AN ABSTRACT OF THE THESIS OF

David H. Buck for the degree of Master of Science in Electrical and Computer Engineering presented on May 28, 1999. Title: Data Rate Improvements for the IEEE 802.11 Wireless Local Area Network Standard Through the use of Code Division Multiple Access and Turbo Coding.

Redacted for privacy

Abstract approved: _____

Mario E. Magaña

The widespread use of Wireless Local Area Networks (WLAN) and the desire for such products from different vendors to operate together has generated a movement towards standardization. Over the last decade, several organizations worldwide have researched and developed such standards, this includes the IEEE 802.11 committee. One of the important considerations in design and marketing WLAN products is the data rate supported by such products. This thesis deals with the development of a modification of the Direct Sequence (DS) physical layer standard in IEEE 802.11 to allow higher data rates beyond the 1-2Mb/s supported by the standard. More precisely, this thesis proposes using Code Division Multiple Access (CDMA) and turbo coding, an aggressive channel encoding technique, to improve the data rate performance 2-3 times over that in IEEE 802.11. A simplified transceiver design is presented and computer simulations are performed to verify the design and implementation considerations.

Data Rate Improvements for the IEEE 802.11 Wireless Local Area Network Standard Through the use of Code Division Multiple Access and Turbo Coding

by

David H. Buck

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented May 28, 1999
Commencement June 2000

Master of Science thesis of David H. Buck presented on May 28, 1999

APPROVED:

Redacted for privacy

Major Professor, representing Electrical and Computer Engineering

Redacted for privacy

Chair of Department of Electrical and Computer Engineering

Redacted for privacy

Dean of Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Redacted for privacy

David H. Buck, Author

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF TABLES

LIST OF ABBREVIATIONS

(AWGN)	Additive White Gaussian Noise
(AMPS)	Advance Mobile Phone System
(ACR)	Autocorrelation-peak to Cross-correlation Ratios
(ARQ)	Automatic Repeat Request
(BER)	Bit Error Rate
(CSMA/CA)	Carrier-Sense Multiple Access with Collision Avoidance
(CTS)	Clear to Send
(CDMA)	Code Division Multiple Access
(CCK)	Complementary Code Keying
(CW)	Contention Window
(CW)	Continuous Wave
(CCSK)	Cyclic Code Shift Keying
(CRC)	Cyclic Redundancy Check
(DBPSK)	Differential Binary Phase Shift Keying
(DQPSK)	Differential Quadrature Phase Shift Keying
(DSP)	Digital Signal Processing
(DS)	Direct Sequence
(DSSS)	Direct Sequence Spread Spectrum
(ECC)	Error Control Coding
(EC)	European Community
(FH)	Fast Hopping
(FCC)	Federal Communication Commission
(FIR)	Finite Impulse Response
(FCS)	Frame Check Sequence
(FER)	Frame Error Rate
(FDMA)	Frequency Division Multiple Access
(FHSS)	Frequency Hopping Spread Spectrum
(GMSK)	Gaussian Minimum Shift Keying
(HIPERLAN)	High Performance Radio Local Area Networks
(ISM)	Industrial, Scientific, and Medical
(IIR)	Infinite Impulse Response
(IR)	Infrared
(ISI)	Intersymbol Interference
(LBT)	Listen Before Talk
(LLR)	Log-Likelihood Ratio
(MBOK)	M-ary Bi-Orthogonal Keying
(MAP)	Maximum a Posteriori
(MAC)	Medium Access Control
(MPDU)	Medium Access Control Protocol Data Unit
(NPMA)	Nonpreemptive Multiple Access
(NRZ)	Non-Return-to-Zero
(NSC)	Non-Systematic Convolutional
(OSI)	Open Systems Interconnect

LIST OF ABBREVIATIONS (Continued)

(OCDM)	Orthogonal Code Division Multiplex
(PLL)	Phase Lock Loop
(PHY)	Physical
(PCF)	Point Coordination Function
(PA)	Power Amplifier
(PSD)	Power Spectral Density
(PDF)	Probability Density Function
(PG)	Processing Gain
(PN)	Pseudorandom Noise
(PPM)	Pulse Position Modulation
(QPSK)	Quadature Phase Shift Keying
(RF)	Radio Frequency
(RSC)	Recursive Systematic Convolutional
(RTS)	Request to Send
(SIR)	Signal to Interference Ratio
(SNR)	Signal to Noise Ratio
(SH)	Slow Hopping
(SOVA)	Soft Output Viterbi Algorithm
(SS)	Spread spectrum
(SFD)	Start Frame Delimiter
(SAW)	Surface Acoustic Wave
(TDMA)	Time Division Multiple Access
(U-NII)	Unlicensed National Information Infrastructure
(VA)	Viterbi Algorithm
(WINForum)	Wireless Information Network Forum
(WLAN)	Wireless Local Area Networks

Data Rate Improvements for the IEEE 802.11 Wireless Local Area Network Standard Through the use of Code Division Multiple Access and Turbo Coding

1. INTRODUCTION

As we continue to use computers to enhance life in our society, the need to network computers for data transfer and access to the Internet becomes increasingly important. Many network topologies have been developed over the past three decades. Recently, during this decade, a new network has emerged, the wireless local area network (WLAN). This network typically uses a radio frequency (RF) or infrared (IR) light beam to communicate information between members of the network. This is different than conventional networks, which typically use wires or cables to link the members of the network. As computer devices become more powerful, smaller in size, and incorporate more networking features, the limitations of a wired network become apparent. Laptops, palmtops and personal digital assistants designed for portability are becoming more widespread. As the growth in this market continues, so does the desire for a network connection that frees people from wired tethers.

The ability for a small device to remain connected to a network regardless of location within a small area has some definite advantages. In some cases, a wired network may be prohibitively expensive or not allowed in some structures in which a WLAN would be a welcomed solution. For example, installation of a wired network on a large factory floor can be quite expensive. Also, historical buildings or temporary offices may have strict rules about drilling holes in walls and running cables for a network. In these situations, a WLAN can be deployed rapidly without altering the structure and with no other cost than the equipment itself.

Work on wireless data communications began in the late 1970s, but it was not until the Federal Communication Commission (FCC) opened the 2.4 GHz Industrial, Scientific, and Medical (ISM) frequency bands in 1985 that commercial products began to be developed. By 1990, WLAN products using the ISM bands began to appear on the

market using direct sequence spread spectrum (DSSS) as well as other techniques. First generation WLANs were large devices that consumed large amounts of power and were not suitable for portable devices. These products were marketed for wireless links for workstations. However, most workstations in existing buildings were already connected to a wired network and creating a wired network in new construction was cheaper than the WLANs. So, the anticipated market for these products did not appear. Second generation WLAN products are being developed for small portable devices and have seen increased popularity over the first-generation WLANs [1].

With the popularity of WLAN products, a problem soon became apparent, interoperability. Products from different companies often would not work together because of the proprietary communication protocols. It became clear that standardization was needed for interoperability so products purchased from different vendors would function in the network. In the U.S., during the early 1990's, the IEEE 802.11 committee was established and the Wireless Information Network Forum (WINForum), an alliance between major computer and communication companies was formed. The European Community (EC) also initiated its own standards group, the High Performance Radio Local Area Networks (HIPERLAN) in 1991.

1.1 WLAN Standards

1.1.1 WINForum

WINForum's goal was to obtain unlicensed personal communication services (U-PCS) bands from the FCC to be used for data communications. In working with the FCC, WINForum developed a "spectrum etiquette" to be used in the proposed bands [1]. In developing the ISM bands, the FCC imposed no regulation on amount of airtime that a device could use. Because of this, an unlicensed ISM device could constantly radiate power that could affect the performance of a nearby WLAN (which usually transmits in short bursts) due to the interference generated by the other device. The "spectrum etiquette" developed by WINForum restricted the frequency and limited transmission time based on a device's transmit power to ensure a fair sharing of the spectrum

resources [2]. The etiquette also called for separate bands for asynchronous (data) transmissions and isochronous (voice) communications, as it argued isochronous communications required long and frequent access to the channel. Originally, WINForum requested 40 MHz of unlicensed space in the PCS band (1850-1990 MHz). However, in 1994 the FCC approved only 2-10 MHz U-PCS bands, citing that the PCS band was not designed for high date rate applications. The FCC did, however, adopt the "spectrum etiquette" proposed by WINForum. As a result, one band was assigned for asynchronous applications and the other for isochronous [3].

In 1995, WINForum and Apple Computer petitioned the FCC for 300 MHz in unlicensed bands in the 5 GHz band for wireless applications up to 20 Mb/s called the Unlicensed National Information Infrastructure (U-NII). In 1997, the FCC approved 3-100 MHz bands with minimal technical requirements for each band. Such requirements did not specify any particular access technique, thus allowing a wide possibility of development to occur. The FCC did not adopt the spectrum etiquette for these bands because it believed that such requirements would preclude the development of certain types of wireless systems [4].

1.1.2 HIPERLAN

HIPERLAN had a different beginning than IEEE 802.11. The IEEE 802.11 standard was fueled by the desire for products to interoperate, while HIPERLAN was driven with no existing products or regulations. A set of requirements was established and the group set about designing a standard to meet them. In 1995, they released a draft standard that contained specifications for a medium access control (MAC) and Physical (PHY) layer, which make up the lowest two layers in the Open Systems Interconnect (OSI) model. The standard called for data rates of 2-23 Mb/s that operated in the 5.2 GHz or 17 GHz bands and support for ad-hoc networking with a multi-hop capability. The multi-hop capability allows a user to forward the packet of another user which increases the range of the cell beyond the radio range of an individual user. A nonpreemptive multiple access (NPMA) protocol is used, which is different than that used in IEEE 802.11 (explained below). This protocol is a carrier sense type, also known

as listen before talk (LBT), that allows users to transmit a fix time after detecting the channel is clear. The MAC layer also supports data encryption and power saving, which are important aspects for data privacy and mobile applications [3]. The PHY layer uses Gaussian minimum shift keying (GMSK) for modulation and employs a BCH (31,26) block error correction code. Sixteen 31 bit codewords are interleaved to form a 496 bit block and up to 47 blocks are used in a packet [5].

1.1.3 IEEE 802.11

As previously mentioned, IEEE 802.11 began as a desire by customers for different WLAN products to operate together. In 1990, the committee began it's work on the standard. By 1996, the majority of the standard had been formalized. The standard focuses on the lowest two levels in the open systems interconnect (OSI) 7 layer model: the medium access control (MAC) and the physical (PHY) layers. What emerged was a single MAC layer definition that interfaced with 3 different PHY layers. The standard was developed for a data rate of 1-2Mb/s depending on the modulation used in the PHY layers. The standard was approved in 1997, though some work is still continuing, mainly around increasing the data rate.

Two network topologies are supported by the MAC standard: ad hoc and infrastructure networking. The first allows users (known as stations) to form a network independent of the backbone network (usually the wired network). An example of this is a group of users in a conference room that connect together to form an ad hoc network for file sharing. The second topology, infrastructure, includes a group of users in which one station can access the backbone network (commonly called an access point). Through a combination of these two topologies, a wireless network of arbitrary size and complexity can be formed [5, 6]. The MAC allows for both asynchronous and time-critical traffic.

Asynchronous traffic is usually associated with data transfer which occurs in short bursts and at random times for each of the users. For this type of traffic, a carrier-sense multiple access with collision avoidance (CSMA/CA) protocol is used. In CSMA, each user senses the channel for traffic and when free, transmits a packet. However, since all

users are listening, the period of time after a station finishes transmitting a frame is the greatest for collisions. So the CSMA/CA protocol in 802.11 uses a random backoff time for each station to minimize the probability of a collision. The backoff time is uniformly distributed over a contention window (CW), the length of which is controlled based on the number of users. Also, for larger frames, request to send (RTS) and clear to send (CTS) packets are used to solve the hidden node problem encountered in carrier sense protocols [5, 6].

Time-bounded services are supported by a point coordination function (PCF) in which a PCF station takes control of the channel and allows one station to have access to the channel. This is done by sending a control frame to all other users in order to silence them for a specified period. The length of time a station can have access to the channel is controlled by the system load [5, 6].

The IEEE 802.11 standard allows three different PHY layers to be used with the MAC, an infrared (IR), and two radio frequency (RF) physical layers; direct sequence spread spectrum (DSSS) and frequency hopping spread spectrum (FHSS). We shall consider only the RF layers, since the work presented in this thesis focuses on the direct sequence (DS) PHY layer. The RF PHY layer operates in the 2.4 GHz ISM band where 80 MHz of bandwidth are available. This ISM band was chosen over the 900 MHz and 5.7 GHz bands because the 2.4 GHz band is available in many other countries around the world, hence expanding the market for IEEE 802.11 compliant products [5].

1.2 Spread Spectrum Systems

Spread spectrum (SS) systems, as the name suggests, spreads the original spectrum of the signal to a much wider bandwidth. These systems were originally developed over the last 50 years for military communication applications. For most of that time, SS systems research and development had been cloaked in secrecy. But, over the last 20 years, significant research has been published openly allowing individuals and companies to explore commercial applications of this once secret technology [7]. With developments over the last decade in small and powerful electronics, many SS systems

have found commercial applications. SS systems have a number of interesting properties that make them well suited for both military and commercial applications.

Since the transmit energy is spread over a wide bandwidth in SS systems, it is possible to hide transmissions in background noise, which makes them difficult for an adversary to detect. This is also useful in commercial applications where bandwidth availability is at a premium. Due to the spreading, the power spectral density (PSD) of a transmission can be made very low. This can allow SS systems to operate in frequency bands that are already occupied by traditional narrow–band communications systems. In fact, the cellular IS-95 standard, which is a SS system, operates in the same frequency bands as the older analog AMPS cellular system.

SS systems use pseudorandom sequences to accomplish the spreading. A pseudorandom sequence is generated by some type of deterministic mathematical algorithm. However, by looking at the sequence alone, it appears to be totally random. Since the sequence is generated by an algorithm, it is possible to regenerate it in the receiver to recover the transmitted data. If the algorithm uses some other information in generating the sequence, called a key, then an adversary trying to intercept the message will have a difficult time recreating the sequence without knowledge of the key. Thus, SS systems can provide information security, which is very important in military communications. However, information security has also become important in commercial applications. Both HIPERLAN and IEEE 802.11 have security features specified in their respective standards.

Another important property of a SS system is the anti-jamming capability. An adversary that attempts to jam the transmitted signal may use a variety of tactics, but most use the idea of corrupting the channel with another transmission to disrupt communications. To receive a SS signal, the receiver in a SS system correlates the received signal with a locally generated copy of the pseudorandom sequence. So, if the adversary's transmission is not correlated with the spreading sequence, then the receiver can reject a good portion of the energy [7]. This is commonly referred to as the interference rejection property. In commercial applications, the jamming may not be intentional, but might be the result of another system or by certain channel conditions.

Nevertheless, the interference rejection in a commercial SS system is useful in improving the quality of transmission.

A frequency hopping spread spectrum (FHSS) system uses a pseudorandom number generator to select the transmit carrier frequency at any given time from a set of frequencies. This is shown graphically in figure 1.1. The transmitter spends only a small fraction of time in each frequency band, so the effect is to spread the transmit energy over a wide band of frequencies. The receiver in FHSS generates a replica of the random

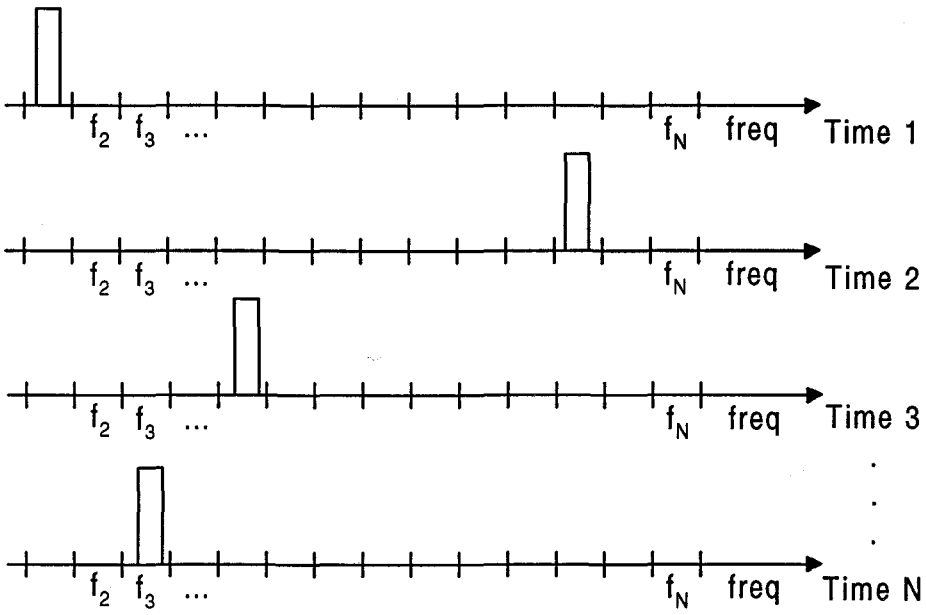


Figure 1.1 Frequency hopping spread spectrum

sequence so it can track the frequency changes and recover the signal. FHSS systems are characterized as either slow hopping (SH) or fast hopping (FH). A slow hopping system will transmit multiple symbols over one frequency before switching to another, and a fast hopping system transmits one symbol over multiple frequencies.

IEEE 802.11 specifies a physical layer that uses SH/FHSS. The 2.4 GHz ISM band used by 802.11 is 83 MHz wide and is divided into 79 channels, each 1 MHz wide. From these 79 channels, 3 sets of hopping patterns are defined, each set contains 26

patterns also called sequences. Each set of sequences contains unique channel numbers not used in the other two sets. This is done to allow a cell to switch to another set of hopping sequences should the interference from other 802.11 FHSS cells become too great. Each sequence was carefully chosen to have certain properties. First, since more than one cell (grouping of stations) will operate in the same geographical region, the sequences were chosen so that there is a low probability that the same channel will be used at the same time by two different cells. This is important to reduce the interference of multiple cells in order to increase the throughput of the network. Second, consecutive hops in each sequence are at least 6 MHz in frequency and the sequence uses all channels assigned to the set [5]. This ensures that the transmit energy is effectively spread over the entire bandwidth. Given these properties, each cell in the network within the same geographical area is assigned a different sequence. This theoretically allows for quite a few collated cells. This idea of using unique sequences to separate users is known as Code Division Multiple Access (CDMA) which will be discussed in more detail later. The FHSS layer uses 2-level Gaussian frequency shift keying (GFSK) modulation for the 1 Mb/s data rate and 4-level GFSK modulation for 2 Mb/s [8].

Direct sequence spread spectrum (DSSS) uses a different technique than frequency hopping to spread the spectrum of the signal. In the transmitter, a binary pseudorandom noise (PN) sequence (each binary value is called a chip) is generated at a higher rate than the data rate. The PSD of the PN sequence is fairly flat and has a much wider bandwidth than the data. The transmitter multiplies the data with the higher rate PN sequence, which spreads the original spectrum in frequency as shown in figure 1.2. An important parameter in a DSSS system is the processing gain (PG), which is a measure of the amount of spreading. The most common definition of PG is ratio of the chip rate to the data rate. Another measure of PG is the ratio of the spread-spectrum bandwidth to the symbol or modulation bandwidth. The two definitions are not the same, as will be shown below [9, 10].

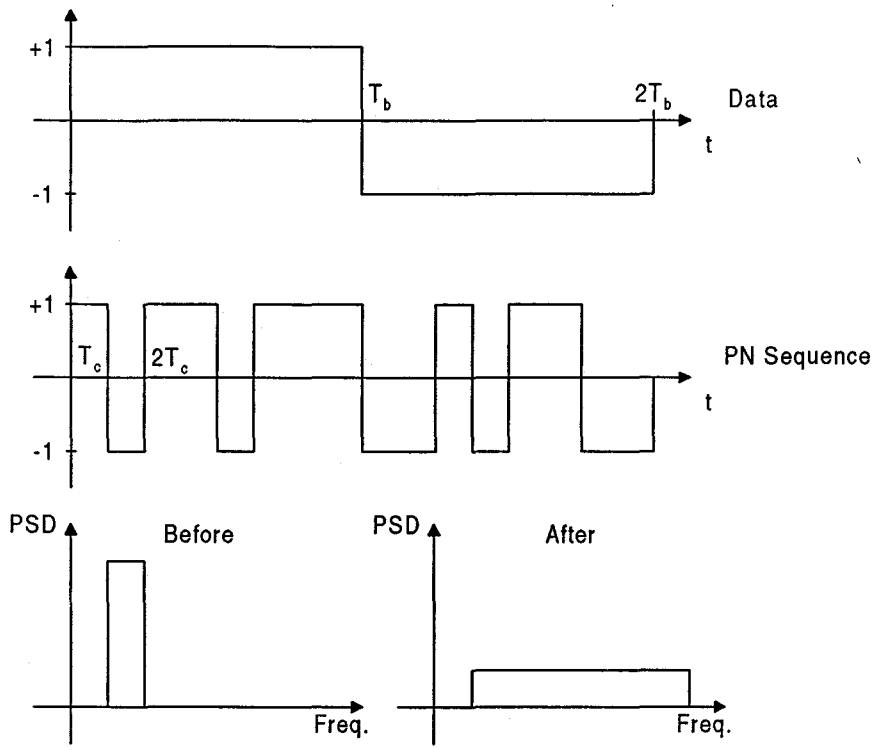


Figure 1.2 Direct sequence spread spectrum

The DSSS physical layer in IEEE 802.11 uses an 11 chip Barker PN sequence for the symbol spreading. The 1 Mb/s data rate uses differential binary phase shift keying (DBPSK) and the 2 Mb/s data rate uses differential quadrature phase shift keying (DQPSK). Differential modulation is used so the receiver does not need an exact phase reference, but can demodulate the symbols by detecting large phase changes of the modulated symbols [7]. For DBPSK, one data bit is modulated into one symbol, which results in a passband modulated bandwidth of 2MHz for a 1Mb/s data rate. When the DBPSK modulated data is spread using the 11 chip sequence, the passband bandwidth expands to 22MHz. Thus the PG is 11 or 10.4dB using either of the PG definitions described above. For 2Mb/s, the DQPSK modulated bandwidth is still 2MHz, which is spread using the 11 chip Barker sequence, resulting again in a spread-spectrum bandwidth of 22MHz. For this case, using the first definition of PG, we obtain a PG of

5.5 or 7.4dB, and using the second definition, we obtain a PG of 11 or 10.4dB. Therefore, the two definitions are not the same and care must be used in assuring the correct definition is used.

In defining the regulations for the 2.4 GHz ISM band, the FCC used the second definition to define the PG. It mandated that DSSS devices operating in this band must have a PG greater than 10dB [9]. Thus, the DSSS physical layer in IEEE 802.11 has the minimum processing gain required to satisfy the FCC regulations. The minimum PG was chosen to minimize the transmission bandwidth due to the limited bandwidth in the ISM band and concerns for implementation complexity. The standard defines 11 DSSS channels in the 2.4 GHz band, each channel center is spaced 5MHz apart. The standard also specifies the transmit spectrum mask requirement. For DSSS cells to operate without interference using this transmit mask, the cells must be placed at least 30MHz apart, which allows 3 cells to share the band [5, 6]. With more aggressive filtering, up to 4 cells could share the band. This is fewer than the number of FHSS cells that can be located together. However, because of the interference rejection of a DSSS system, groups of cells can be placed much closer together than for FHSS. So, it is somewhat difficult to tell which system has the advantage in cell density [11].

Because of the relatively small processing gain used in the DSSS layer, choosing the PN sequence was an important consideration. The Barker PN was chosen because it is well known for its optimal autocorrelation properties. The autocorrelation of this sequence exhibits a single peak and uniformly low sidelobes, which increases demodulation performance in the presence of noise and multipath fading [9]. The standard specifies a scrambler to be used to scramble all bits transmitted by the DSSS physical layer. This scrambler is used to whiten the transmitted data and avoid any continuous wave (CW) transmission that might violate transmit mask regulations.

The frame format for the DSSS physical layer is shown in figure 1.3. The SYNC field contains all 1's that are used for synchronization at the receiver. The start frame delimiter (SFD) contains the physical layer dependent parameters. The SIGNAL field contains the modulation information used in the data transmission. The SERVICE field is not used and reserved for future use.

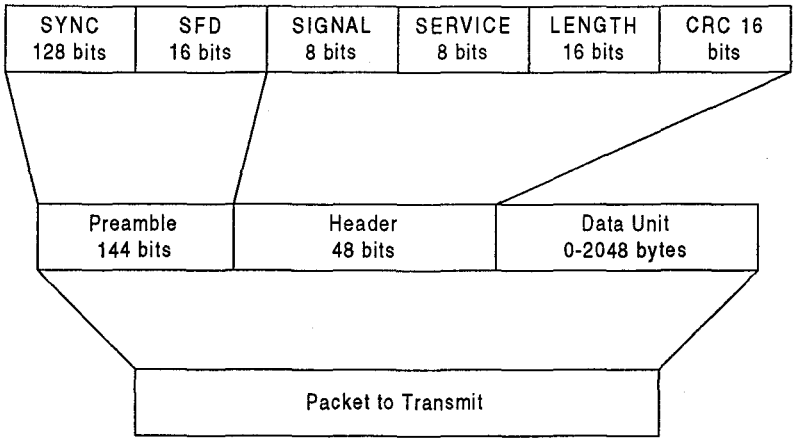


Figure 1.3 IEEE 802.11 DSSS packet structure

The LENGTH field contains the time in microseconds required to receive the data (packet length). The SIGNAL, SERVICE, and LENGTH fields are protected by a CRC-16 frame check sequence (FCS) to determine if any errors have occurred during transmission in these fields. The Medium Access Control Protocol Data Unit (MPDU) contains the transmitted data. The data field has a variable length from 0-2048 bytes. This field is not protected by any error detection or error correcting code [6].

1.3 Code Division Multiple Access

Code division multiple access (CDMA) is an extension of a spread spectrum system. CDMA uses multiple spreading codes to distinguish between the different users in the system. This is different than frequency division multiple access (FDMA) which allots a frequency band for each user or time division multiple access (TDMA) which allots a time slot for each user. In CDMA all users use the same frequency band and can transmit at the same time. The users are separated from each other at the receiver by the unique codes assigned to each one of them. To do this, the receiver correlates the received signal with the spreading code for the desired user. So, for this mechanism to

work, the spreading codes assigned to each user must not be correlated with each other so the receiver can select the desired user's data. There are many ways to generate PN sequences that exhibit low cross-correlation properties, and the choice of these codes is an important design consideration in the development of a CDMA system. CDMA can be applied to either direct sequence (DS/CDMA) or frequency hopping (FH/CDMA) systems. However, most modern commercial systems are DS/CDMA, although we have seen the IEEE 802.11 FHSS physical layer is actually a FH/CDMA system. Since DS/CDMA systems are far more common, henceforth, CDMA will refer to DS/CDMA.

CDMA has found many military applications over the last 50 years, but during this decade CDMA has also found commercial applications, especially in the cellular telephone industry. A cellular system has many requirements such as large capacity, low cost of operation, variety of services, and small inexpensive phones for users. First generation analog FDMA cellular system (AMPS) put into service in the 1980's quickly became saturated in many markets as the popularity of cellular phones grew. As a result, it was realized that a digital cellular system was needed to solve the capacity problem and to provide new services to customers. In 1993, the CDMA IS-95 standard developed by QUALCOMM was adopted as the next generation North American cellular standard. It showed promise of capacity improvement of 10-15 times over the analog systems and capacity greater than other digital standards proposed [10, 12].

1.4 IEEE 802.11 Data Rate Improvements

After the IEEE 802.11 draft standard was released in 1997, the committee created two working groups to study data rate improvements for the standard. At the same time, many companies developing 802.11 products were also investigating ways to increase the data rate. This led to the development of many proprietary communication protocols to accomplish the improvement. The idea was to support multiple modulation schemes that would allow on the fly data rate switching. In fact, the 802.11 frame structure was developed for this capability. The 802.11 packet header is always sent at 1Mb/s using DBPSK modulation. The header contains the data rate information for the rest of the

packet, so the receiver can switch to the other modulation to receive the rest of the packet. Thus, WLAN products could contain the 802.11 compliant modulation schemes so that they would interoperate with products from other vendors. But if a network contained multiple products from the same vendor, the devices could switch to the proprietary modulation to increase the data rate. This is a similar situation that occurred in the first generation of 56k analog modems before the V.90 standard was developed. The goal data rate set by many was 10Mb/s. This is equivalent to the data rate of many wired networks, which would allow WLANs to become wire network replacements, not just supplements. This is indeed a strong marketing tactic to increase the appeal of WLANs since customers would not have to suffer lower data rates by converting to wireless networks.

An excellent discussion of a variety of modulation techniques (including some proprietary ones) to increase the data rate of 802.11 is given in [13]. All of the techniques discussed in [13] are for the DSSS physical layer. This is because the FHSS physical layer has some serious limitations that make higher data rates difficult to achieve. First, the FCC part 15 (15.247) regulations requires the transmit power spectral density (PSD) be down by 20dB at the band edges. This 20dB bandwidth keeps the energy of one channel from spilling over into the adjacent channel. Also, the regulations require at least 75 hopping channels in the 2.4 GHz band, which limits the maximum bandwidth of each channel to 1.1MHz [14]. Thus, bandwidth cannot be expanded to allow higher data rates and the strict transmit mask requirements severely impair the performance of more complex modulation schemes. The FHSS layer also has a transmit power disadvantage compared to the DSSS layer because it uses less efficient modulation. Thus, the 1W (30dBm) transmit power limit imposed by the FCC would likely be encountered by a more complex FHSS modulation to maintain a radio with the same range and bit error rate (BER) performance. An overview of a few of the proprietary techniques is discussed next, followed by the work done by the working group of the IEEE 802.11 committee to improve the data rate in the 2.4 GHz ISM band.

Harris Semiconductor developed a 11Mb/s physical layer using M-ary bi-orthogonal keying (MBOK) modulation scheme. A diagram of the transmitter is shown

in figure 1.4. This modulation scheme uses a grouping of 3 bits to choose between a set of 8 Walsh functions for both the I and Q channels.

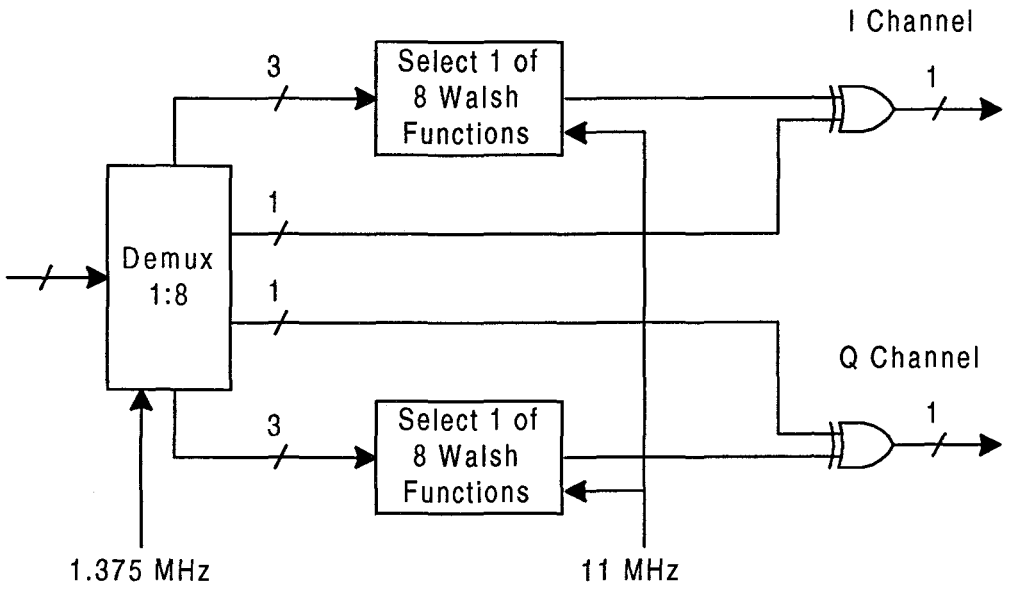


Figure 1.4 MBOK modulation

Seq #	Walsh Functions
1	00000000
2	01010101
3	00110011
4	01100110
5	00001111
6	01011010
7	00111100
8	01101001

Table 1.1 8-bit Walsh functions

Walsh functions are sets of binary value sequences whose cross-correlation values are zero, making them perfectly orthogonal. This allows a correlating receiver to distinguish one sequence from another, allowing the transmitted bits to be decoded. The set of 8-bit Walsh functions is shown in table 1.1. The XOR gates in figure 1.4 make use of the fact that the complement of a Walsh function is also orthogonal, which allows two more bits per symbol to be encoded essentially free. The demultiplexer clock runs at 1.375 MHz and the chip clock runs at 11 MHz so that the transmit bandwidth after upconversion is still 22 MHz [13].

Bell Labs (a division of Lucent Technologies) developed a physical layer based on pulse position modulation (PPM). The Barker code used in 802.11 has a two-valued autocorrelation. Because of this sharp autocorrelation function, this sequence can be cyclically shifted in position, resulting in a sequence that is trans-orthogonal to the original. The two sequences are not purely orthogonal since the cross-correlation between the two sequences is –1. However, this trans-orthogonal behavior only slightly degrades the performance of the system. Of the 11 possible positions, 8 are used to allow 3 bits to be encoded per symbol. An additional bit can be encoded by selecting the complement of the shifted sequence, just like MBOK modulation. Futhermore, if both the I and Q channels use this modulation, then a total of 8 bits can be encoded into 1 symbol. Lucent Technologies has devised a variant of this technique by overlapping the adjacent symbols by 3 positions, since only 8 positions of the 11 are used. This increases the data rate by about 20% [13].

A couple of companies have developed modulation schemes based on orthogonal code division multiplex (OCDM). This method uses multiple spreading codes to send streams of data over orthogonal channels. OCDM is essentially CDMA except that in CDMA each user is assigned a unique code. In OCDM, the original IEEE 802.11 CSMA/CA protocol is used, but one user can use multiple codes to increase the data rate. A block diagram of a transmitter for OCDM is shown in figure 1.5. This technique increases the data rate, but does not suffer from the multiple access problems of CDMA, such as the near-far problem. Therefore, the added complexity usually associated with a CDMA receiver (strict power control, etc.) is not required.

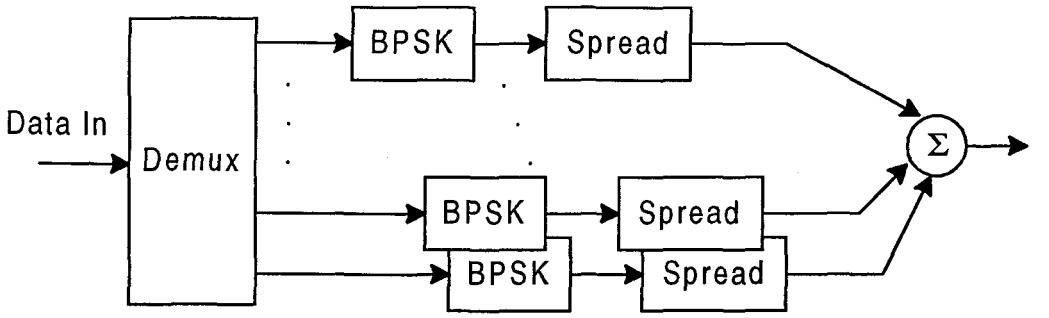


Figure 1.5 Orthogonal code division multiplex modulation

Sharp Electronics has announced that its 10Mb/s OCDM modem will use cyclic code shift keying (CCSK) Barker words for the spreading codes. Golden Bridge uses Walsh codes in its modem for the spreading, which results in slightly better performance due to the perfect orthogonality of the codes [13]. One drawback of this modulation scheme is that when the individual channels are summed together, the possibility of large amplitude variations in the transmit waveform exists. This causes a large peak-to-average power ratio which can cause difficulties in the design of linear power amplifiers. However, if the spreading codes are chosen carefully, then this peak-to-average ratio may be minimized.

In January 1999, the IEEE 802.11 standard committee approved complementary code keying (CCK) modulation for high data rate standard for the 2.4 GHz band. CCK modulation was developed jointly by Harris Semiconductor and Lucent Technologies. The 8 data bits are modulated in both I and Q channels simultaneously to create 64 code words, much higher than the 8 codewords used in MBOK. This allows for future higher data rates. The CCK modulation is more robust against multipath than MBOK which allows better system performance under these conditions [15].

1.5 Error Control Coding

The subject of error control coding (ECC) has filled the pages of many books. ECC coding is as much a subject of art as it is of science. In 1948, Claude Shannon proved that it was possible to transmit data with high reliability provided that the data transmission rate did not exceed the channel capacity C. For an additive white Gaussian noise (AWGN) channel, the channel capacity is given by

$$C = B \cdot \log_2[1+S/N] \quad (\text{bits/sec}) \qquad (1.1)$$

where B is the channel bandwidth, S is the average signal power, and N is the noise power. Shannon did not indicate how this reliability could be achieved in real world communication systems. Since 1948, extensive research has been done to devise ECC schemes to accomplish this goal [10, 16]. ECC improves the reliability of transmitted data by encoding the data with some extra information to introduce redundancy. The idea is to make a particular block or string of data unique from all other possible data blocks or strings. The more unique the encoded data is, the more likely the receiver will recognize the data even if some of the bits are corrupted by the channel. This allows a portion or all of the errors caused by the channel to be detected and in some cases corrected, thereby improving the transmission reliability.

ECC has many useful benefits that make it attractive for a wide variety of applications. Simple schemes can be used to detect errors in a block of data. Error detection codes are often used in communications systems and are almost always required for systems that transmit data. Data transmissions have a very low tolerance for error, so it is essential that an error detection mechanism be used to detect these errors. Once errors are detected in the data, the receiver can generate a request for retransmission. A good example of these is in networks protocols. IEEE 802.11 specifies a cyclic redundancy check (CRC) code to check for errors in the header portion of the packet [6]. The 1997 release of the IEEE 802.11 standard does not specify a CRC code for the data, but likely this is done at a higher network layer. Another application of ECC is in systems that have limited transmit power. If a error correction code is used, then the bit error rate (BER) can be improved and coded system will require less transmit power than the equivalent uncoded system. The difference in transmit power required for

a coded and uncoded system to maintain the same BER is known as the coding gain. Coding gain is an important measure to gauge the effectiveness of an ECC and is often used to compare different ECC schemes. An example of such a system that uses error correction codes is deep space probes. Since the probes have very limited transmit power and are so far away, very complex error correction codes are required just to make communication possible. Also, since transmissions take many minutes to reach earth, retransmission protocols alone are not practical in these applications.

ECC can have many benefits, though at a cost. ECC requires more bits be transmitted than the uncoded system. This will increase the transmission bandwidth for a fixed data rate or will require a reduced data rate to maintain the same bandwidth. ECC also requires more processing in the transmitter and receiver which increases complexity. However, as modern electronics and digital signal processing (DSP) gets cheaper and more powerful, the introduction of ECC into communication systems becomes more practical.

Traditionally, coding methods have been divided into two categories: block codes and convolutional coding. Block codes take data blocks of k bits and encode them into n bit codewords where $n > k$. Since only 2^k of the possible 2^n codewords are used, the codewords are chosen to maximize the difference between the codewords. The number of bit positions in which the two codewords differ is defined as the Hamming distance d. The minimum Hamming distance, d_{min} of a code is the smallest d for all possible codewords. The minimum Hamming distance defines the error detection and error correction capability of a code. If $d_{min} > s+1$, then the code is capable of detecting s errors in the codeword. If $d_{min} > 2t+1$, then the code is capable of correcting t errors in codeword. The code rate of a block code is defined as $R = k/n$ and is a measure of the efficiency of the code [16].

Convolutional codes are much different than block codes. In principle, a convolutional code converts a data stream into one long codeword while a block code breaks it up into blocks and encodes each block separately. A convolutional encoder uses a shift register to insert redundancy into the data stream. Since the shift register has memory, the current encoding symbol is affected by previous symbols. This differs from

block codes which are memoryless in that the codewords do not depend on previous codewords. A block diagram of a simple convolutional encoder is shown in figure 1.6.

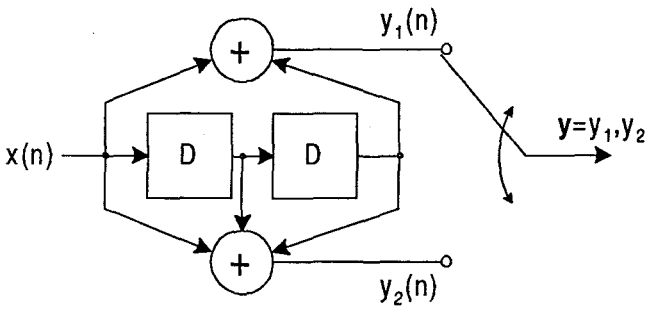


Figure 1.6 Rate 1/2 convolutional code

The rate of this encoder is 1/2 since for each input bit, the encoder produces two coded output bits. The commutator switches at twice the data rate to provide a serial output stream that is double the information rate. As each input is fed into the shift register, the values of the memory elements are tapped off and added modulo 2 to form the coded output bits. The constraint length K of a convolutional code is defined as the number of output bits affected by a single input bit. Thus for this encoder, the constraint length is 3, and in general a convolutional code with m memory elements in the shift register, the constraint length is given by $K = m+1$.

The convolutional encoder can be viewed as a finite impulse response (FIR) filter. By viewing it this way a difference equation relationship can be determined for each output $y_i(n)$ as function of the input $x(n)$. For the encoder in figure 1.6, the difference equations are

$$y_1(n) = x(n) \oplus x(n-2) \tag{1.2}$$

and

$$y_2(n) = x(n) \oplus x(n-1) \oplus x(n-2) \tag{1.3}$$

The impulse responses can be determined by inspection as the coefficients of difference equations. The impulse responses are often called the generating sequences or generating

polynomials and are written in vector form (either binary or octal). For this encoder, the generating polynomials are $g_1 = 101_2 = 5_8$ and $g_2 = 111_2 = 7_8$. With this representation, in general we can formulate each output as

$$y_i(n) = \sum_{l=0}^{m} x(n-l) \cdot g_i(l) \qquad (1.4)$$

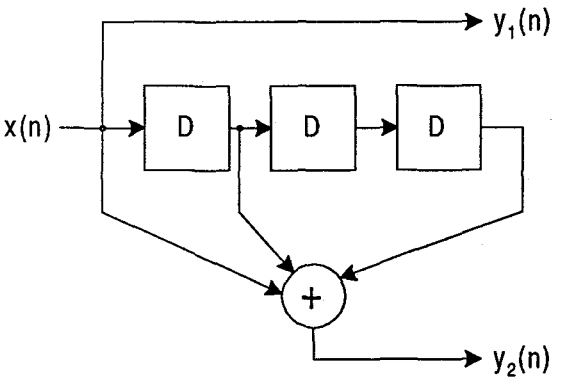


Figure 1.7 Rate 1/2 systematic convolutional code

which is the discrete convolution of the input with the impulse response, hence the name convolutional code. A systematic convolutional code is defined as a code in which the original input data shows up unchanged in the coded output stream. An example of a systematic code is shown in figure 1.7. In contrast, a non-systematic code does not output the input directly to the output stream as is the case for the code in figure 1.6 [17].

Convolutional codes can also be viewed as a finite state machine. By considering all possible inputs and state conditions we can construct a state diagram. The state diagram gives a visual picture of the structure of the code and shows the state transitions. Figure 1.8a shows the state diagram for the convolutional code in figure 1.6. The labeling on each branch of the state diagram is as follows: the first number indicates the input bit to the encoder and the last two numbers give the output bits. Another visual tool for convolutional codes is the trellis diagram. The trellis diagram includes the information contained in the state diagram, but it shows the state transitions as a function

of time. Figure 1.8b shows the trellis diagram for the convolutional code in figure 1.6. Each node along the horizontal axis indicates a time step. The labeling on each branch in the trellis indicates the output bits corresponding to that state transition. The trellis is useful in seeing path traces for different inputs and is useful for the Viterbi decoding algorithm described below.

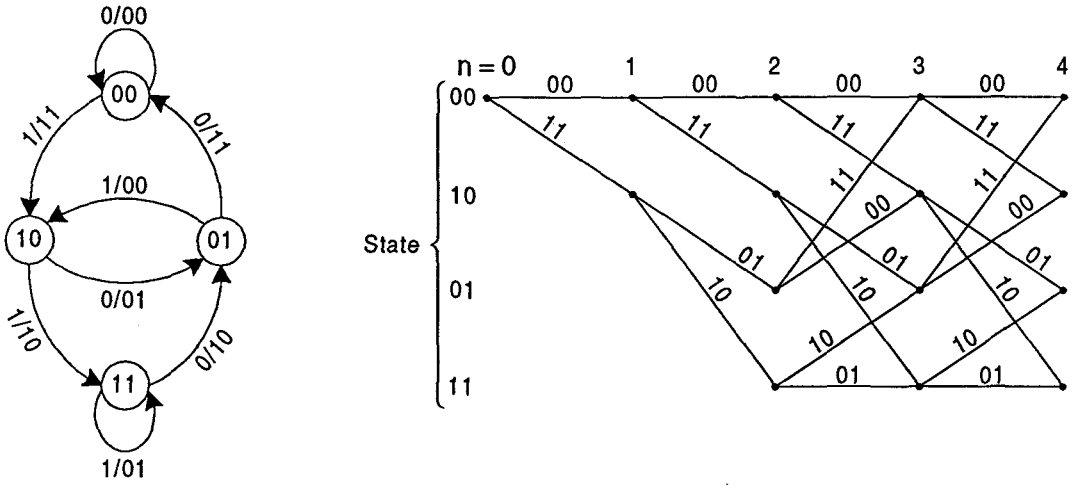


Figure 1.8 Visual diagrams for convolutional code in figure 1.6

There are several methods that can be developed to decode a convolutional code, but by far the most widely accepted and used is the Viterbi algorithm (VA). The VA is a maximum likelihood algorithm that takes advantage of the trellis structure to reduce the complexity of the decoding process. For a given input sequence, the encoder traces a path through the trellis to determine the codeword. On the receiver end, the decoder's goal is to retrace that path to regenerate the input bits. Since errors are assumed to have occurred in the received codeword, the VA computes the most likely path through the trellis with the given received codeword. Two major simplifications were realized by Viterbi. First, by observing the trellis it can be seen that only two paths terminate on

each node. This means that at each node a measure of likelihood, called a metric, can be computed for each branch and only the largest metric needs to be stored corresponding to the most likely path. Second, Viterbi realized that the computation of the branch metrics at each node was independent of the previous nodes. This allows for a recursive procedure in decoding which allows fast decoding to be implemented [10,18].

Since the Viterbi decoding algorithm computes a path through the trellis and not the location of errors in the received codeword, determining the minimum Hamming distance of a convolutional code is more difficult than for block codes. Also, since convolutional codes do not have fixed codeword lengths like block codes, the Hamming distance will vary with codeword length. Thus, the error correcting capability of convolutional codes is expressed using the minimum free distance d_{free} which can be determined through exhaustive computer search. A convolutional code can correct t errors if $d_{free} \geq 2t+1$ and the convolution code in figure 1.7 has a $d_{free} = 5$. Good convolutional codes with maximal free distance are determined through computer search and are listed in most ECC textbooks such as [17].

1.6 Turbo Codes

In 1993, Berrou, Glavieux, and Thitimajshima announced a new parallel concatenated code method they called turbo code. The word turbo was chosen due the similarity of operation of these codes to a turbocharger, which uses the exhaust gases of an engine to boost the manifold pressure in the intake (a feedback structure to improve performance). Turbo codes represent a break through in coding in that they promised BER performance near the Shannon bound for low E_b/N_0 ratios. This results in a coding gain of up to 10 dB or more over an uncoded system and a coding gain of 4-5 dB over current systems that use conventional block or convolutional codes. This coding gain allows the system to use much less transmit power to achieve the same performance which can extend the battery life of mobile devices, reduce transceiver complexity, and reduce the overall cost per unit of the device.

Turbo encoding works by combining two relatively short and weak convolution codes together in a parallel structure to make a very strong code. Interleaving is used in the turbo code structure in order to improve a code's performance in the presence of burst errors. The decoding is performed by using an iterative procedure. This means that after each decoding cycle, useful information gained in decoding is fed back and can be used by the next cycle to improve performance. Thus, as the number of decoding iterations increase, the BER of the received sequence improves.

Turbo codes have several properties that make them suitable for application in WLAN. First, as already mentioned, turbo codes improve transmission power efficiency by providing large coding gains. This is especially important in a multi-access environment because reduced transmit power reduces the interference between users. This translates into better overall network throughput and increased capacity in dense wireless network situations. Second, WLAN systems use large packet sizes which turbo codes need to operate effectively. Third, because turbo codes are based on convolutional codes they are well suited to IEEE 802.11 applications due to the variable packet length. Fourth, turbo codes exhibit a property that the attainable BER levels out at about 10^{-6}, which can be too high for sensitive data. However most WLAN systems (including IEEE 802.11) have additional protection through automatic repeat request (ARQ) protocols to retransmit packets in error. Thus, the BER limitation inherent in turbo codes is not a major drawback for applications in WLANs.

1.7 Thesis Content

This thesis considers how a combination of CDMA and channel encoding can be used to improve the data rate of the IEEE 802.11 standard. CDMA has been shown to be a proven method in providing robust and efficient digital communication. In this thesis a modified CDMA method will be used to increase the data rate of the IEEE 802.11 standard, known as orthogonal code division multiplex (OCDM). The OCDM modulation allows a flexible approach to data rate improvements. Indoor communications systems are often limited by interference from other system users, not

by background noise. Multipath interference can also severely impact the performance of such systems. In these situations it can be difficult to maintain adequate system performance and data throughput. CDMA and OCDM systems have a degree of built in immunity to both problems. The processing gain inherent in SS systems allows the system to reject interfering signals. Also, since unique codes are assigned to users in a CDMA system, or links in a OCDM system, interference can be lowered by reducing the number of users (links). In the case of OCDM, if the system experiences a high level interference, it can lower the number of simultaneous data links (decrease data rate) to reduce interference. Since the OCDM system is interference limited, there is a gradual reduction in system performance as the number of links is increased. Thus, by changing the data rate gradually a balance between data rate and system performance can be achieved, which will maximize overall system throughput. In contrast, the other modulation schemes discussed in the previous section allow basic data rates of 1Mb/s, 2Mb/s, and enhanced data rates of usually 5.5Mb/s and 11Mb/s. This leaves gaps in the available data rate. In the OCDM system proposed in this thesis, data rates can be changed in 1Mb/s or 2Mb/s steps, allowing for a more flexible approach in dealing with interference.

The IEEE 802.11 standard does not specify error correction coding (ECC) to be used to improve the performance of the radio link. This is contrary to the current trends in wireless digital communications where most systems have some form of ECC. ECC can do much to enhance the reliability of a digital communication system. First, the ECC can improve the bit error rate (BER) of the transmitted data to reduce the number of frame errors (FER). This increases the throughput of the overall system, by reducing the number of frame retransmissions. Second, the coding gain of the ECC allows the transmitter to reduce the amount of radiated power, which can lower the power consumption of small battery powered devices. Third, if an interleaver is used in the ECC, resistance to burst errors caused by multipath degradation or other user interference can be increased.

Since the processing gain used in the IEEE 802.11 standard is small, it will be shown that ECC is required to use OCDM modulation. This thesis proposes using a turbo code for ECC due to the large coding gains attainable for relatively short constraint

length codes. The coding gain of the turbo code will allow multiple links in the OCDM system, thereby increasing the data several times that of the basic 1 and 2 Mb/s physical layers of the IEEE 802.11 standard.

2. IEEE 802.11 DSSS COMPLIANT DESIGN

In this chapter we will consider the transmitter and receiver design for a DSSS IEEE 802.11 compliant device. The development of the IEEE 802.11 compliant design is necessary for two reasons. First, since the standard is relatively new, not much exists in open publication about the design and performance of the DSSS physical layer. Thus, design and simulation are necessary to obtain a baseline of performance to compare with simulation results of the higher rate system developed in the next chapter. Second, the higher rate system will be a modification of the basic DSSS compliant physical layer. Thus, it is important to understand the basic layer before it can be modified for higher data rates.

Some assumptions are necessary to simplify the system design to a manageable level:

1. System design will only consider the physical transmission layer of the standard. Within the physical layer, only those functions that affect transmission of data will be considered. Management and control functions of the physical layer will not be considered.
2. Carrier tracking- it will be assumed that ideal carrier phase information is obtained by the receiver to downconvert the received signal.
3. Chip and bit synchronization- it will be assumed that ideal chip and bit timing information is obtained by the receiver for chip and bit demodulation.
4. Data from the MAC layer will be modeled as random bits used for transmission.

2.1 IEEE 802.11 Compliant Specifications

The IEEE 802.11-1997 DSSS physical layer standard [6] specifies the functions required for transmission of data over the 1Mb/s and 2Mb/s DSSS physical layers. The standard specifies the scrambler, spreading code, modulation, and transmit mask along

with the control and management functions of the physical layer. These specifications are necessary to ensure that implementations by different vendors will interoperate with one another. A block diagram of a simplified IEEE 802.11 compliant 2Mb/s DQPSK transceiver is shown in figure 2.1. The 1Mb/s DBPSK transceiver uses only one of either the I or Q channels.

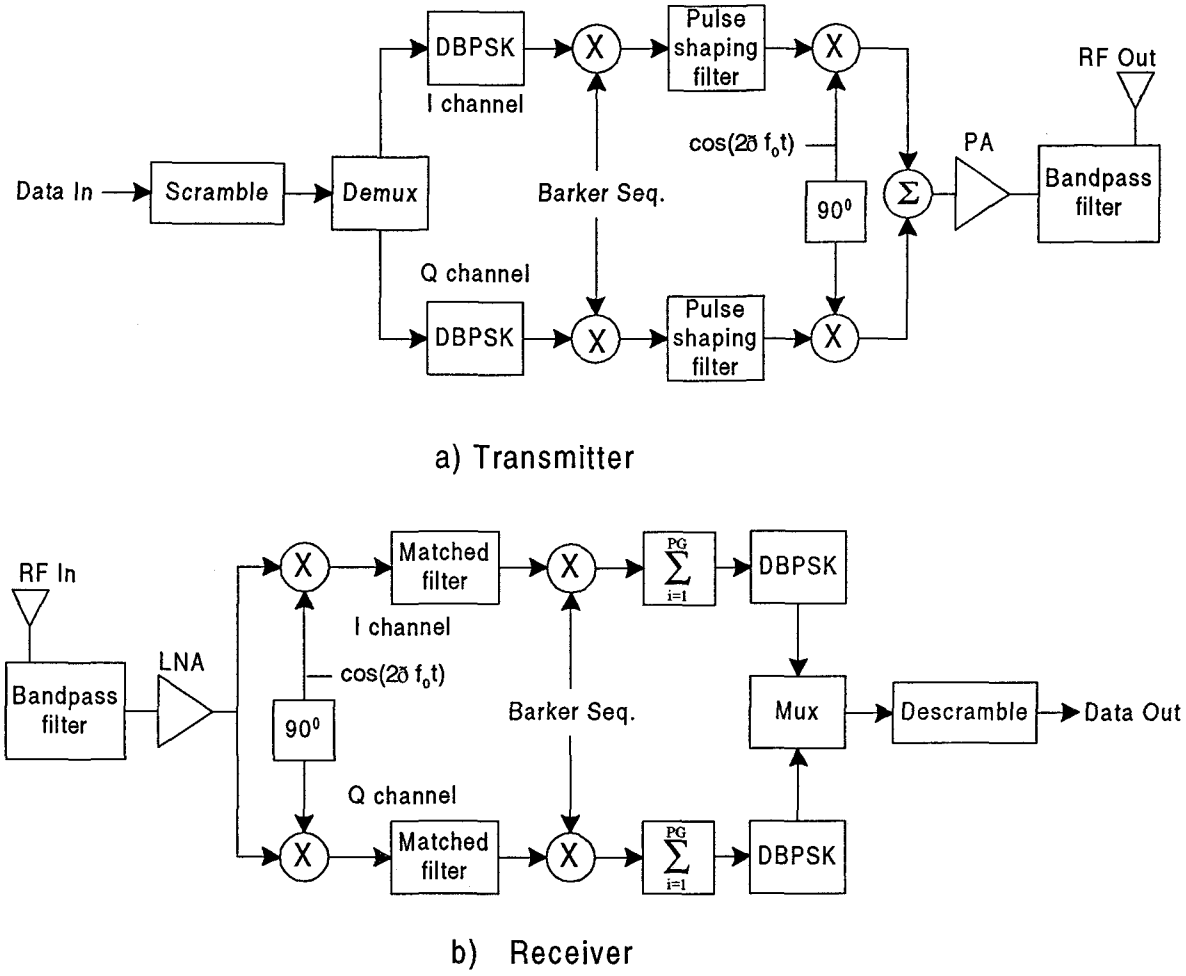


Figure 2.1 IEEE 802.11 physical layer transmitter and receiver

The scrambler/descrambler specified in the standard is used to prevent a CW transmission (which would violate the FCC regulations governing the ISM band) if a

long string of zeros or ones comes from the MAC. The transmitter scrambler specified in the standard is in the form of a infinite impulse response (IIR) filter described by the transfer function $G(z) = \frac{1}{z^{-7} + z^{-4} + 1}$ and difference equation $y(n) = x(n) \oplus y(n-4) \oplus y(n-7)$, where $\oplus$ represents modulo 2 addition which is equivalent to the XOR operation. The descrambler on the receiver side is specified by the inverse finite impulse response (FIR) filter given by $H(z) = z^{-7} + z^{-4} + 1$ and difference equation $y(n) = x(n) \oplus x(n-4) \oplus x(n-7)$. Figure 2.2 shows a typical implementation of the scrambler/descrambler using a shift register and XOR gates. Since the scrambler provides no redundancy or coding gain, the descrambler will extend each bit error made by the receiver to 3 bit errors. This causes some reduction in performance and as such requires a higher energy-per-bit to noise ratio (E_b/N_0) to maintain the same bit error rate (BER).

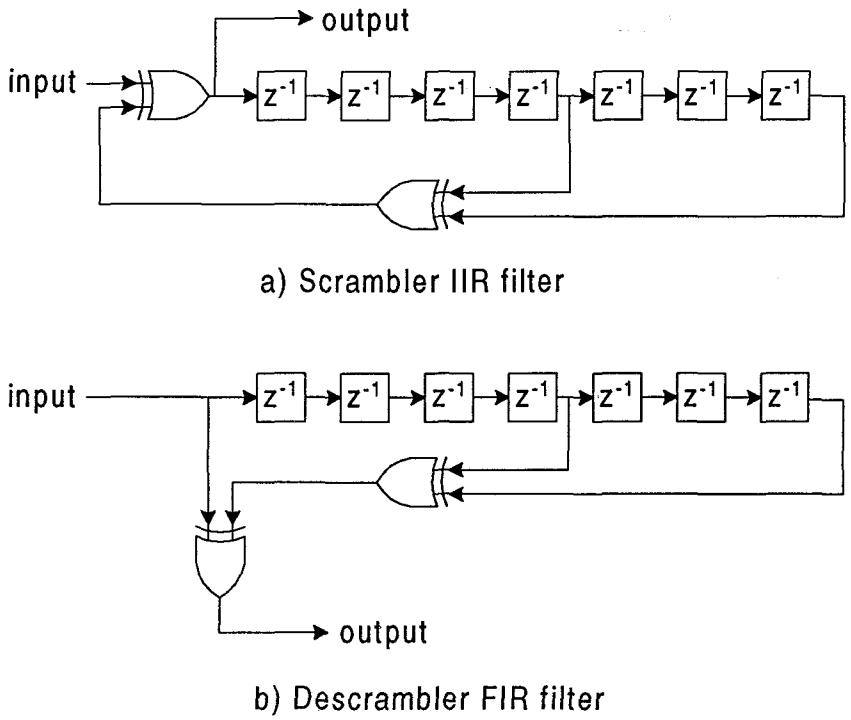


Figure 2.2 Typical shift register implementation

Each bit transmitted by the IEEE 802.11 DSSS physical layer is spread by the following 11 chip Barker PN sequence:

+1, -1, +1, +1, -1, +1, +1, +1, -1, -1, -1

The autocorrelation of this sequence is shown in figure 1.5a. The autocorrelation has a large main lobe and uniformly low side lobes which is important for good performance in systems that use such short codes.

The 1Mb/s IEEE 802.11 DSSS physical layer uses differential binary phase shift keying (DBPSK) modulation. The 2Mb/s physical layer uses two DBPSK modulators, one on the I channel and one on the Q channel, known as DQPSK. Figure 2.3 shows a block diagram and the resulting signal constellation of DQPSK modulation.

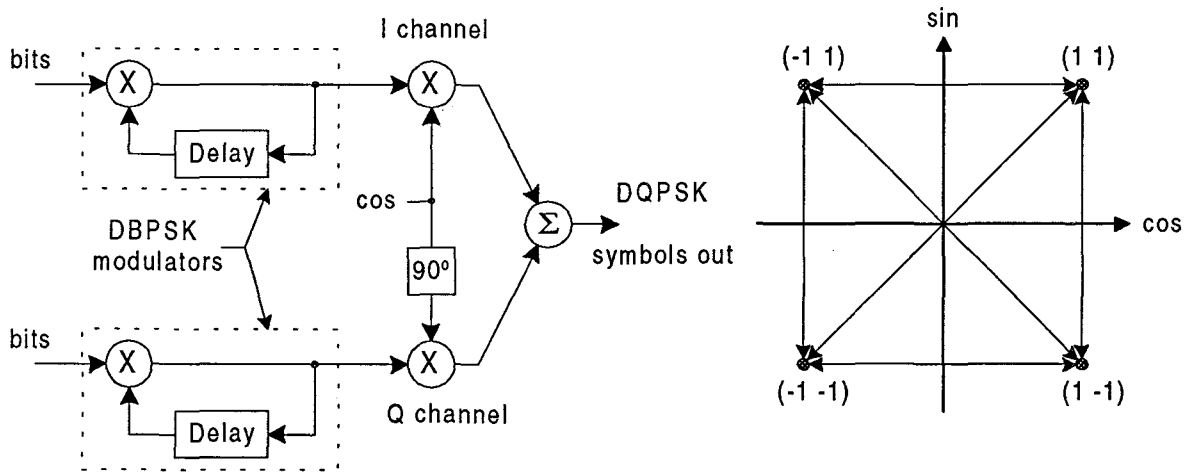


Figure 2.3 DQPSK modulation

The differential encoding is used to allow a simpler receiver design since an exact phase reference is not required to demodulate the symbols. Instead, the receiver looks for the characteristic 180° phase shift in each channel to detect symbol changes. Although this

modulation technique allows a simpler receiver design, it is at the cost of performance. Since DBPSK uses two consecutive chips to encode a symbol, if the receiver makes one symbol error, two chip errors will result in the decoding. This will have a slight effect on the BER performance of the receiver, and the combined degradation of the DBPSK modulation and the scrambler was shown in [11] to be about 1dB worse in E_b/N_0 performance than coherent BPSK without scrambling.

2.2 Pulse Shaping and Bandpass Filter

The IEEE 802.11 standard specifies a transmit mask for compliant devices which is shown in figure 2.4. This mask is used to reduce interference between DSSS physical layer cells that are operating on adjacent channel frequencies. Also shown in figure 2.4 is the $\sin x/x$ spectrum that results when the sinusoidal carrier is modulated with full width or non-return-to-zero (NRZ) rectangular pulses.

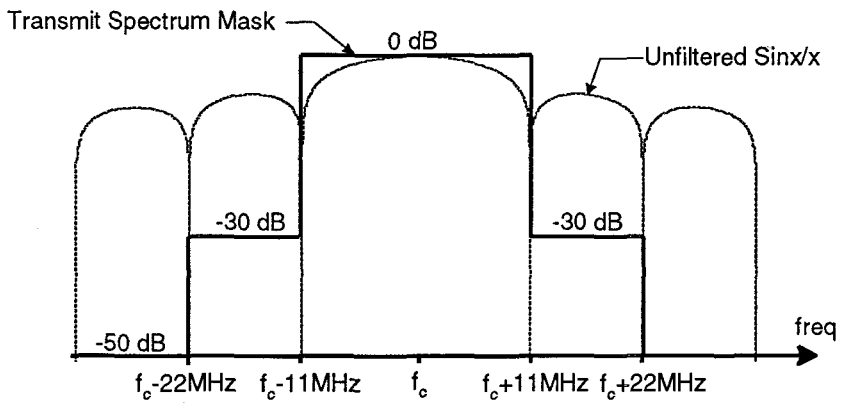


Figure 2.4 IEEE 802.11 DSSS physical layer transmit mask

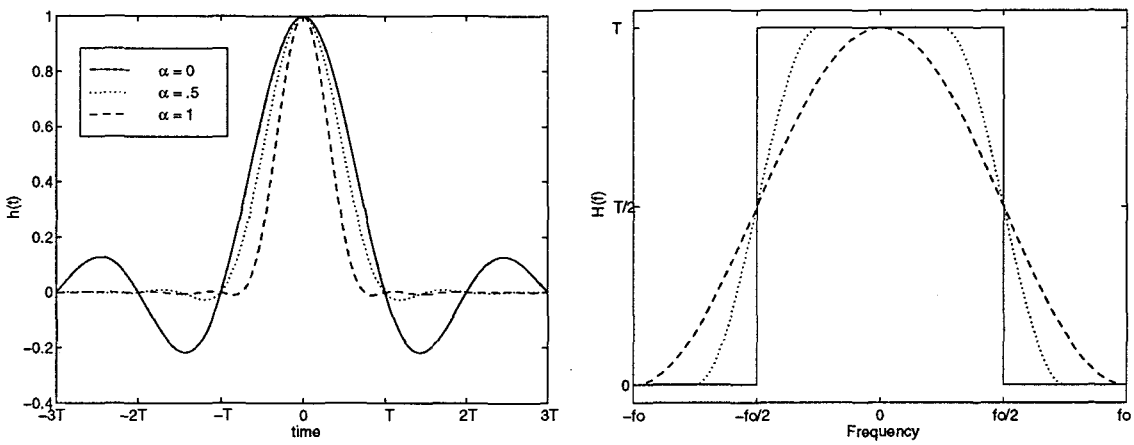
Since the mask is stricter than the $\sin x/x$ spectrum, some type of pulse shaping and/or bandpass filtering is required in the transmitter. Pulse shape filtering is often designed to combat intersymbol interference (ISI) that is commonly experienced in many environments. One such filter that satisfies Nyquist's criterion for zero ISI is called the raised cosine filter. This filter has an impulse response

$$h(t) = \left[\frac{\sin(\pi t/T)}{\pi t/T}\right]\cdot\left[\frac{\cos(\alpha\pi t/T)}{1-(2\alpha t/T)^2}\right] \tag{2.1}$$

and a Fourier transform

$$H(f) = \begin{cases} T & 0 \le |f| \le \dfrac{1-\alpha}{2T} \\ \dfrac{T}{2}\left\{1+\cos\left[\dfrac{\pi T}{\alpha}\left(|f|-\dfrac{1-\alpha}{2T}\right)\right]\right\} & \dfrac{1-\alpha}{2T} \le |f| \le \dfrac{1+\alpha}{2T} \\ 0 & |f| > \dfrac{1+\alpha}{2T} \end{cases} \tag{2.2}$$

where $0 \le \alpha \le 1$ is a bandwidth expansion factor and $T = 1/R$ where R is the data rate [10]. The impulse responses for a few values of α are shown in figure 2.5a and the resulting spectra are shown in figure 2.5b.



a) Impulse response b) Power spectral density

Figure 2.5 Raised cosine

For $\alpha = 0$, the raised cosine pulse becomes a sinc pulse, which is the minimum bandwidth pulse that satisfies Nyquist criterion for zero ISI. However, the sinc pulse has a slow time decay on the order of $1/t$ which is undesirable if the receiver has any timing jitter. So using $\alpha > 0$ expands the bandwidth and results in a quicker time decay on the order of $1/t^3$ to increase robustness against sampling offsets. Typically an $\alpha \in (.3, .5)$ is used. It is also noteworthy to recognize that since the raised cosine pulses are more bandwidth efficient than rectangular pulses, careful design can allow the data rate to by increased by up to 40% while still comforming to the IEEE 802.11 spectrum mask. This fact can not be used in the IEEE 802.11 compliant design, but will be used in the higher data rate system proposed.

For this design, a raised cosine filter with $\alpha = .5$ will be used. Since the raised cosine filter is non-causal and infinite in duration, careful consideration must be given to the implementation of this filter. If a delay is introduced, the truncated impulse response can be sufficiently represented using a tap-delay line or transversal filter (figure 2.6) [18].

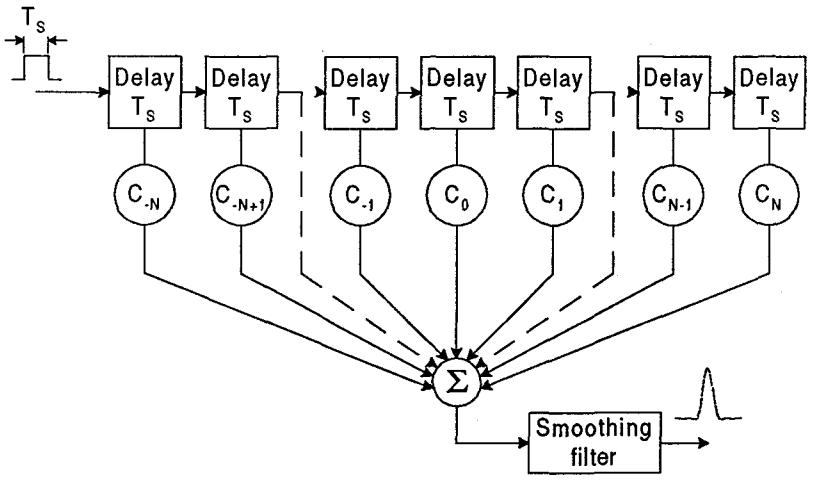


Figure 2.6 Tap delay line filter implementation

However, the truncation of the impulse response will generate side lobes in the spectrum of the filter. Thus, the truncation length must be large enough to reduce the sidelobes to a level so the pulse is compliant with the transmit mask. Figure 2.7 shows the spectrum of the impulse responses for different truncation lengths τ. From figure 2.7, a truncation length of 4T satisfies the spectrum mask, but 6T will be used to provide a margin of safety.

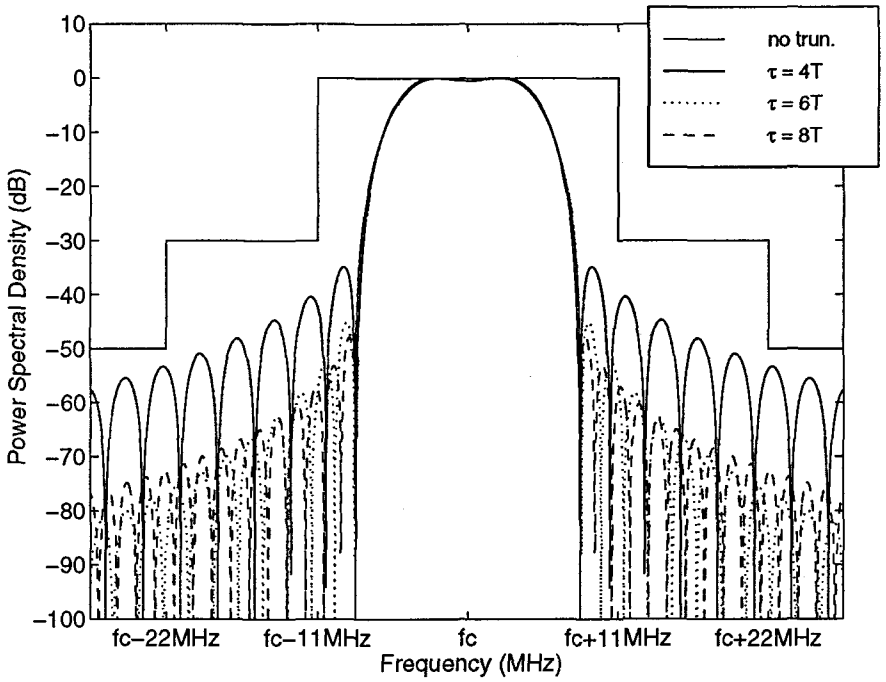


Figure 2.7 Sidelobes for various truncation lengths

The bandpass filtering shown in the transmitter diagram (figure 2.1a) is often necessary to due to the non-linearities of the power amplifier (PA) in the transmitter. This is an important consideration in the design of CDMA systems. Often, especially in battery powered devices, the power amplifiers are designed to operate close to the saturation region to maximize power efficiency. Because of this, a number of factors can cause the amplifier to be operated in saturation for short periods causing the production

of spectral products. In DSSS systems, these products show up as spectral regrowth at the band edges of the signal. These products, if large enough can spill over into adjacent bands causing interference. This necessitates the use of a bandpass filter to reduce the spectral regrowth to acceptable levels. Surface acoustic wave (SAW) filters are commonly used in these applications due to their size, high selectivity, and low cost. For simplicity, in this design it will be assumed that PA is operated in a linear region and so the bandpass filtering is not required. On the receiver side, the bandpass filtering is used for channel selectivity to remove the interference from other bands.

3. OCDM DESIGN FOR HIGHER DATA RATE

3.1 Introduction

The design of this higher data rate physical layer system is based on orthogonal code division multiplex (OCDM). OCDM is very similar to CDMA in that multiple orthogonal codes are used. However, OCDM assigns multiple codes to each user which allows increased data rates by simultaneous transmission over multiple links. CDMA on the other hand, assigns one code to each user to allows all users to access the channel at the same time. Since the IEEE 802.11 standard already defines the multiple access technique (CSMA/CA), CDMA is of little help because major modification to the standard would be required to use CDMA. But since OCDM uses multiple codes to increase data rate and does not allow multiple access, it does not require modification of the MAC standard. Also since one user transmits at a time due to CSMA/CA protocol, OCDM will not face some of the problems encountered in CDMA systems, like the near-far problem. This allows a simpler design over CDMA systems which is important in WLAN products that should have a low cost per unit to make them marketable.

OCDM has some advantages over other modulation schemes that have been proposed for higher data rate systems. First, since OCDM is a modification of a CDMA system, it benefits from the two decades of public research in development, simulation, and implementation of such systems. Although this thesis proposes a relatively simple receiver design, much of the recent work in high performance and efficient receivers for CDMA systems could be utilized in the future by modifying this design.

Second, OCDM allows a great deal of flexibility in network operation. Although these options will not be investigated in this thesis, it is worth briefly noting them. OCDM and CDMA systems are often interference-limited as opposed to noise-limited, which means that there is a soft relationship between number of links (data rate) and transmission quality. This allows flexibility in several respects. First, data rate capability can be exchanged for transmission quality. Although high quality of service is required for data transmission, it may be possible to sacrifice transmission quality for increased

data rate for time bounded services like voice traffic which tend to bog a network down. Because voice transmissions can usually operate at a higher BER than data, throughput of the network can be increased by assigning more links to stations with voice traffic. And, since only one user transmits at a time, the added interference will not affect the other users in the same cell. This is not the case with the other modulation schemes that have fixed data rates.

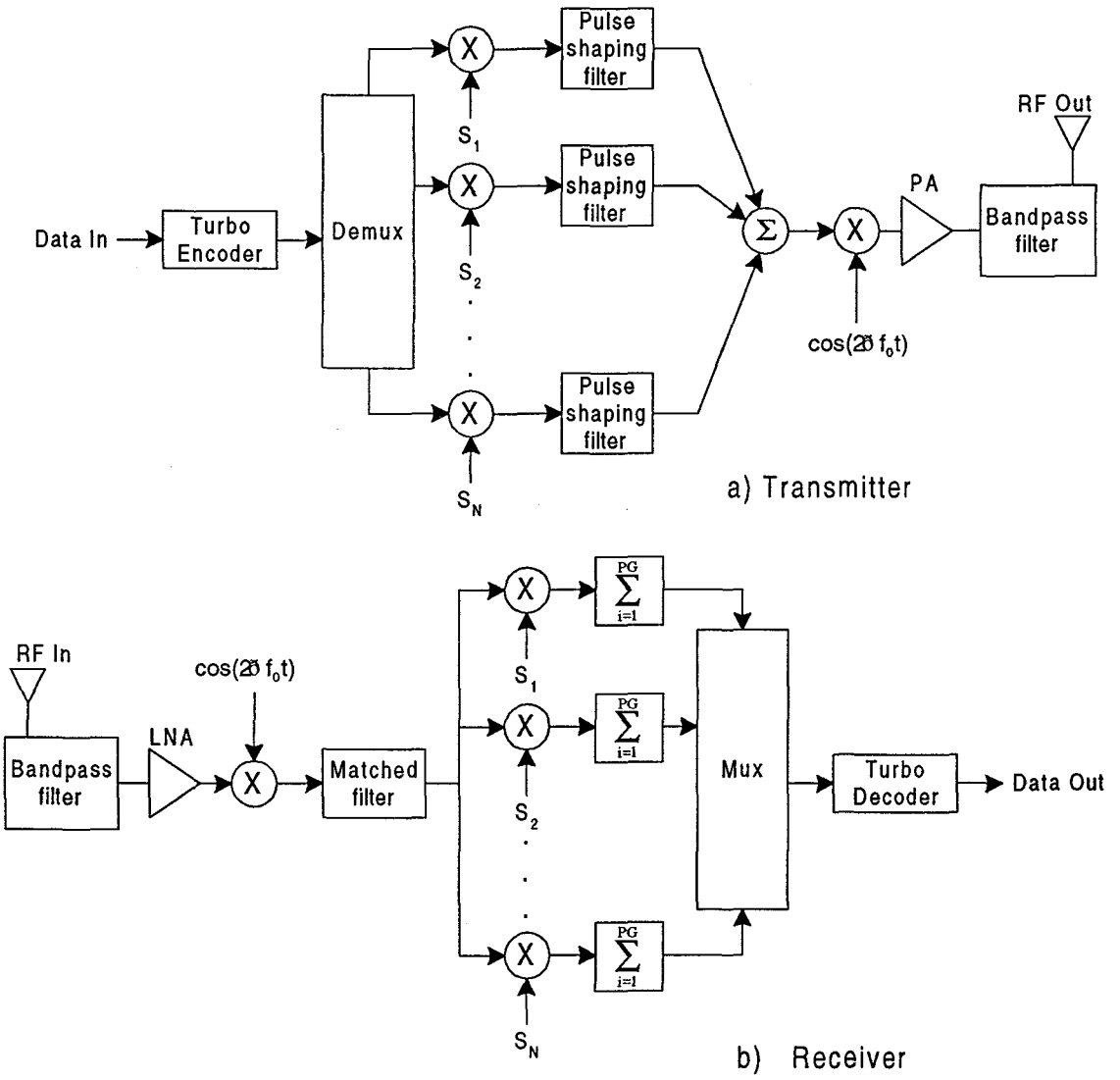


Figure 3.1 Proposed OCDM transceiver design

Second, a modification could allow an OCDM system to be used to increase the number of cells operating in a dense network environment. The IEEE 802.11 DSSS standard allows only 3 cells to operate in the same locale since only 3 separate DSSS frequencies are available in the relatively narrow 2.4 GHz ISM band. But since the orthogonal codes allow simultaneous transmission on the same frequency band, the cells could be assigned unique sets of codes to allow more than 3 cells in the same locale. This would increase the overall interference, but the data rate of each cell could be adjusted to maintain the interference at an acceptable level. It is clear that from these two examples, many possibilities in network operation exist due to the flexibility of OCDM.

Figure 3.1 shows the proposed BPSK transceiver design for the higher data rate system using OCDM. A QPSK design would include an identical copy of the figure 3.1 but modulated with a sine carrier.

3.2 Design of Spreading Codes

The choice of PN spreading codes involves many factors. First, the sequences should appear to be completely random. Second, each sequence should have very good autocorrelation properties. This means that the autocorrelation function of each sequence should exhibit a large central peak and uniformly low sidelobes. Third, the cross-correlation of the sequences should be as small as possible. Since cross-correlation is an indicator of the orthogonality of the codes, the lower the cross-correlation the better the orthogonality. This is a very important property because it allows the receiver to separate the links from one another and reduce interference. The first two characteristics can be formulated into the following set of requirements for binary m-sequences:

1. Balance property - in each sequence, the relative frequency of 0's and 1's are $1/2$.
2. Run property – $1/2$ of all run lengths (number of consecutive occurrences of 0's and 1's) are of length 1, $1/4$ are of length 2, $1/8$ are length 3, $1/2^n$ are of length n, where n is finite.

3. Two level autocorrelation function - $R(\tau) = \begin{cases} n & \text{if } \tau = 0 \\ -1 & \text{if } \tau \neq 0 \end{cases}$

Note that the above requirements do not give any indication of the cross-correlation properties of m-sequences.

A class of sequences that satisfies the above requirements are known as maximal length sequences. Maximal length sequences are sequences that are generated by an n^{th}-order primitive binary generating function

$$H(D) = c_0 + c_1 D + c_2 D^2 + \cdots + c_n D^n = \sum_{i=0}^{n} c_i D^i , \qquad (3.1)$$

where $c_i \in [0, 1]$ and D is the unit delay operator. Because the sequence is created with a generating function, it can be implemented with a shift register. A shift register with n memory elements has $N = 2^n - 1$ possible non-zero states, thus the sequence repeats with a period of $N = 2^n - 1$. Hence the autocorrelation function of the sequence is periodic. Much research has been done to find ways to use maximal length sequences to construct families of codes that exhibit good cross-correlation properties. Among the most well known are the Gold codes. Gold codes use two n^{th}-order primitive polynomials H_1 and H_2 to generate two maximal length sequences u_1 and u_2. The family of Gold codes is then defined by the set

$$G(u_1, u_2) = \left\{ u_1, u_2, u_1 \oplus u_2, u_1 \oplus Du_2, u_1 \oplus D^2 u_2, \cdots, u_1 \oplus D^{N-1} u_2 \right\}, \qquad (3.2)$$

where $\oplus$ is modulo 2 addition. Thus Gold codes have $2^{n-1} + 1$ sequences of length N. As n increases, the number of primitive polynomials of order n increases rapidly. This offers many possible choices of the two polynomials, however not all combinations of polynomials will result in Gold codes with good cross-correlation properties. In [19] it is shown that if a preferred pair of primitive polynomials is chosen, then all sequences in the resulting Gold code set will have a bounded 3-valued cross-correlation function. The preferred pair is chosen by the following procedure:

1. Choose a primitive polynomial $H_1(D)$ of order n (Most error correcting code textbooks have tables of these, i.e. [17]).
2. Determine the lowest order generating element of $H_1(D)$, α^a ([17])
3. Compute $b = (\alpha^a)^t \bmod 2^n - 1$ where t is given by

$$t = \begin{cases} 2^{(n+1)/2} + 1 & \text{for n odd} \\ 2^{(n+2)/2} + 1 & \text{for n even} \end{cases} \tag{3.3}$$

4. Find the primitive polynomial $H_2(D)$ with generating element α^b. $H_1(D)$ and $H_2(D)$ are the preferred primitive pair.

Let v_1 and v_2 be any two sequences in the Gold set. Given that $H_1(D)$ and $H_2(D)$ are a preferred primitive pair, then the cross-correlation $\theta(v_1, v_2)(\tau)$ is bounded by

$|\theta(v_1, v_2)(\tau)| \leq t$ for all τ, where t is given above.

For this design, Gold codes of length 512 ($n = 8$) were chosen. This length was chosen over a shorter length because of the autocorrelation-peak to cross-correlation bound ratio (ACR), which can be defined for Gold codes as

$$ACR = \frac{R(\tau)|_{\tau=0}}{t} = \frac{2^{n+1}}{t}, \tag{3.4}$$

where t is given above. Table 3.1 shows the ACR for different polynomial orders n. For shorter codes (i.e. 128 or 256), this ratio can be somewhat low which will increase interference and reduce the performance of the receiver. The larger this ratio, the more distinct the desired user or link is over the background interference.

n	$N = 2^{n+1}$	t	ACR
5	64	9	7.1
6	128	17	7.5
7	256	17	15.1
8	512	33	15.5
9	1024	33	31.0

Table 3.1 Autocorrelation peak to cross-correlation bound ratios (ACR)

The two preferred primitive polynomials chosen for this design are $H_1(D) = D^9 + D^8 + D^4 + D + 1$ and $H_2(D) = D^9 + D^8 + D^4 + D^2 + 1$ which result in the two recursive difference equations $u_1(n) = u_1(n-9) \oplus u_1(n-8) \oplus u_1(n-5) \oplus u_1(n-1)$ and $u_2(n) = u_2(n-9) \oplus u_2(n-7) \oplus u_2(n-5) \oplus u_2(n-1)$. The block diagram of the shift register implementation is shown in figure 3.2. The different Gold sequences are obtained by using different initial conditions in the shift registers.

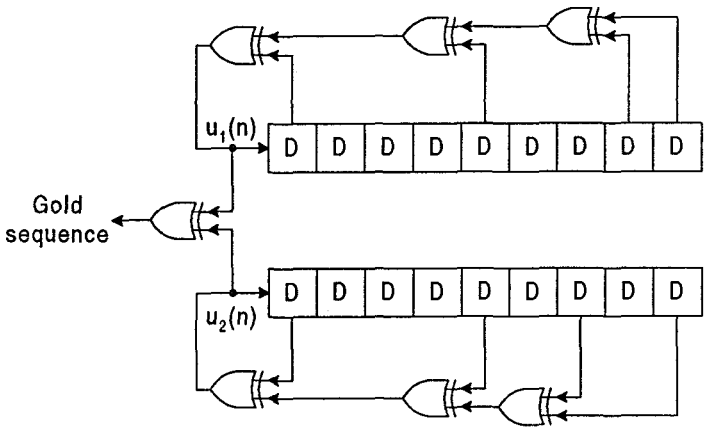


Figure 3.2 Gold sequence shift register implementation

Since the cross-correlation of any two sequences is bounded, the choice of the Gold codes can be chosen arbitrarily, and for this design the following Gold codes are used $G(u_1, u_2) = \{u_1, u_2, u_1 \oplus u_2, u_1 \oplus Du_2, u_1 \oplus D^2u_2, \cdots, u_1 \oplus D^{13}u_2\}$. A random search of Gold codes was performed to find combinations of Gold codes that provided the lowest peak to average power ratio. However, after extensive searching, it was found that improvements of only .2 dB could be realized, which are not very significant.

Coherent BPSK modulation is required when using OCDM since the differential phase information of DBPSK is lost when multiple links are summed together before upconversion by the carrier. This requires that the receiver generate an accurate phase reference for coherent demodulation. This is not a major increase in complexity since

phase lock loop (PLL) chips have become highly integrated and can easily be embedded in receiver design. Many of the other proposed modulation techniques discussed in the introduction also require coherent BPSK over DBPSK modulation. QPSK modulation is used in this design by adding an additional bank of correlators for the Q-channel in the receiver. This doubles the data rate without added interference, however twice as much transmit power is required.

3.3 Matched Filtering

The same raised cosine pulse shaping/matched filter with roll-off factor $\alpha = 0.5$ and truncation length $\tau = 6T$ is used in this higher rate design. Since the raised cosine filter is more bandwidth efficient than rectangular pulses, the data rate can be increased 40% with this chosen filter and still maintain compliance with the spectrum mask. The increased data rate will help offset the code rate of the turbo code that will be discussed in the next section. The PSD for this filter with the 40% increased data rate (1.4 Mb/s for uncoded BPSK) is shown in figure 3.3. From the figure it is clear that this system satisfies the spectrum mask requirement. Using this increased data rate might require the use of a stricter bandpass filter due to the non-linearities of the power amplifier, however, for this thesis, it is assumed that the PA is operated in the linear region.

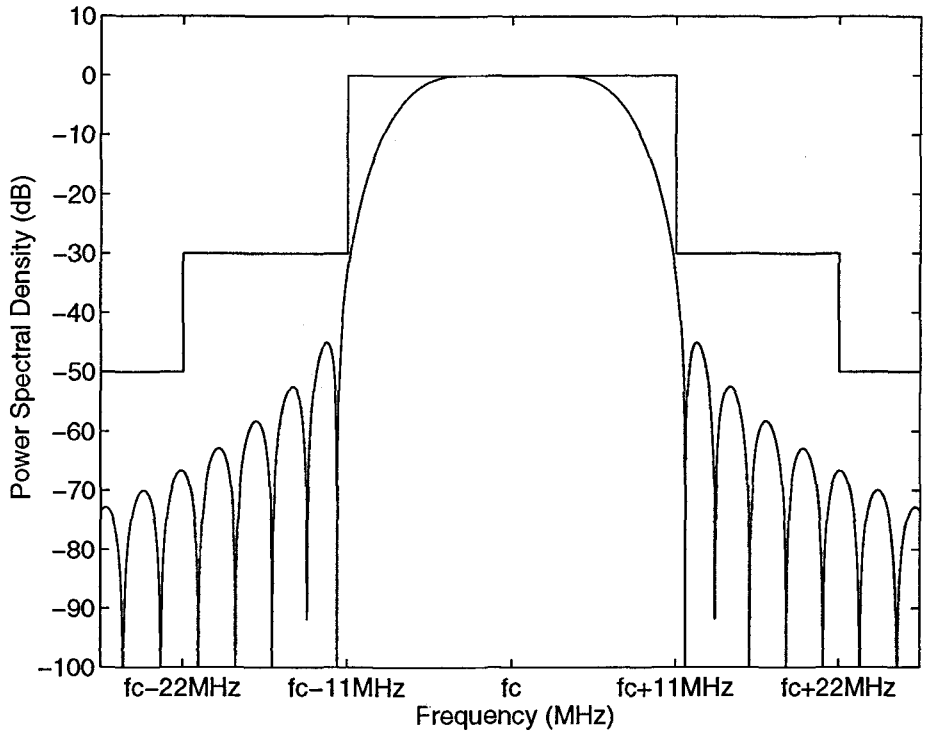


Figure 3.3 PSD of pulse shaping/matched filter for 40% increased data rate

3.4 Turbo Code Design

Before the design of the turbo code proposed in this thesis is presented, an analysis is necessary to justify the need for channel encoding for the OCDM higher rate design. As mentioned previously, the proposed OCDM system uses multiple, simultaneous, and equal power transmissions separated by unique codes. The reason channel encoding is required is that the processing gain of the IEEE 802.11 standard is too small to allow multiple transmissions of equal power to occur at the same time. This can be shown as follows. We first begin with the frame error rate (FER) specification of the IEEE 802.11 DSSS standard. The standard specifies a maximum FER of 8×10^{-2} or 8% for a packet length of 1048 bytes (8384 bits) including the header. Since the compliant physical layer contains no error correcting capability, all the bits in the packet

must be transmitted without error to avoid a frame error. Thus the FER can be expressed analytically as

$$\begin{aligned} FER &= P(\text{frame error}) = P(1\text{ bit error}) + P(2\text{ bit errors}) + \cdots + P(\text{all bits in error}) \\ &= 1 - P(\text{no bit errors}) = 1 - (1 - BER)^N \end{aligned}$$

were N is the length of the packet in bits. Solving the above equation for the BER gives

$$BER = 1 - \sqrt[N]{1 - FER} \tag{3.5}$$

and for $FER = 8 \times 10^{-2}$ and $N = 8384$, we obtain a minimum BER requirement of

$$BER_{min} = 1 - \sqrt[8384]{1 - .08} = 1 \times 10^{-5}.$$

Now, for BPSK modulation it is well known that the theoretical BER performance for the optimum receiver in an AWGN environment is given by

$$P_e = BER = Q\left(\sqrt{\frac{2E_b}{N_0}}\right), \tag{3.6}$$

where $Q(x) = \int_x^\infty \frac{e^{-u^2/2}}{\sqrt{2\pi}} du$. A plot of the BER vs. E_b/N_0 is shown in figure 3.4. From figure 3.4, we see that a minimum E_b/N_0 of 10 dB is required to obtain a BER of 10^{-5}.

For a spread spectrum system, the input signal to noise ratio (SNR) is given by

$$\frac{S}{N_0} = \frac{E_b}{N_0} \cdot \frac{1}{PG}. \tag{3.7}$$

For a signal cell OCDM system with M links of equal power and assuming no interference from neighboring cells, the input SNR can be shown to be related to the number of links M by

$$\frac{S}{N_0} = \frac{1}{M-1}. \tag{3.8}$$

This is the same capacity (number of users) formula as a cellular CDMA system with perfect power control. And, indeed the two situations are equivalent because perfect power control is achieved in the proposed OCDM system by allowing one user to transmit at a time and by transmitting over each link with equal power. Combining equations 3.7 and 3.8, the capacity of the uncoded OCDM system at a desired FER is given by

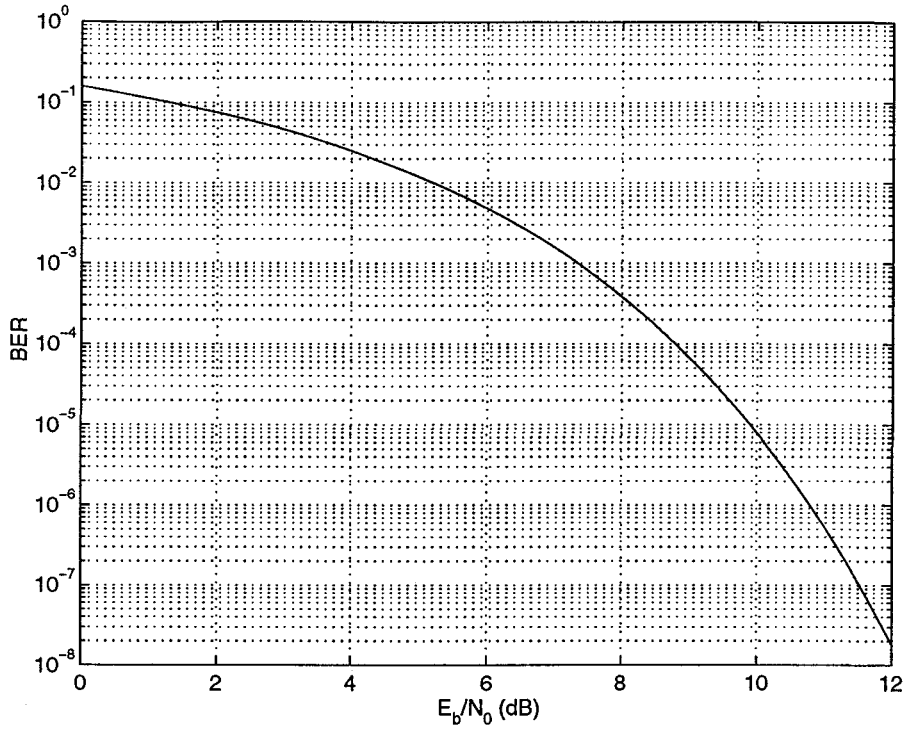


Figure 3.4 BER vs. E_b/N_0 for BPSK modulation

$$M = 1 + \frac{PG}{\frac{E_b}{N_0}\min}. \tag{3.9}$$

Thus, for the IEEE 802.11 DSSS compliant physical layer, the capacity is

$$M = 1 + \frac{11}{10} = 2.1.$$

The above result shows that, theoretically, the uncoded system could support two links at the maximum allowable FER. However, in reality when interference from other cells and interchip interference are considered, it will be difficult to support two links and maintain the required performance. Therefore, two solutions exist to increase the number of links: increase the processing gain of the system or use an error correction code. Increasing the processing gain is undesirable because it will increase the transmission bandwidth and further limit the number of DS cells that can operate in the ISM band. Error correction coding on the other hand will not change the transmission bandwidth as

long as the data rate of each link is reduced by the code rate. This at first glance appears to contradict the goal of increasing the data rate. However, an old addage says that sometimes you need to make things worse before you can make them better. This is the case for error correction coding, in that the added coding gain will allow the use of several links which will make up for the data rate reduction of one link and improve the overall data rate. Typical coding gains for turbo codes range $8-10\,\text{dB}$ $(6.3-10)$ over an uncoded system. By including the coding gain factor in the capacity formula, an estimate of the number of links in the turbo coded system can be obtained by

$$M = 1 + \frac{PG \cdot CG}{\frac{E_b}{N_0}\min} \tag{3.10}$$

For the given system parameters, the coded capacity is

$$M = 1 + \frac{11 \cdot 6.3}{10} = 7.9 \text{ for } CG = 8\,\text{dB} \text{ and } M = 1 + \frac{11 \cdot 10}{10} = 12 \text{ for } CG = 10\,\text{dB}.$$

So, if the code rate is $1/2$, then the overall data rate for the coded system will be 4 to 6 times greater than the uncoded data rate for the same modulation. However, these calculations do not include other cell interference, interchip interference (due to non-zero cross-correlation between sequences), or implementation losses. Thus, the actual number of attainable links will likely be smaller. But, nonetheless, it appears that a turbo code can provide for a significant increase in the data rate of a IEEE 802.11 WLAN system.

A block diagram of a turbo encoder is shown in figure 3.5. The basic idea of a turbo code is to combine two relatively simple and individually weak convolutional codes in such a way as to make a much stronger code. Since the complexity of the Viterbi decoding grows exponentially with an increase in constraint length, the decoding for the two shorter codes is easier than the decoding for the longer code to obtain the same performance. This makes the turbo code attractive from an implementation point of view.

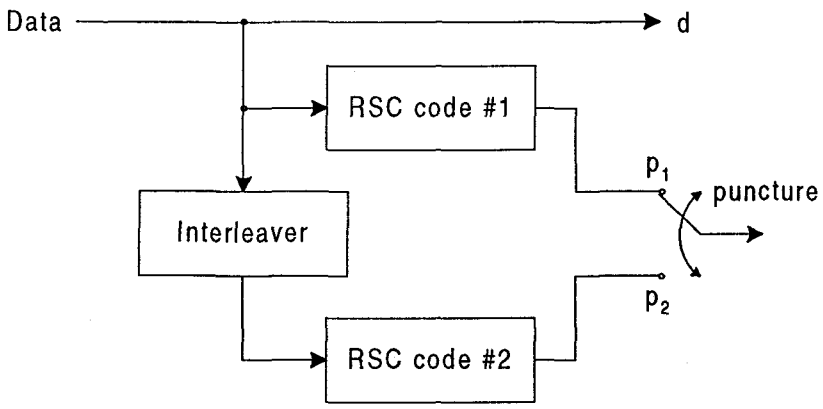


Figure 3.5 Turbo encoder

The structure of turbo codes differ from conventional convolutional codes in several ways. First, the turbo code uses two recursive systematic convolutional (RSC) codes unlike the conventional non-systematic convolutional (NSC) codes that are normally used. A systematic code is used because it allows two parallel convolutional codes to be easily combined. Also, the original input data is only sent once thereby making the system more efficient. A recursive code is one in which some of the states are fed back to the input of the encoder. This results in a structure that has an infinite impulse response (IIR), meaning that if an input to the encoder contains a single '1', then an infinite number of '1's will be produced at the output. A block diagram of a RSC code is shown in figure 3.6. Note that the structure of this code is similar to that of the NSC code in figure 1.7, in fact the two codes are equivalent. The generating polynomials of these two codes are the same, however, the formulation of the a(n) and y(n) are different. From figure 3.6, we can determine that $y(n) = a(n) \oplus a(n-2)$ and $a(n) = x(n) \oplus a(n-1) \oplus a(n-2)$. In general, y(n) and a(n) are given by

$$y(n) = \sum_{\tau=0}^{m} a(n-\tau) \cdot g_1(\tau) \bmod 2 \qquad (3.11)$$

and

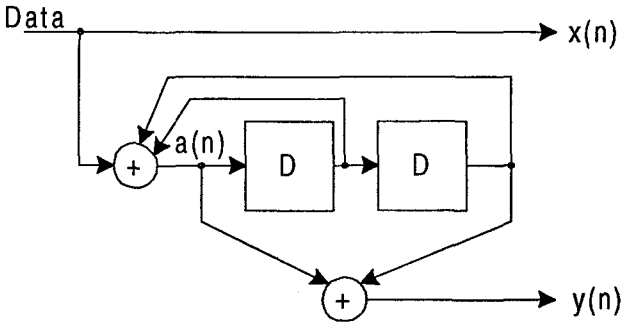


Figure 3.6 Rate 1/2 RSC code

$$a(n) = \sum_{\tau=1}^{m} a(n-\tau) \cdot g_2(\tau+1) \bmod 2 \qquad (3.12)$$

With this formulation it can be verified by substitution that the RSC and NSC codes of figure 1.7 and 3.6, respectively, indeed have the same generating polynomials. The two codes also have the same state diagram and same trellis, but a given output sequence does not correspond to the same input sequence for both codes. This indicates that the two codes map inputs to outputs differently. It is well known that most RSC codes outperform NSC codes at low E_b/N_0 values. Also, RSC codes tend to perform better than NSC codes at higher code rates. Since turbo codes are designed to obtain large coding gains at low E_b/N_0 values and can be punctured to increase the code rate, the use of RSC codes is an obvious choice [20, 21].

An extensive analysis of turbo codes was performed in [21]. Berrou concluded that for a rate 1/2 (with puncturing) turbo code, RSC code generator polynomials $g_1 = 37_8$ and $g_2 = 21_8$ (constraint length 5) resulted in good performance. Thus, the $(37_8, 21_8)$ turbo code is used for this design, but the interleaver will be different than that proposed by Berrrou to contend with the variable packet length of the IEEE 802.11 system.

Puncturing of the concatenated code is used to improve the code rate. Not all RSC codes are good candidates for puncturing and a list of good punctured codes is given

in [17]. The decoding of a punctured code is done by inserting erasures (spaces with undetermined value) and using the decoding algorithm to determine the deleted bits. Not only can puncturing improve the code rate, but it can also offer a decrease in decoding complexity over an equivalent non-punctured code. The code selected for this design is puncture compatible and increases the code rate from 1/3 to 1/2.

The interleaver used between the two RSC codes is used for several reasons. First, an interleaver in the encoder helps to maximize the weight distribution of codewords. This means that if an input sequence to the first encoder results in a low weight output, then by interleaving the input before the second encoder, it is highly unlikely that the resulting output from this encoder will be low weight as well. By maximizing the weight distribution of the codewords, the free distance d_{free} of the code is increased and in turn improves the error correcting capability of the code [20]. Second, the interleaver combats the effect of burst errors at the decoder. If a burst of errors occurs in close succession, the first decoder will not likely be able to correct all of them. However, if the interleaver distributes the burst errors over the entire codeword, then the second decoder has a much better probability of correcting the errors. Thus it can be stated that with the interleaver between the two decoders in a turbo code, it is more unlikely that a received sequence will cause a decoder failure in both decoders.

Interleaver design for turbo codes is currently an active area of research. At the moment, it is believed that pseudo-random interleaving provides better performance over more traditional block interleaver [20]. Block interleaving also presents somewhat of a problem for the IEEE 802.11 system because of the variable packet length. Therefore, a variable length pseudo-random interleaver is used in this design. Due to the lack of research on pseudo-random interleavers, it is somewhat difficult to ascertain exactly how the interleaver should be generated to obtain the best performance for a given turbo code. Thus, this design randomly generates a new interleaver for every packet. Therefore, since some of the generated interleavers will likely be poor, while others will likely be good, this design should represent the turbo code performance for an average interleaver. Implementation of this type of interleaver in hardware maybe a concern, however, if the baseband processing is done in a high speed DSP processor (which appears to be the case with most current IEEE 802.11 products), then the pseudo-random interleaver design can

be accomplished in firmware. Also, future research in this area will most certainly yield pseudo-random interleavers with good properties that are easier to implement.

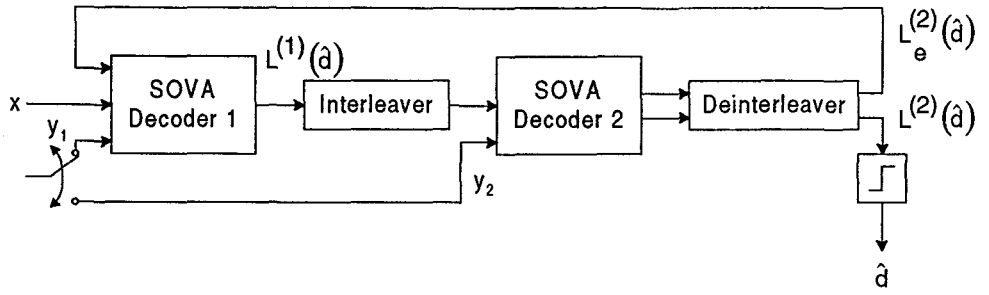


Figure 3.7 Turbo decoder

Turbo codes use an iterative decoding procedure. Figure 3.7 shows the block diagram of the turbo decoding structure. The iterative decoding procedure is based on the soft-decision output Viterbi algorithm (SOVA). A soft decision means that instead of assigning a hard decision '1' or '0' to each bit, an estimate is produced along with a measure of confidence of that estimate. This can be illustrated with figure 3.8 which shows the received signal probability density function (PDF) for a BPSK modulated signal corrupted by AWGN. In hard decision, the bit value is determined by the threshold at $x = 0$. In soft decision the x-axis is divided into regions and assigned numbers that represent the confidence of the received value from a strong '1' to a strong '0'. Soft decision algorithms usually can provide several decibels (dB) of additional coding gain in channels with AWGN [10, 17].

The traditional Viterbi decoding algorithm accepts hard decision inputs and provides hard decision outputs. In the years since its introduction, efficient methods have been developed to modify the VA to accept soft decision inputs and provide soft decision

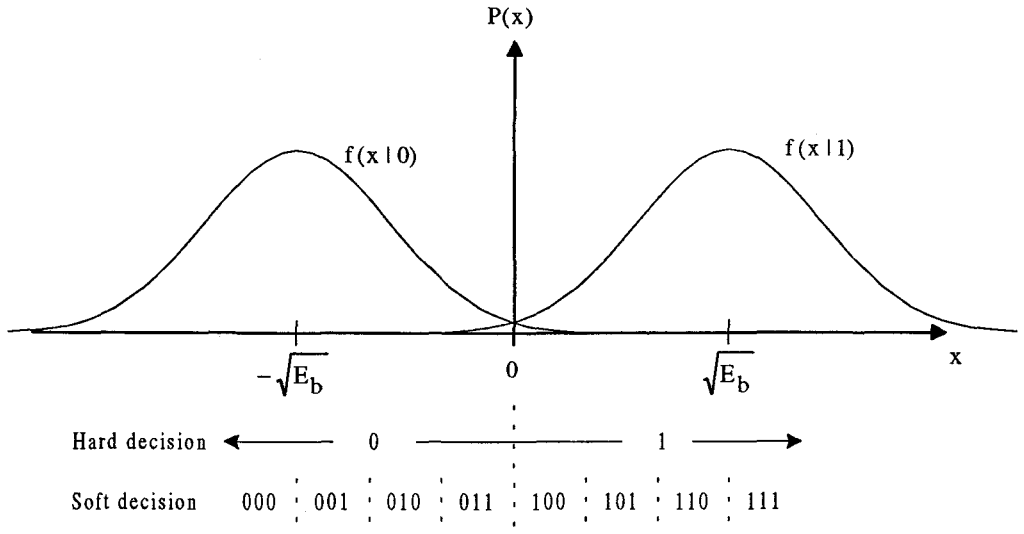


Figure 3.8 Hard and soft decision for BPSK

outputs [22-24]. The Hagenauer algorithm is used in the decoding for this thesis and is discussed later.

The basic idea of the iterative decoding is to use information gained through decoding to improve the performance of the next decoding cycle. As the number of iterations increases, this information is refined again and again, and if the decoder is designed properly, it will result in better performance. This is in essence an algorithm that searches through the trellis over and over to find the most probable path. If the traditional VA with hard decisions is used for the component decoders, not much is gained by using iterative decoding [20]. This is why the SOVA algorithm is used in the iterative turbo decoder, because it provides large improvements in BER.

The following analysis will briefly describe how the decoding process works and discuss some implementation issues. For a binary channel that is corrupted by noise, the received signal probability P(x) (figure 3.8) is given by

$$P(x) = P(x \mid d = 1) \cdot P(d = 1) + P(x \mid d = -1) \cdot P(d = -1) \tag{3.13}$$

where the conditional probabilities $P(x \mid d = i)$ and $P(d = i \mid x)$ are related by Bayes' theorem, i.e.,

$$P(d=i \mid x)=\frac{P(x \mid d=i)\cdot P(d=i)}{P(x)} \quad \text{for } i=1,-1. \tag{3.14}$$

The Viterbi algorithm in the decoder for this thesis uses the maximum a posteriori (MAP) decision rule which states that one should choose

$$\hat{d}=\begin{cases}+1 & \text{if} \quad P(d=+1 \mid x)>P(d=-1 \mid x) \\ \text{or} & \\ -1 & \text{if} \quad P(d=-1 \mid x)>P(d=+1 \mid x)\end{cases} \tag{3.15}$$

where $\hat{d}$ is the decoded data bit value. The above equation can be expressed as a ratio known as the likelihood function

$$\hat{d}'=\frac{P(d=+1 \mid x)}{P(d=-1 \mid x)} \quad \text{and by substitution of Bayes' Theorem, this becomes}$$

$$\hat{d}'=\frac{P(x \mid d=+1)\cdot P(d=+1)}{P(x \mid d=-1)\cdot P(d=-1)} \tag{3.16}$$

Now by taking the logarithm of both sides of the above equation, we obtain a useful metric for the decoding process called the log-likelihood ratio (LLR)

$$L'(\hat{d})=\log\left[\frac{P(x \mid d=+1)\cdot P(d=+1)}{P(x \mid d=-1)\cdot P(d=-1)}\right]=\log\left[\frac{P(x \mid d=+1)}{P(x \mid d=-1)}\right]+\log\left[\frac{P(d=+1)}{P(d=-1)}\right] \tag{3.17}$$

which can be written as

$$L'(\hat{d})=L_c(x)+L(d) \tag{3.18}$$

where $L_c(x)$ is the LLR term from the soft-decision channel measurements of the detector and $L(d)$ is the LLR of the a priori information from the data bits. Therefore, $L'(\hat{d})$ represents the LLR of a soft-decision detector. Note, that by taking the sign of $L'(\hat{d})$, the hard decision bit estimates $\hat{d}$ (± 1) are obtained. Thus, the sign of $L'(\hat{d})$ determines the bit estimate and the magnitude of $L'(\hat{d})$ is the confidence measurement of the estimate [20].

It can be shown [25] that for a systematic code, the LLR of the output of a soft-decision decoder is given by

$$L(\hat{d})=L'(\hat{d})+L_e(\hat{d}) \tag{3.19}$$

where $L'(\hat{d})$ is given in equation 3.18 and $L_e(\hat{d})$ is the extrinsic LLR of the decoder which represents the extra knowledge that has been obtained by the decoding process. Substituting in equation 3.18, we obtain

$$L(\hat{d}) = L_c(x) + L(d) + L_e(\hat{d}) \tag{3.20}$$

Thus, using this relationship, the LLR outputs of the two decoders in the turbo code in figure 3.7 can be expressed as

$$L^{(1)}(\hat{d}) = L_c(x) + L(d) + L_e^{(1)}(\hat{d}) \tag{3.21}$$

and $$L^{(2)}(\hat{d}) = L_e^{(2)}(\hat{d}) + L_e^{(1)}(\hat{d}) + 2L_c(x) + L(d) \tag{3.22}$$

where the notations $^{(1)}$ and $^{(2)}$ represent the first and second decoders respectively [20]. From the second decoder equation, we can see that LLR output of the detector measurement is over emphasized, and should be subtracted out by the second decoder. Also, given the assumption in this thesis that the data bits are random (i.e. $P(d=+1) = P(d=-1) = 1/2$), then $L(d) = 0$. With these two conditions the LLR (soft-decision output) of the second decoder becomes

$$L^{(2)}(\hat{d}) = L_e^{(2)}(\hat{d}) + L_e^{(1)}(\hat{d}) + L_c(x) \tag{3.23}$$

The LLR output $L_e^{(2)}(\hat{d})$ of the second decoder is fed back in figure 3.7 to be used in the next cycle of decoding. The iterations continue until the desired performance is achieved, at which time the data bit estimates $\hat{d}$ are obtained by hard decision of $L^{(2)}(\hat{d})$ [20, 22].

The algorithm used in this thesis to determine the extrinsic LLR values $L_e(\hat{d})$ is taken from [22]. It uses a much simplified computation of the extrinsic LLR metric which does not require complex calculations or knowledge of the input SNR. The derivation of this algorithm is somewhat long and will not be presented here, however the steps in the algorithm are outlined below.

Recursion:

a) Classical Viterbi algorithm

For each state in the trellis at time k (s_k):

1. Compute the soft decision path metric

   $M^{(i)}(s_k) = M^{(i)}(s_{k-1}) + M^{(i)}(s_k, s_{k-1})$ for both transitions $i = 1, 2$.

2. Find the minimum of the possible path metrics

   $M(s_k) = \min[M^{(i)}(s_k)]$.

3. Store the minimum path metric $M(s_k)$ and the corresponding surviving bit estimate $\hat{b}_k(s_k)$.

b) Soft-decision computation

For each state s_k:

1. Compute and store the difference between the two path metrics, i.e. $\Delta = \max[M^{(i)}(s_k)] - \min[M^{(i)}(s_k)]$.

2. Initialize the extrinsic LLR $\hat{L}_k(s_k) = +\infty$.

3. For $j = k - m$ to $j = k - \delta$

   Compare the two paths merging at s_k.

   If $\hat{b}_j^{(1)}(s_j) \neq \hat{b}_j^{(2)}(s_j)$, then update $\hat{L}_j = \min[\hat{L}_j, \Delta]$

where m is the number of states and δ is the window length, which should be chosen to be 4-5 times the constraint length of the code.

Berrou in [23] develops a low complexity architecture for the algorithm presented above. The architecture results in a soft-output Viterbi decoder that is approximately twice as complex over a conventional decoder. However, the structure does not reduce the speed of the decoder, and hence decoding speeds are as fast as traditional Viterbi decoders. Decoding speed is an important consideration for the iterative decoder that must complete multiple decoding cycles. The decoder for this design will be required to run several times faster than a traditional Viterbi decoder to maintain a decoding delay that is acceptable.

4. SIMULATION

In this chapter, the simulation setup and the simulation results will be presented. The first section will deal with the simulation setup, including the assumptions made in performing the simulations. The second section will present the results obtained from simulations of the IEEE 802.11 compliant 1 and 2 Mb/s physical layers. The third section will present the simulation results from the higher data rate design.

4.1 Simulation Setup

All simulations performed to verify the design proposed in this thesis were conducted using Matlab. Matlab's flexible structure that allows external m-files and functions to be created makes it an excellent simulation tool for this thesis. Also, Matlab's ability to interface with user written compiled C code (MEX functions) proved extremely useful for the simulations of this thesis.

The simulations performed in this thesis were packet-based simulations of the physical layer. As such, the management and control functions of the physical and MAC layers were not considered. The simulation results presented show only the performance of the physical layer and do not consider network performance (such as throughput) which is determined by the number of system users and the MAC functions. Thus, the simulation results of this thesis show the radio performance that each user can expect when allocated transmission access to the channel.

To simplify the simulations to a reasonable level so useful and timely results can be obtained, several assumptions are used in the development of the simulation code:

1. Simulations are performed at baseband. No upconverting or downconverting is performed due to bandwidth considerations. Also, perfect carrier phase knowledge at the receiver is assumed so phase lock loop (PLL) simulation is not required.

2. Perfect chip synchronization is assumed, alleviating the need for a chip tracking algorithm.
3. All bits of the packet are randomly generated so that $P('0') = P('1') = 1/2$.
4. The packet preamble and header information are not generated. This is valid since perfect chip timing is assumed and management and control functions are not considered.
5. The channel is modeled as additive white Gaussian noise (AWGN). Multipath and interference from neighboring cells is not considered.

The simulation code was developed with modularity in mind, meaning that the code is structured so that functions can be added or deleted with relative ease and minimum modification. Two separate simulation files are used, one for the IEEE 802.11 compliant physical layer and one for the proposed physical layer. Many of the tasks in these files are completed with supporting sub-functions. Enable switches are also used to turn on and off many of the functions, allowing flexibility in evaluating system performance.

4.2 Simulation Results for Compliant Design

The results of the simulation for the compliant physical layer come in the form of plots of the probability of bit error P_{be} and the probability of frame error P_{fe} vs. the energy per bit to noise density ratio E_b / N_0. The simulation counts the number of bit errors and frame errors and divides by the total number of bits and frames to obtain the BER and FER. The BER and FER are frequency estimates of the true probabilities P_{be} and P_{fe} respectively. The frequency interpretation of probability theory for discrete random variables states that as the number of occurrences of an event A increase, the relative frequency of that event converge to the actual probability P(A) [26]. This can be stated mathematically as

$$P(A) = \lim_{N \to \infty} \frac{Num(A)}{N} \tag{4.1}$$

where Num(A) is the number of occurrences of the event A.

The question then becomes how many occurrences are necessary to obtain a good measure of the actual probability, which are known as confidence intervals. Since the confidence interval is affected by the physical layer functions (filtering, coding, etc.), it is difficult to determine this exactly. For errors that are independent and identically distributed, a good rule of thumb is suggested in [27]. The situation applies to these simulations as the received distortion is AWGN. This rule of thumb states that for a 99% confidence interval, that after 10 errors the BER will be within a factor of 2 (high or low) of P_{be}, and after 100 errors the BER will be within a factor of $1.3P_{be}$.

For the simulations in this thesis, many bit errors are generated at low E_b / N_0 values, and very good measurements are obtained. However, at larger E_b / N_0 values, many bits are required to obtain one error, which increases simulation time exponentially. Therefore, due to limited computational resources, the probabilities P_{be} and P_{fe} are estimated with approximately 10 error occurrences at high E_b / N_0 values.

Figure 4.1 shows the simulation results for the IEEE 802.11 DSSS compliant physical layer for the 1Mb/s standard. The curves in figure 4.1 show the BER vs. E_b / N_0 performance using BPSK modulation, DBPSK modulation, BPSK modulation and scrambling, and DBPSK and scrambling. The data markers indicate the simulated results and the lines the theoretical performance. These curves illustrate the performance degradation of the compliant physical layer functions specified in the standard.

The DBPSK curve shows that the non-coherent DBPSK modulation results in a BER that is twice that of coherent BPSK. This is consistent with the analysis of chapter 3, since the phase of the carrier is switched by 180° on each occurrence of a '0' in the bit stream. Thus, a channel error will cause two demodulation errors in DBPSK, doubling the BER. The simulation results match the theoretical performance almost exactly.

The scrambling operation to whiten the transmitted spectrum causes 3 descrambled bit errors for every 1 demodulated bit error at high E_b / N_0 values, resulting in a BER 3 times that of the BPSK system without scrambling. However, the simulations

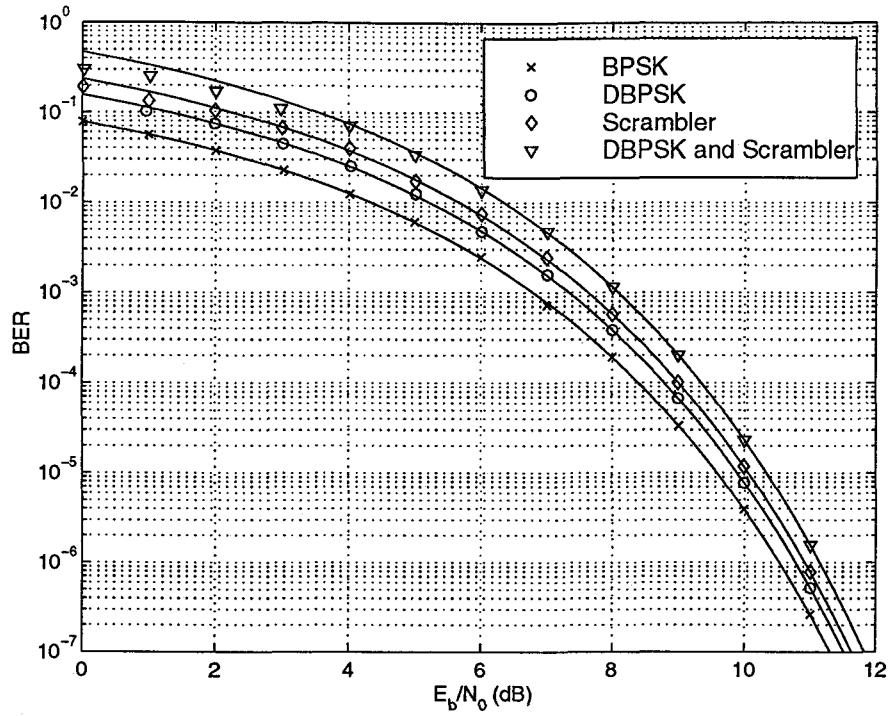


Figure 4.1 IEEE 802.11 compliant link BER performance

show that the scrambler BER performance is slightly better than the theoretical at low E_b / N_0 values. This can be explained by the descrambling filter structure. At low E_b / N_0 values, more channels errors are generated and the probability of multiple errors within the memory length of the FIR descrambling filter is increased. For certain multiple errors patterns, the rule that 3 errors are generated for each bit error does not hold, and 4-5 errors may be generated by certain patterns of two errors. This results in the slightly better BER performance.

The IEEE 802.11 DSSS compliant physical layer is the combination of DBPSK modulation and the scrambling/descrambling. Thus, the compliant BER performance is 6 times worse than BPSK without scrambling. This results in a performance reduction of about 1 dB for a BER of 10^{-5}. Note that the theoretical performance of the compliant layer is slightly worse than the simulated result at low E_b / N_0 values. This difference is again contributed by the FIR descrambling filter.

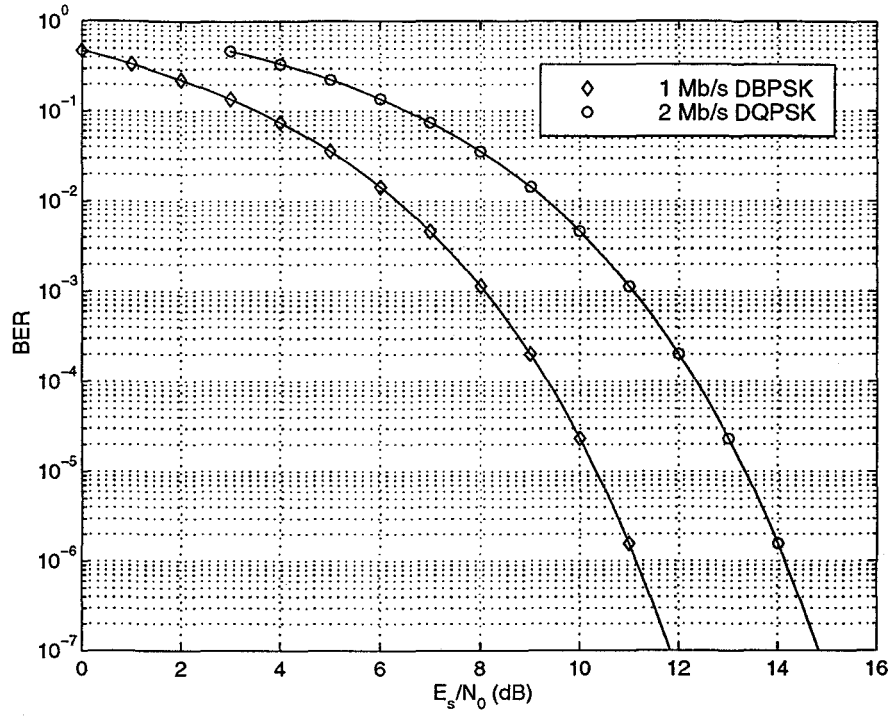


Figure 4.2 IEEE 802.11 compliant 1 Mb/s and 2 Mb/s physical layer BER performance

Figure 4.2 shows the IEEE 802.11 compliant physical layer BER vs. the received symbol-to-noise power ratio (E_s / N_0) performance for both the 1Mb/s DBPSK and 2Mb/s DQPSK modulation. Note that the E_s / N_0 is different from the E_b / N_0. The E_s / N_0 is not normalized with the number of bits/symbol and hence shows that DQPSK requires twice the transmit power of DBPSK for the same symbol rate. If the DBPSK and DQPSK curves were normalized with their respective number of bits/symbol, the resulting BER vs. E_b / N_0 curves would be the same, indicating that DQPSK is no more power efficient than DBPSK.

The FER vs. E_s / N_0 for both IEEE compliant DBPSK 1Mb/s and DQPSK 2Mb/s physical layers are shown in figure 4.3. The markers show the simulated values and the lines indicate the theoretical performance. Again, we see that the DBPSK and DQPSK curves differ by a factor of 2 (3 dB) in E_s / N_0 performance. The horizontal line on

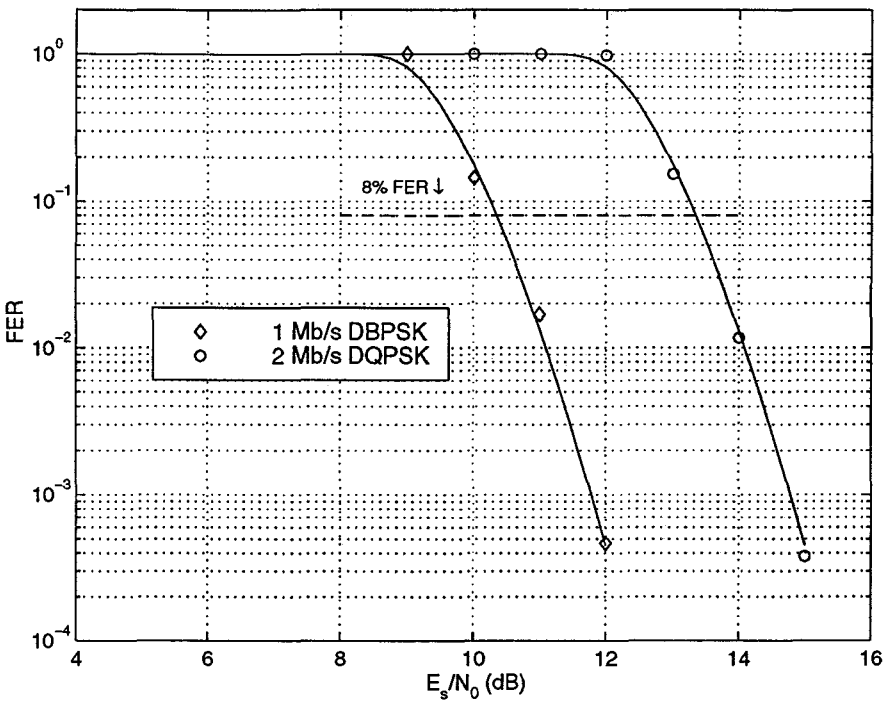


Figure 4.3 IEEE 802.11 compliant 1 Mb/s and 2 Mb/s physical layer FER performance

figure 4.3 represents the 8% FER which is the maximum FER allowed by the standard. So the simulations show that approximately 10.5 and 13.5 dB E_s / N_0 is required to meet the specification for the DBPSK and DQPSK modulation respectively.

4.3 Higher Data Rate System Performance

Due to the complexity of the turbo code simulations, performance plots for this section will be limited to approximately 10^{-5} BER. This is valid for the system under consideration since a BER of 10^{-5} will give the E_b / N_0 required to meet the IEEE 8% FER specification. Also, it has been shown through simulation [20, 21] that turbo code performance begins to flatten out after about 10^{-6} BER. As in section 4.2, approximately 10 bit errors were simulated before the simulated BER was accepted.

Due to the complexity of turbo code analysis, theoretical performance curves are difficult to obtain and are not developed in this thesis. Many factors affect the turbo code performance, including the RSC code, interleaver, puncturing, decoding algorithm, etc. Currently, simulation appears to be the most tractable measure for turbo code performance.

Figure 4.4 shows the performance of one link of the proposed $(37_8, 21_8)$ turbo code used in the design of the higher data rate system. Each plot represents the BER vs. E_b / N_0 performance for a different number of decoding iterations.

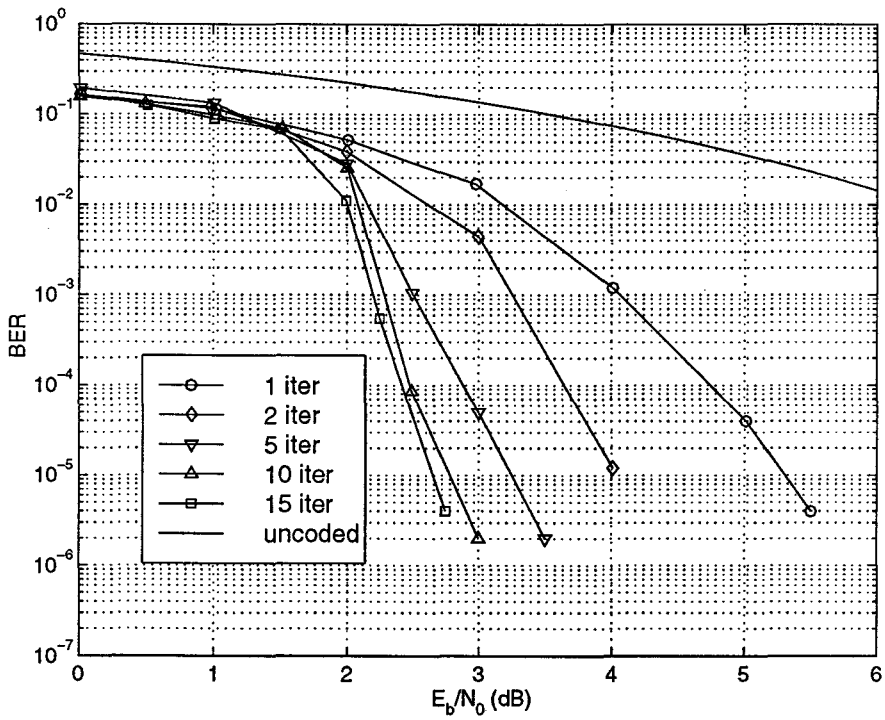


Figure 4.4 Turbo code BER performance for 1 link versus the number of decoding iterations

The performance of the IEEE 802.11 compliant link (uncoded) is included in figure 4.4 for comparison purposes. Each turbo code plot shows the average BER performance for

100 different randomly generated interleavers. This is done to give an indication of the turbo code performance for an average pseudo-random interleaver.

From figure 4.4 it is clear that the BER performance, as expected, improves as the number of iterations increases. This indicates that the extrinsic information gained in the decoding cycle is indeed useful in improving the performance of the subsequent decoding cycle. For small numbers of iterations, the performance gained by increasing the number of iterations is large. However, as the number of iterations increases, a point of diminishing returns is reached and more iterations do not give that much better results. Thus, for a practical system such as the proposed higher data rate system of this thesis, an engineering judgment must be made to determine the number of decoding iterations that should be used. As more iterations are used in the decoding process, there is a proportional increase in the decoding delay. Thus, if a large number of iterations are to be used, then a fast SOVA decoder is required to maintain an acceptable decoding delay. Therefore, given the intended application of the proposed system and tradeoffs at hand, 5 iterations will be chosen for the decoding algorithm. It is felt that from the results of figure 4.4, 5 iterations give relatively good performance and yet keep the number of decoding iterations to a reasonable level for implementation purposes.

For the purposes of comparison, figure 4.5 shows the performance gain using a BCH block code. BCH block codes are the most powerful known algebraic codes and have been used widely in many applications [17]. This block code takes 85 data bits and encodes them into 127 bits. The code is capable of correcting up to 6 errors in each block, hence is called a (127,85,6) BCH block code. From figure 4.5, the BCH block code provides about 3 dB of coding gain over the uncoded IEEE 802.11 compliant physical layer at a BER of 10^{-5}. This results in less coding gain than the turbo code of figure 4.4 for only one iteration. Thus, turbo codes are definitely superior over blocks codes at the cost of a higher decoding complexity.

Berrou performed BER simulations of the $(37_8, 21_8)$ turbo code in [21]. The simulated BER results Berrou obtained were somewhat better than the results of figure 4.4. For example, for 2 iterations, Berrou obtained a BER of 10^{-5} at approximately 2.7 dB, a 1.3 dB improvement over the results in figure 4.4. For 6 iterations, Berrou's results give a BER of 10^{-5} at 1 dB, roughly a 2 dB improvement over these simulations.

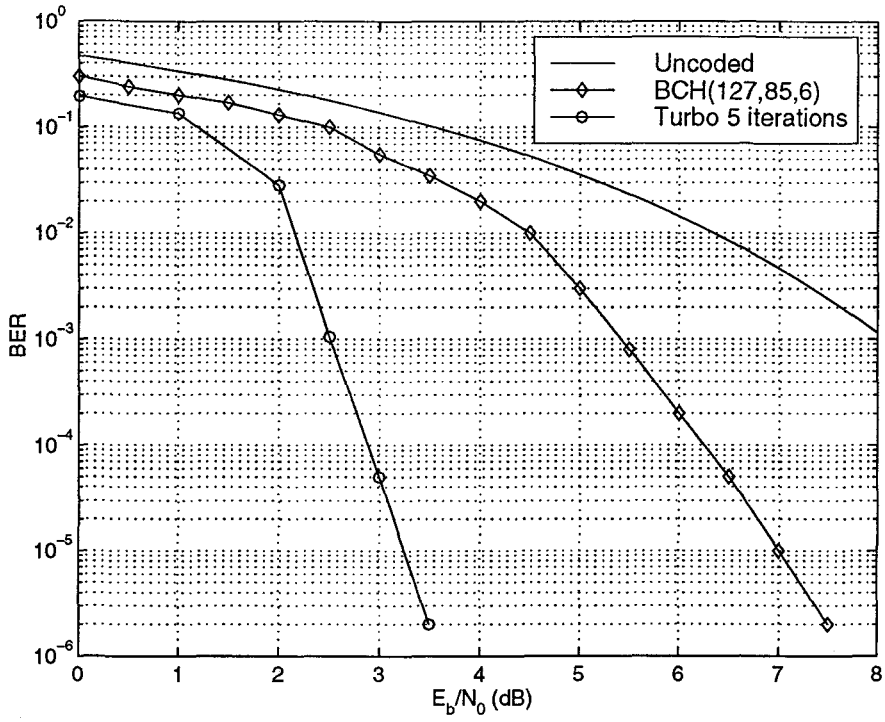


Figure 4.5 BCH (127,85,6) block code and Turbo code BER performance

This may not sound like a significant improvement, however, later it will be shown that small improvements in coding gain can significantly increase the number of simultaneous links (resulting in increased data rate) in this proposed system.

There are two main reasons for Berrou's increased performance in comparsion to the results of figure 4.4. First, Berrou's decoding algorithm uses a more complex and accurate method to determine the extrinsic LLR values in the Viterbi decoding. This results in better extrinsic information to feed back for the next decoding cycle and causes greater performance gains for an increase in the number of iterations, especially for small numbers of iterations. Second, Berrou uses a fixed length block interleaver that is designed specifically to improve the performance of the turbo code.

From figure 4.4, the coding gain improvement for the $(37_8, 21_8)$ turbo code for 5 decoding iterations is determined to be 6.5 dB (4.5) compared to the uncoded BPSK (figure 4.1) and 7.5 dB compared to the IEEE 802.11 DBPSK system. Using equation

3.10, repeated below for convenience, the theoretical number of possible links for the proposed system is

$$M = 1 + \frac{PG \cdot CG}{\frac{E_b}{N_0}\min} = 1 + \frac{11 \cdot 4.5}{9.1} = 6.4 \text{ links}$$

For comparison, the number of links that could be obtained if Berrou's interleaver and decoding algorithm were used for 3 iterations (coding gain of approximately 8 dB = 6.3) is

$$M = 1 + \frac{11 \cdot 6.3}{9.1} = 8.6 \text{ links}$$

Thus, for a 1.5 dB improvement in coding gain, this adds 2 more links and increases the data rate by 33%. So, it is clear that relatively small increases in coding gain can provide large improvements in data rate.

Figure 4.6a shows the simulated BER vs. E_b / I_0 performance for the proposed higher data system when 2 links are used and 5 decoding iterations are performed. Figure 4.6b shows the resulting simulated FER for 2 links. Note, that the plots show the energy per bit to interference ratio E_b / I_0 performance, which is different than the energy per bit to noise ratio E_b / N_0. The interference is calculated as follows. First, the total transmit power S_T of all links is measured. Since each of the link signals is transmitted with the same power, the transmit power of one link is S_T / M where M is the number of links. For a link of interest m_i, the other $M-1$ links are the interference this link experiences. This is an identical situation as a base station receiver in a cellular CDMA system with M users when perfect power control is assumed. Since the proposed system is a spread spectrum system, the spreading in the transmitter makes each of the signals for the different links look somewhat white in frequency, but bandlimited. Thus, if we assume that the interference from the other links is Gaussian, then the input signal to interference ratio (SIR) with AWGN can be determined by

$$\frac{S}{I_0} i = \frac{S_T / M}{\frac{S_T (M-1)}{M} + N_0} \tag{4.2}$$

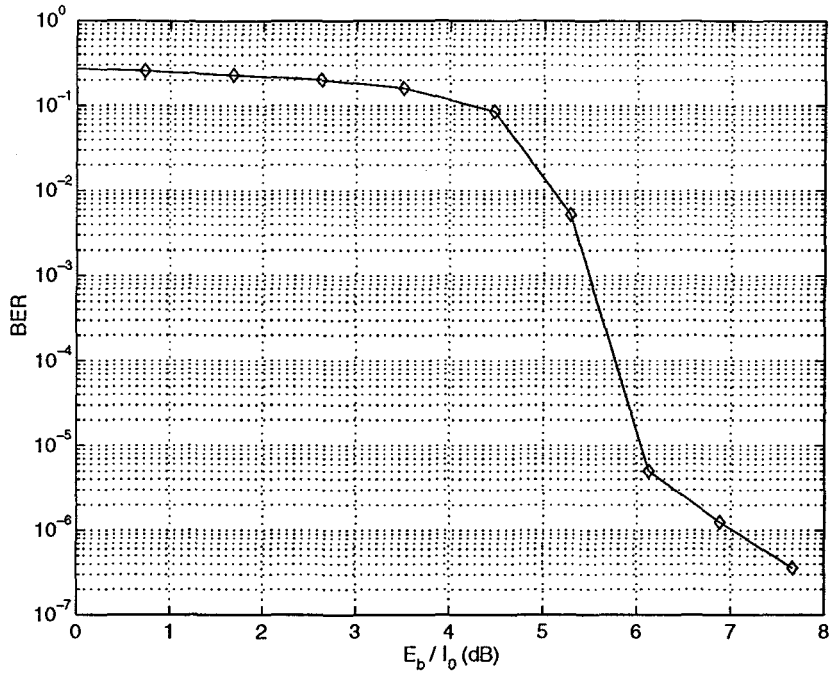


Figure 4.6a BER performance of the proposed system for 2 links

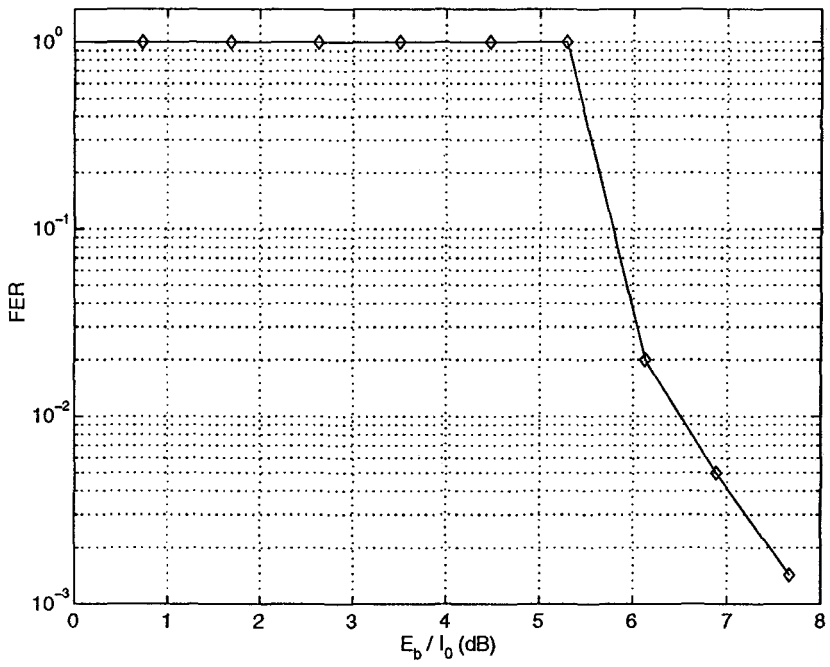


Figure 4.6b FER performance of the proposed system for 2 links

where N_0 is the AWGN power. The assumption that the interference is Gaussian is not valid for small M as is the case for this system. However, this does not matter as it will show up as a degradation in performance. This is clear from figure 4.6a which shows that the BER vs. E_b / I_0 performance is roughly 3 dB worse than for BER vs. E_b / N_0 performance in figure 4.4 for 5 iterations. This degradation in performance is explained by the non-zero cross correlation of the Gold spreading codes used in this design. If better codes were used with smaller cross-correlation values, one would expect better performance.

Figures 4.7a and 4.7b show the BER and FER performance for 3 links of the proposed system. For the multiple link simulations, enough packets were simulated to generate approximately 10 packet errors at high E_b / I_0 values before the FER value was accepted. This required approximately 10,000 packets to be generated to get the 10 errors at a FER of 10^{-3}. Thus, the FER results should reflect the true probability of packet error P_{fe} to within a factor of 2. The BER plots, which would take far fewer packets to simulate, are the resulting bit error rates from the FER simulation. Thus, the BER simulations reflect very accurately the true probability of bit error P_{be}. A coarse sampling of E_b / I_0 values was used due to the complexity of the simulations and limited computational resources.

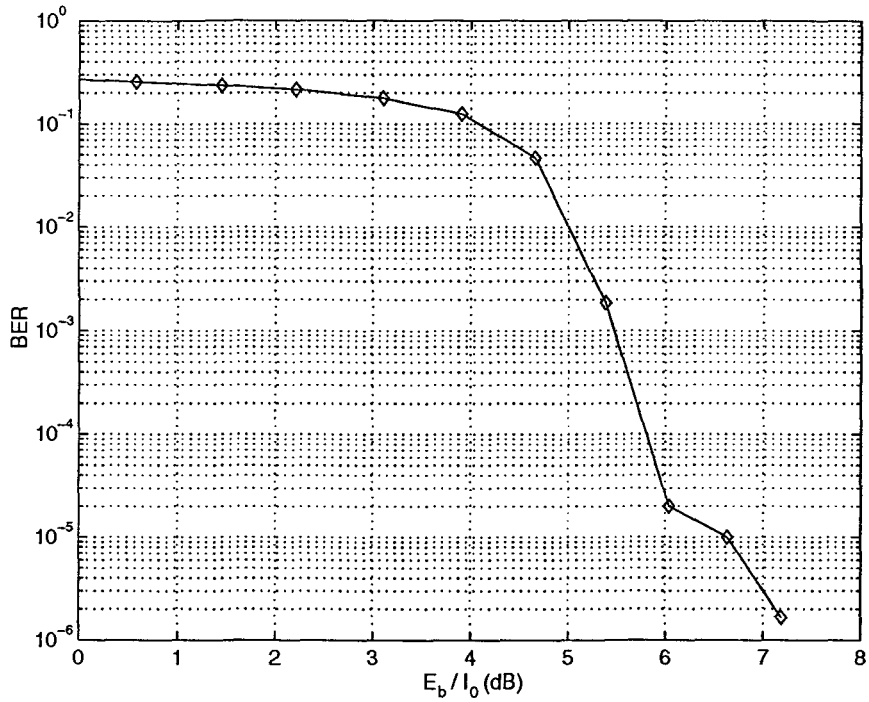


Figure 4.7a BER performance of the proposed system for 3 links

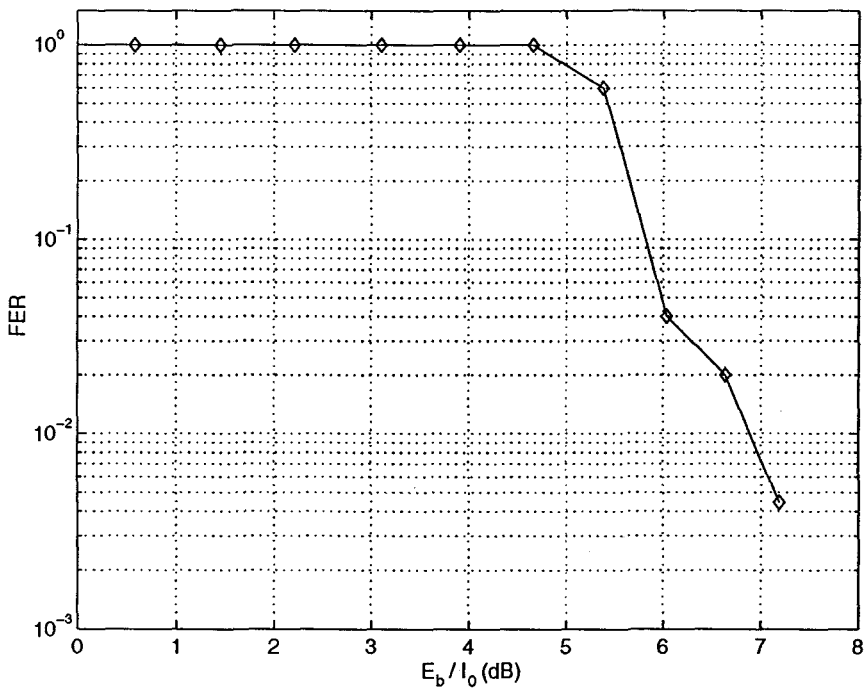


Figure 4.7b FER performance of the proposed system for 3 links

Figure sets 4.8, 4.9, and 4.10 show the BER and FER performance for 4, 5, and 6, links respectively. The performance for 4 and 5 links is essentially the same as for fewer links, which is to be expected since the E_b / I_0 includes the increased interference.

For figure 4.10a and 4.10b, the performance is limited by the amount of interference generated by the other links and the system no longer can achieve the BER needed to satisfy the required FER specified by the IEEE 802.11 standard. This can be shown by considering equation 4.2. If the noise power N_0 is constant and the total signal power S_T is much larger than the noise power, then equation 4.2 can be approximated by

$$\frac{S}{I_0}_i \approx \frac{S_T/M}{\frac{S_T(M-1)}{M}} = \frac{1}{M-1} \tag{4.3}$$

Equation 4.3 shows that the input signal to interference ratio is bounded in an interference limited system. Thus, the energy per bit to interference ratio for the proposed OCDM system for 6 links is bounded by

$\frac{E_b}{I_0} = \frac{PG/R}{M-1} = \frac{11/0.5}{6-1} = 4.4$ (6.4 dB). This bound is indicated by the vertical line on figures 4.10a and 4.10b. We see from the simulated results that the performance is limited by this bound no matter how large the transmit power may be.

The simulation results presented show that the proposed higher data rate system is capable of sustaining 5 simultaneous links at the required 8% FER specification. The simulated system falls short of the theoretical number of links by one. This can be explained by the fact that the theoretical analysis did not include the interchip interference generated by the Gold spreading sequences. The interchip interference is caused by the non-zero cross correlation values of the Gold sequences, and reduces the performance of real world systems.

In chapter 3 it was shown that each link is capable of transmitting at a rate of 1.4 Msymbols/sec. This allows a data rate per link of 0.7 Mb/s for a BPSK implementation or a data rate of 1.4 Mb/s per link for a QPSK implementation. Therefore, the maximum data rate sustainable by this proposed system is 7 Mb/s. This represents over a three fold increase in data rate compared to the IEEE 802.11 compliant physical layer.

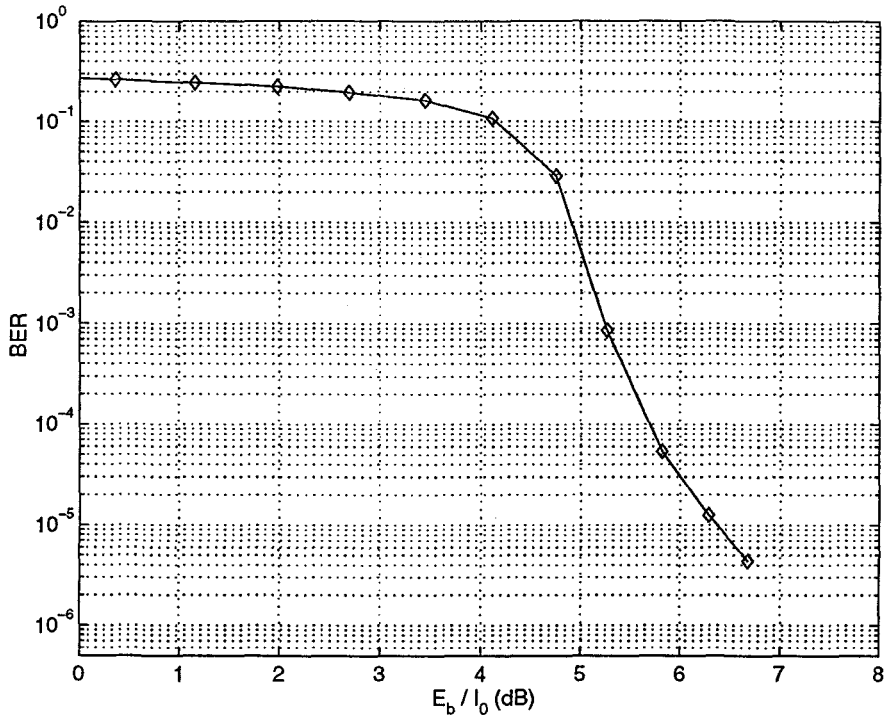


Figure 4.8a BER performance of the proposed system for 4 links

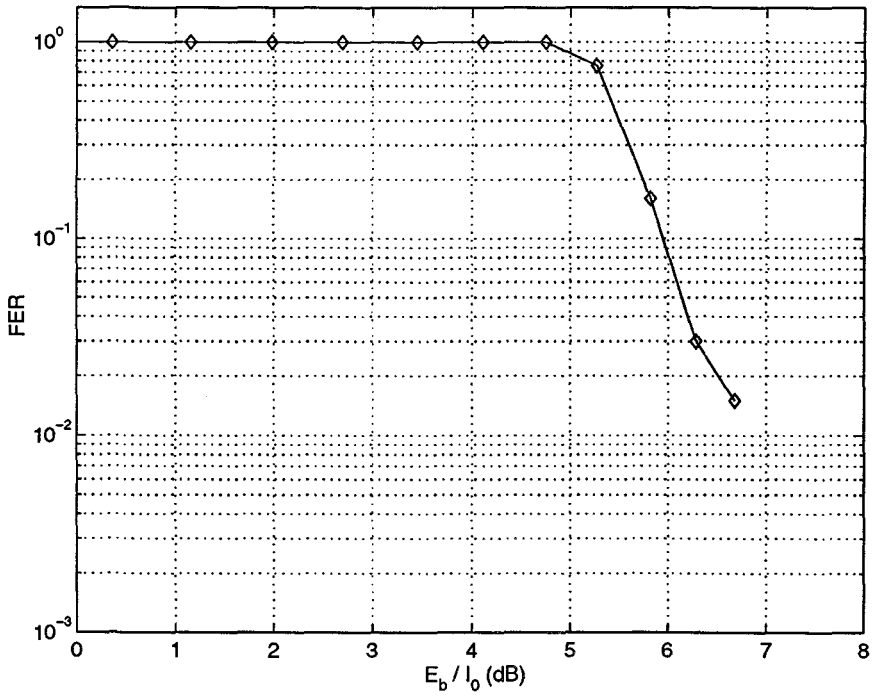


Figure 4.8b FER performance of the proposed system for 4 links

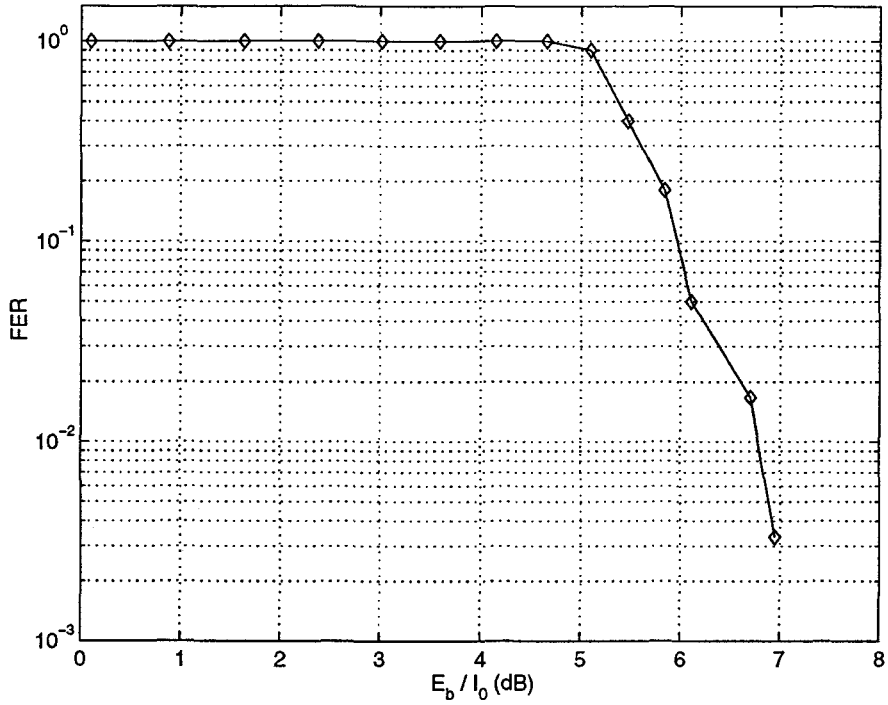


Figure 4.9a BER performance of the proposed system for 5 links

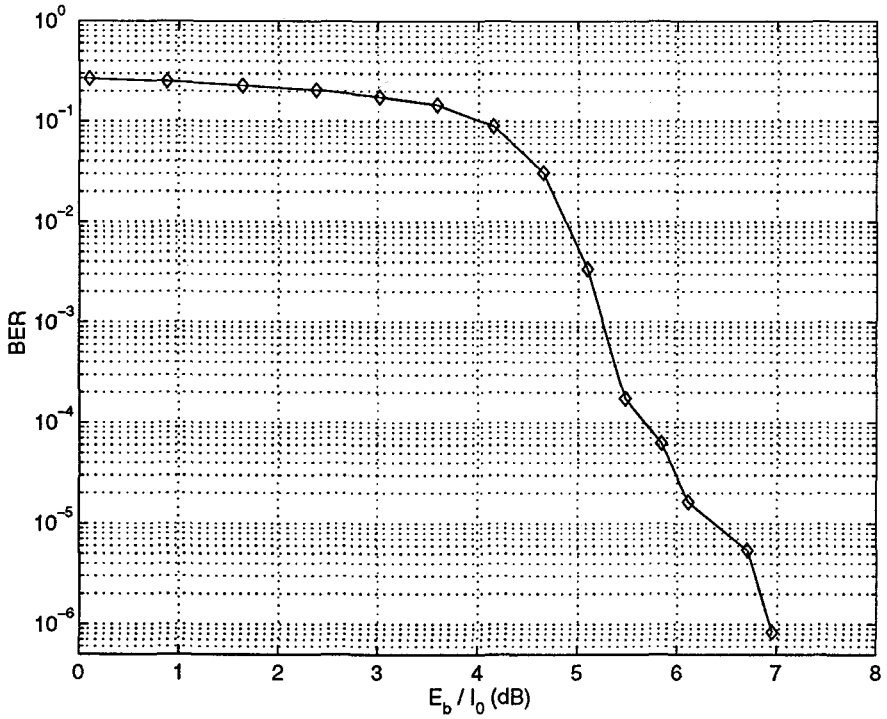


Figure 4.9b FER performance of the proposed system for 5 links

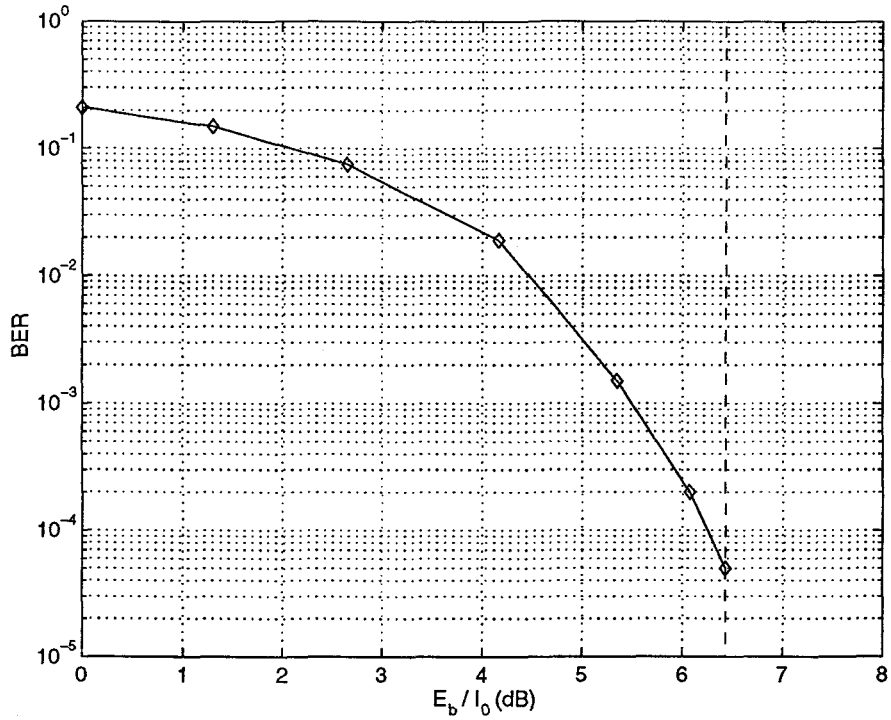


Figure 4.10a BER performance of the proposed system for 6 links

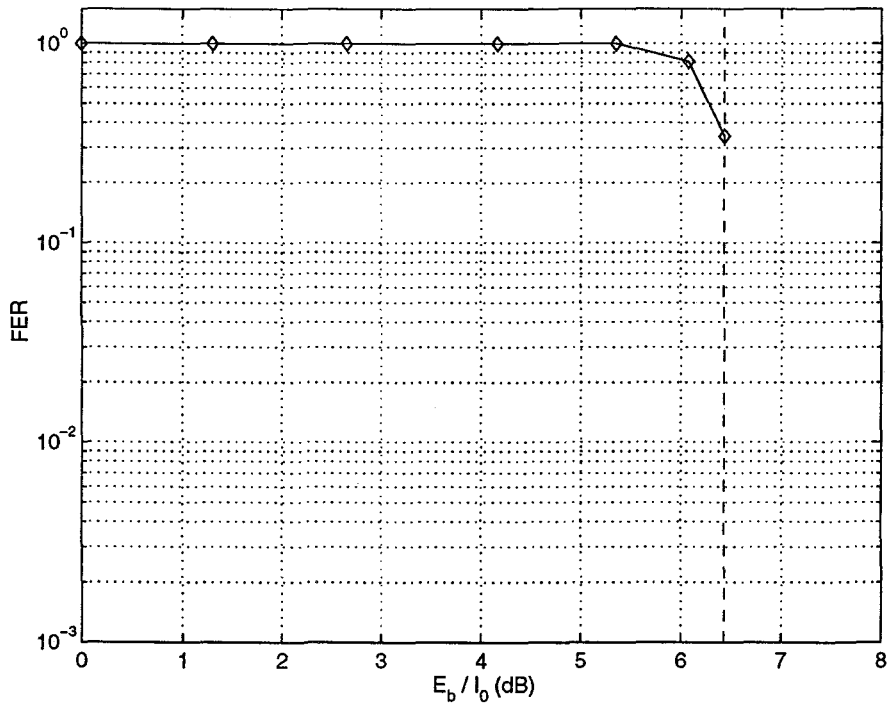


Figure 4.10b FER performance of the proposed system for 6 links

5. SUMMARY, CONCLUSIONS, AND FUTURE RESEARCH

5.1 Summary

The market for Wireless Local Area Networks (WLAN) has expanded rapidly over the last decade and continues to grow. WLAN standards have provided the much needed capability of products from different vendors to interoperate. However, in the case of IEEE 802.11-1997 draft standard, data rate is limited to 2 Mb/sec. This means that users would only have access to lower data rates when they use IEEE compliant devices, as compared to wired networks. But, because data rate is an important selling feature, two IEEE 802.11 working groups (a and b) and many companies designing IEEE 802.11 products have developed modulation schemes to improve the data rate of the IEEE standard. The proposed higher data rate system developed and simulated in this thesis is one solution to that problem. The system developed in this thesis uses a combination of spread spectrum multiple access (OCDM) and a turbo code, an aggressive channel encoding technique, to increase the data rate of the IEEE standard by 2-3 times.

Chapter 2 detailed the design for the IEEE 802.11 compliant physical layer. This development was necessary for several reasons. First, the literature on this subject is limited, so evaluation was required to obtain a baseline of performance. Second, the proposed higher data rate system is closely related to the compliant physical layer, so the development was a natural first step in this process. The issues dealt with in this design included the implementations of the scrambler/descrambler filters, DBPSK modulation, and pulse shaping.

The next step was to develop the higher data rate design. This process was explained in chapter 3. The higher data rate design was based upon orthogonal code division multiplexing (OCDM). This is a modification of CDMA that uses multiple spreading codes to allow simultaneous transmissions of equal power over several data links. First, the design and implementation of the spreading codes was discussed. Length 512 Gold codes were chosen due to the high autocorrelation and low 3-valued cross-correlation properties. Second, the pulse shaping/matched filter designed in chapter

2 was reconsidered to obtain a 40% increase in data rate. Third, the design of the turbo code was considered. This began with an analysis of the uncoded system to determine the coding gain needed to support the required number of links to achieve the desired data rate. From this the turbo code was selected, along with choosing an interleaver and decoding algorithm.

In chapter 4 the simulations of both the IEEE compliant and proposed higher data rate physical layers were presented. First, the simulation results of the compliant layer were presented. These results included simulations of the degradation caused by both the DBPSK modulation and the scrambling/descrambling in the compliant layer. Both BER and FER simulations were presented and a comparison to the theoretical results was done. Next, the BER rate performance of the proposed system was considered for 1 link. This showed the performance gain of the turbo code over the IEEE compliant layer. From these simulations the number of decoding iterations was chosen and the coding gain determined. With this coding gain, the theoretical number of possible links was determined. Finally, the packet-based FER and BER simulation results for all links up to 6 was presented.

5.2 Conclusions

This thesis had two major contributions. The first contribution was to design and simulate the IEEE 802.11-1997 DSSS physical layer. Chapter 2 presented the design of this compliant layer. The major functions of the layer were analyzed and were found to work as predicted by the standard. Both the BER and FER computer simulation results demonstrated that the BER of this layer was 6 times greater than coherent BPSK, which resulted in a 1 dB reduction at a BER of 10^{-5}. The results also indicate that roughly 10.5dB of E_b / N_0 is required to obtain the maximum FER of 8% for a AWGN channel. This compares well to the results given in [11] and to the theoretical results predicted in chapter 2.

The second and more significant contribution of this thesis was the design and simulation of a new physical layer to improve the data rate of an IEEE 802.11 device. The design is based on spread spectrum multiple access techniques and turbo coding. The simulation results clearly show that 5 simultaneous links are possible over an AWGN channel. This gives a maximum data rate of 7 Mb/s with QPSK modulation. This is a data rate improvement of 3.5 times over the 2 Mb/s IEEE 802.11 DQPSK layer.

In conclusion, this thesis was successful in achieving the goals established at the outset. This author believes that this thesis could open the door for future research in applying strong error correcting codes to improve the performance of next generation WLAN systems.

5.3 Future Research

There are many possible topics for future research in the WLAN area. By no means does the IEEE 802.11 standard represent the last word in WLANs. The IEEE 802.11 standard represents the second generation in WLAN development in the United States. Many outstanding issues remain to be resolved in future generations of WLANs.

Specifically, with reference to this thesis, several areas could be investigated. First, an analysis of the multipath performance of the proposed physical layer in this thesis is necessary. Indoor wireless systems encounter strong multipath signals and proposed systems must perform well in these environments. Second, a more detailed investigation of the usage of turbo codes in this application would likely result in improved coding gain, and thus increase the data rate. This research could include the design and evaluation of the pseudo-random interleavers. This is a very important component in determining turbo code performance. Also, an investigation into the decoding algorithm would also likely result in increased performance. As was seen in section 4.3, small improvements in the coding gain can result in significant increases in the data rate.

Bibliography

1. K. Pahlavan, T. Probert, and M. E. Chase, "Trends in Local Wireless Networks," *IEEE Commun. Mag.*, vol 33, no. 3, March 1995, pp. 88-95.

2. J. E. Padgett, C. G. Gunther, and T. Hattori, "Overview of Wireless Personal Communications," *IEEE Commun. Mag.*, vol 33, no. 1, January 1995, pp. 28-41.

3. K. Pahlavan, A. Zahedi, and P. Krishnamurthy, "Wideband Local Access: Wireless LAN and Wireless ATM," *IEEE Commun. Mag.*, vol 35, no. 11, November 1997, pp. 34-40.

4. Federal Communications Commission, *In the Matter of Amendment of the Commission's Rules to Provide for Operation of Unlicensed NII Devices in the 5 GHz Frequency Range*, ET Docket No. 96-102, adopted Jan. 9, 1997.

5. R. O. LaMaire et al., "Wireless LANs and Mobile Networking: Standards and Future Directions," *IEEE Commun. Mag.*, vol 34, no. 8, Aug. 1996, pp. 86-94.

6. IEEE Std 802.11-1997, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, Piscataway, NJ: IEEE Standards Board, June 26, 1997.

7. M. K. Simon et al., *Spread Spectrum Communications Handbook.* New York: McGraw-Hill, 1994.

8. B. P. Crow et al., "IEEE 802.11 Wireless Local Area Networks," *IEEE Commun. Mag.*, vol 35, no. 9, Sept. 1997, pp. 116-126.

9. C. Andren, "Short PN Sequences for Direct Sequence Spread Spectrum Radios," Wireless Communications Technical Papers, Harris Corp., http://www.semi.harris.com/prism/papers/shortpn.pdf, May 11, 1997.

10. J. D. Gibson, *The Mobile Communications Handbook.* Boca Raton, FL: CRC Press, 1996.

11. C. Andren, "A Comparison of Frequency Hopping and Direct Sequence Spread Spectrum Modulation for IEEE 802.11 Applications at 2.4 GHz," Wireless Communications Technical Papers, Harris Corp., Melbourne FL, http://www.semi.harris.com/prism/papers/ds-v-th.pdf, May 11, 1997.

12. V. K. Garg, K. Smolik, and J. E. Wilkes, *Applications of CDMA in Wirelees/Personal Communications.* London: Prentice-Hall, 1997.

13. C. Andren, "11Mb/s Modulation Techniques," Wireless Communications Technical Papers, Harris Corp., Melbourne FL, http://www.semi.harris.com/prism/papers/modtech.pdf, Aug. 19, 1998.

14. FCC specification 15.247 part 15, Oct 1, 1998.

15. C. Andren, "Modulation Techniques for High Speed WLAN Systems," Wireless Design Online, http://news.wirelessdesignonline.com/design-features/19980826-2747.html , Aug 26, 1998.

16. M. C. Jeruchim, P. Balaban, and K. S. Shanmugan, *Simulation of Communication Systems.* New York: Plenum Press, 1992.

17. S. B. Wicker, *Error Control Systems for Digital Communication and Storage.* Upper Saddle River, NJ: Prentice Hall, 1995.

18. B. P. Lathi, *Modern Digital and Analog Communication Systems.* New York: Oxford University Press, 1989.

19. R. Gold, "Optimum binary sequences for spread spectrum multiplexing," *IEEE Trans. Inform. Theory*, IT-13, Oct. 1967, pp. 619-621.

20. B. Sklar, "A Primer on Turbo Code Concepts," *IEEE Commun. Mag.*, vol 36, no. 12, December 1997, pp. 94-102.

21. C. Berrou and A Glavieux, "Near Optimum Error Correcting Coding and Decoding: Turbo Codes," *IEEE Trans. on Comm.*, vol. 44, no. 10, October 1996, pp. 1261-1271.

22. J. Hagenauer and P. Hoeher, "A Viterbi Algorithm with Soft-Decision Outputs and its Applications," *Proc. of IEEE Globecom* '89, Dallas, Texas, pp. 47.11-47.17, Nov. 1989.

23. C. Berrou, "A Low Complexity Soft-Output Viterbi Decoder Architecture," *IEEE Proc. on Inter. Conf. on Comm.*, May 1993, 737-740.

24. D. Kwan and S. Kallel, "A Rate-k/n Heuristic Soft-Output Viterbi Algorithm (SOVA) that is Postdetector-Compatible," *IEEE Trans. on Comm.*, Vol 46, no. 5, May 1998, pp. 621-626.

25. C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes," *IEEE Proc. ICC '93*, Geneva, Switzerland, May 1993, pp. 1064-1070.

26. R. Durrett, *The Essentials of Probability.* Belmount, CA: Duxbury Press, 1994

27. W. Tranter and K. L. Kosbar," *IEEE Commun. Mag.*, vol 32, no. 7, July 1994, pp. 26-35.

28. I. Bar-David and R. Krishnamoorthy, "Barker Code Modulation for High-Rate Communication in the ISM Bands," *Bell Labs Technical Journal*, vol 1, Autumn 1996, pg. 21-40.

29. J. Fakatselis, "Proposed IEEE 802.11 Direct Sequence Spread Spectrum Physical Layer Characteristics," Wireless Communications Technical Papers, Harris Corp., Melbourne FL, http://www.semi.harris.com/prism/papers/802dspap.pdf.

30. R. Gold, "Maximum Recursive Sequences with 3-Valued Recursive Cross-Correlation Functions," *IEEE Trans. Inform. Theory*, IT-14, Jan. 1968, pp. 154-156.

31. S. Jost and C. Palmer, "New Standards and Radio Chipset Solutions Enable Untethered Information Systems: PRISM™ 2.4 GHz "Antenna-to-Bits" 802.11 DSSS Radio Chipset Solution," Wireless Communications Technical Paper, Harris Corp., Melbourne FL, http://www.semi.harris.com/prism/papers/ws3.pdf.

32. R. Kohno, R. Meidan, and L. Milstein, "Spread Spectrum Access Methods for Wireless Communications," *IEEE Commun. Mag.*, vol 33, no. 1, Jan. 1995, pp. 58-67.

33. J. Rothweiler, *IEEE Potenials*, February/March 1999, pp 23-25.