if nbit > nsamps
      totnoise = randn(1,slen*samps*nsamps);  %%% Gen noise nsamps number of bits long
      nbit = 1;                  %%% Counts which noise bit to use
   end
```

```matlab
        noise = sqrt(npower(ii))*totnoise(slen*samps*(nbit-1)+1:slen*samps*nbit);
        nbit = nbit + 1;

        clear rbest rrand;
        for j = 1:numbsig
            rbest(j,:) = rbw(j,:).*cbest(1,:).*cbest(j,:);    %%% spread and despread symbols
            rrand(j,:) = rbw(j,:).*crand(1,:).*crand(j,:);    %%% spread and despread symbols
        end

        %%% Current received signal for Best-Case Gold Coded signals
        rbest(numbsig+1,:) = noise.*cbest(1,:);
        rbest(numbsig+2,:) = sum(rbest);

        %%% Current received signal for randomly Coded signals
        rrand(numbsig+1,:) = noise.*crand(1,:);
        rrand(numbsig+2,:) = sum(rrand);

        %%% Estimate the current "best-case" received bit
        rxbitb = makeint(sum(rbest(numbsig+2,1:slen*samps),2));

        %%% Estimate the current "random" received bit
        rxbitr = makeint(sum(rrand(numbsig+2,1:slen*samps),2));

        totaltested = totaltested + bits;

        %%% Determine if estimated bits are in error and increment total
        %%% number of errors
        errorsb = errorsb+abs((data(1,1)-rxbitb)/2);
        errorsr = errorsr+abs((data(1,1)-rxbitr)/2);

    end

    berb(ii) = errorsb/totaltested;
    berr(ii) = errorsr/totaltested;

end
```

Bibliography

Canadeo, Courtney, M. *Ultra Wide Band Multiple Access Performance Using TH-PPM and DS-BPSK Modulations*. MS thesis, AFIT/GE/ENG/03-06. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 2003.

Cho, Joon H. and James S. Lehnert. "Optimum Chip Waveforms for DS/SSMA Communications with Random Spreading Sequences and a Matched Filter Receiver," *Wireless communications and Networking Conference, 1999.* 574-578. (Septemebr 1999).

Geraniotis, Evaggelos and Behzad Ghaffari. "Performance of Binary and Quaternary Direct-Sequence Spread-Spectrum Multiple-Access Systems with Random Signature Sequences," *IEEE Transactions on Communications*, 39(5): 713-724 (May 1991).

Jain, Raj. *The Art of Computer Systems Performance Analysis*. New York: John Wiley & Sons, Inc., 1991.

Kok, Boon K. and M. A. Do. "Effects of Spreading Chip Waveform Pulse-Shaping on the Performance of DS-CDMA Indoor Radio Personal Communication Systems in a Frequency-Selective Rician Fading Channel," *IEEE International Symposium on Consumer Electronics*. 198-201. (December 1997).

Lehnert, James S. *Final Report on Phase One Activities*. Tactical Targeting Network Technologies Subcontract No. 4400-00-3308. September 2002.

Leon-Garcia, Alberto. *Probability and Random Processes for Electrical Engineering*. Reading, Massachusetts: Addison-Wesley, 1994.

Parl, S. "A New Method of Calculating the Generalized Q-function," *IEEE Transactions on Information Theory*, 26(1): 121-124 (January 1980).

Peterson, Roger L., Rodger E. Ziemer and David E. Borth. *Introduction to Spread Spectrum Communications*. New Jersey: Prentice Hall, 1995.

Pursley, Michael B. "Performance Evaluation for Phase-Coded Spread Spectrum Multiple Access Communication-Part I: System Analysis," *IEEE Transactions on Communications*, 25(8): 795-799 (August 1977).

Yao, Kung. "Error Probability of Asynchronous Spread Spectrum Multiple Access Communication Systems," *IEEE Transactions on Communications*, 25(8): 803-809 (August 1977).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
23-03-2004	**Master's Thesis**	Aug 2003 – Mar 2004

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
MULTIPLE ACCESS INTERFERENCE CHARACTERIZATION FOR DIRECT-SEQUENCE SPREAD-SPECTRUM MCOMMUNICATIONS USING CHIP WAVEFORM SHAPING	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
Glen, Matthew, G., Captain, USAF	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)	8. PERFORMING ORGANIZATION REPORT NUMBER
Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way Street, Building 640 WPAFB OH 45433-7765	AFIT/GE/ENG/04-10

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/IFGD Attn: Mr. Mark Minges 2241 Avionics Circle DSN: 785-1608 x3401 WPAFB OH 45433-7334 e-mail: mark.minges@wpafb.af.mil	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This research characterizes Multiple Access Interference (MAI) effects on Direct-Sequence Spread-Spectrum Multiple Access (DS/SSMA) system performance through simulation in Matlab®, and explores the impact of multiple access code selection, chip waveform shaping, and multiple access code length on bit error rate (BER) for both synchronous and asynchronous multiple access networks. In addition, the simulated DS/SSMA model permits rapid research into the effects of additional factors on BER.

Prior to experimental testing, model validation is conducted through single user trials and by comparison with existing research for similar designs. For synchronous and asynchronous networks Gold coding improves BER by 7.5 and 4.0 dB, respectively, relative to aperiodic random spreading codes. Synchronous network results show that chip waveform shaping provides no significant BER improvement for the Blackman and Lanczos shapes. However, asynchronous network results show a potential BER improvement for Blackman and Lanczos shapes. Increasing code length from 31 to 511 resulted in a 7.5 dB BER improvement. Collectively these results directly relate changes in BER to waveform cross-correlation statistics.

15. SUBJECT TERMS
Spread Spectrum, Multiple Access, Code Division Multiple Access, Interference, Coding, Cross Correlation, Correlation Techniques, Auto Correlation, Communications Networks, Data Links, Networks, Radiotelephones

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Michael A. Temple, Ph. D. AFIT/ENG
a. REPORT	b. ABSTRACT	c. THIS PAGE	UU	78	19b. TELEPHONE NUMBER *(Include area code)* (937) 255-6565, ext 4279; e-mail: Michael.Temple@afit.edu
U	U	U			

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18