ADAPTIVE CODING FOR WIRELESS DATA NETWORKS

by

Tingfang Ji

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Electrical Engineering: Systems)
in The University of Michigan
2001

Doctoral Committee:

Professor Wayne E. Stark, Chair
Assistant Professor Achilleas Anastasopoulos
Professor David L. Neuhoff
Assistant Professor Brian Noble

ABSTRACT

ADAPTIVE CODING FOR WIRELESS DATA NETWORKS

by
Tingfang Ji

Chair: Wayne E. Stark

Current cellular systems typically have spectral efficiency in the range of 0.03-0.05 b/s/Hz/sector and the talk time is about 3 hours. The dramatic increase in demand for high speed wireless data services requires the next generation wireless networks to significantly improve the throughput and energy efficiency while maintaining a reasonable delay. In this thesis, we investigate rate adaptive coding and automatic repeat request (ARQ) schemes to meet these new challenges.

An analytical framework is investigated for performance evaluation of general type-I and type-II hybrid ARQ protocols based on rate compatible error correcting codes over finite-state channels. As an example, an analysis of Reed-Solomon code based systems is shown to yield insight into the effect of the protocols, channel parameters, design parameters, and decoder implementations on the system performance. A family of powerful rate compatible multiple product codes are constructed and shown to have superior performance compared with S-random interleaved turbo codes at moderate signal-to-noise ratio. A few practical rate adaptive ARQ pro-

tocols are then applied to an asynchronous direct-sequence code division multiple access network and compared in terms of the total network throughput, average delay, and throughput under energy constraints.

To my parents

ACKNOWLEDGMENTS

I would like to express many thanks to my advisor Professor Wayne Stark for his excellent guidance and encouragement during the course of my study. His time and effort in discussing our research and his extensive knowledge in the field of communications have been tremendous resource.

I would also like to thank each of my committee members, Professor Wayne Stark, Professor Achilleas Anastasopoulos, Professor David Neuhoff, and Professor Brian Noble for the time they spent reading and evaluating my thesis.

I would also like to thank the financial support from DARPA GloMo Project contract No. F30602-97-C-0314 and Army Research Office MURI project DAAH01-96-0337.

I would also like to thank all my friends at the University of Michigan. Without our friendships, life in Ann Arbor would not have been quite so pleasant. I would also like to thank Beth Olson, Ann Pace, and Susan Yale for their kind help in graduate school.

I would also like to thank my uncle Shiliang and aunt Anqi for encouraging me to pursue a doctoral degree. Without their support, I would not have been enjoying me graduate study in the United States.

Finally I would like to express my utmost gratitude to my parents for their endless love and support. I would also like to thank my brother and folks at home for taking good care of my parents while I am away from home.

TABLE OF CONTENTS

CHAPTERS

LIST OF FIGURES

Figure

LIST OF TABLES

Table

CHAPTER I

Introduction

1.1 Motivation

Demands for wireless broadband connection are growing explosively. Most of the emerging wireless networks focus on providing integrated packet-oriented transmission of text, voice, video, and computer data. Current cellular systems are designed to handle mostly voice services, which usually have data rates on the order of a few thousand bit per second (bps). For data services, data rates vary from tens of bps, for messaging and email services, to more than 1 Mbps, for multimedia streaming. For voice services, the delay is usually required to be less than tens of millisecond (ms) to ensure intelligent conversations. In many cellular system standards, the voice bit streams are broken into frames of 20 ms duration, and the frame error rate is required to be less than 0.01 for 90 to 95% of users at any time. For most data services, the delay constraint is less stringent, but the error probability requirements are quite different as some applications require error free communications, such as web browsing (file transferring), while others are resilient to sparse errors, such as video streaming. Therefore, designing new generation wireless broadband data networks offers new challenges and opportunities.

A layered architecture has been adopted in most data networks such as the Internet. In this thesis, we are concerned with the lowest two layers of the open system interconnection seven layer architecture [79]. The lowest layer of the hierarchy, the physical layer, provides a virtual link for transmitting bits between any pair of nodes joined by a physical communications channel. The layer above the physical layer, the data link layer, transforms the unreliable transmission of bits at the physical layer into a link that appears free of undetected transmission errors to the upper layer. In the rest of the section, we are going to show that in wireless networks, joint design of the physical layer and the data link layer is central to the improvement of network performance.

1.1.1 Wireless Physical Layer

The basic difference between wireless communications and wired communications is that users with wireless connection can enjoy tether-less communications and access to information at any location by sharing a common medium — *space*. To meet the exponential growth of the data rate requirement of emerging applications, more fibers or cables can be laid down in wired communications. However, since a common medium, the radio frequency, is shared by all users in wireless communications, the system capacity is finite in nature by physical laws. Hence, meeting the explosive growth of demands with finite resources makes physical layer design for wireless communication particularly challenging.

As shown by Shannon in his phenomenal work on information theory, there is a fundamental limit on the data rate of error free communications as a function of the frequency spectrum, the statistics of the channel, and the transmitting power [76]. For the voice band telephone channel, the V.90 modems are already operating at close to Shannon's capacity, but most wireless systems are still spectrally inefficient due to special challenges of the wireless medium. We will discuss these challenges and the physical layer techniques that mitigate these channel impairments and constraints in the following two subsections.

1.1.1.1 Channel Impairments and Constraints

One of the challenges in a wireless system is multipath fading. Fading occurs due to multiple paths between the transmitter and receiver which result from reflection, diffraction, and scattering of radio waves during propagation in space. The delay, amplitude, and phase for signals received on each path are time varying so the signals along different paths sometimes add constructively and sometimes cancel out each other. When multipath signals cancel each other, the signal is said to be in a null or a deep fade. It is not uncommon to have 30 decibel (dB) fades in mobile communications. It is obvious that such large channel dynamics make low-error-rate communications over wireless channels much harder than over wired channels that have relatively stationary characteristics.

Multi-user interference is anther major limiting factor in the performance of many multi-user wireless communication systems. Since many users are sharing the same medium, signals from other users will interfere with the reception of the desired signal. In the current cellular systems, mobile users are connected to others through a *basestation*. The assignment of a mobile device to a basestation is based on the geographical location of the user. The area in which users are connected to one basestation is called a *cell*. Interference can also come from basestations and users from other cells. Since users are located randomly geographically, some users need larger power to connect with the basestation, and others use less powers. The heterogeneous nature of the power and location of users makes some users subject to much more severe interference than others.

Since most mobile units operate on batteries, the power efficiency or energy efficiency is a critical performance metric for practical purpose. Power consumption includes both the radiated signal power and the power dissipated by the electronic circuits in transmitters and receivers. As we know, the power dissipation of digital circuits grows proportionally with the clock frequency of the processor. Applications with high data rate, which require high clock frequency in the digital signal processing chips, hence, make the design of energy efficient systems ever more challenging. The transmitting power and frequency spectrum of radio signals are also regulated and constrained by government bodies in every country, such as the Federal Communications Committee of the United States.

1.1.1.2 Forward Error Correcting Codes

To mitigate the channel impairments and constraints, there has been intensive research on modulation, forward error correcting (FEC) codes, and other physical layer technologies.

Modulation is a process of transforming a message into a form suitable for transmission. In digital modulation techniques for wireless systems, messages are usually mapped onto a finite signal set at a radio frequency. Each signal in the signal set may have different amplitude, phase, or frequency. If each signal in a modulation signal set carries one bit of information, the modulation scheme is called a binary modulation scheme; otherwise, it is called a multi-level modulation scheme. A demodulator estimates the transmitted signal based on the received signal. If the alphabet size of the demodulator output equals the alphabet size of the transmitted signal set, the demodulator is said to be a hard-decision demodulator. Otherwise, it is called a soft-decision demodulator. For example, consider a binary modulation scheme. A three level soft-decision demodulator generates 0 or 1 if the probability of demodulating to the correct bit is close to one, otherwise generates an *erasure*. Many FEC codes take advantage of the soft-decision information from the demodulator to provide more reliable communications.

FEC coding is used to retrieve messages from the corrupted version of the transmitted signals at the demodulated output. The corruption could be due to thermal noises from electronic devices, interferences from other users, or multipath fading. The modulator, channel, and demodulator are often considered as a discrete "super channel" by the FEC codes. By introducing known redundancy to the information bit stream, the encoder imposes a certain structure on the transmitted codewords at the transmitter. Then, by exploiting the structure of the received words, the receiver can effectively recover the corrupted parts of the signal from other reliable parts of the signal.

Since the redundancy due to coding consumes both bandwidth and energy resources, it is ideal to have FEC codes that achieve error requirements with minimum transmitting power and redundancy. The channel capacity, the highest achievable data rate, has been studied by Shannon and other information theorists for wired and wireless channels. As shown by Shannon, for communications at a rate less than the channel capacity, a randomly chosen FEC code can provide a packet error rate that decreases exponentially with the codeword length. Furthermore, the sphere-packing bound, as shown in (1.1), provides a lower bound on the achievable error performance for a given set of bandwidth expansion and delay parameters [77]. In the following example, we are going to illustrate the relation among bandwidth efficiency, power efficiency, delay, and error performance for the best possible FEC codes over additive white Gaussian noise (AWGN) channels.

Consider an equivalent discrete system model of a system with binary modulation over an AWGN channel. Consider a binary FEC code that encodes K information bits and generates N code bits. Let $R = \frac{K}{N}$ denote the code rate. The larger R, the less bandwidth expansion the code has. The smaller K, the less encoding and decoding delay the code has. Suppose a code bit is modulated onto a bipolar signal $X \in \{+\sqrt{E_s}, -\sqrt{E_s}\}$. The demodulator output Y is given by $Y = X + G$, where G is a Gaussian random variable of zero mean and variance $N_0/2$. The transmitted energy per information bit is given by $E_b = E_s/R$. A system with smaller E_b/N_0 is more power efficient than a system with larger E_b/N_0.

The sphere packing lower bound on word error probability of codes with N code bits and K information bits over an AWGN channel is given by [77]

$$P_e \geq P(N-1, \sqrt{\frac{2NE_s}{N_0}}, \sqrt{N-1}\cot\theta_1) \tag{1.1}$$

where $P(f, \delta, x)$ is the cumulative density function of a non-central t-distribution of f degrees of freedom and a bias of δ, and θ_1 satisfies

$$\frac{(N-1)}{N\pi^{1/2}} \frac{\Gamma(\frac{N}{2}+1)}{\Gamma(\frac{N+1}{2})} \int_0^{\theta_1} (\sin\theta)^{N-2} d\theta = \frac{1}{2^K}. \tag{1.2}$$

In Figure 1.1, the sphere-packing lower bounds on the required E_b/N_0 for a word error rate 10^{-2} are shown versus the code rate R for information block lengths from 10 to 5000 bits. It is observed that the sphere-packing bound on the required E_b/N_0 increases as a function of the code rate and decreases as a function of the information block length. In current cellular systems, worst case designs are used for choosing the code rate, signal power, and information block length. For instance, consider a channel with an E_b/N_0 that follows a Gaussian distribution of mean 3 dB and standard deviation 1.83 dB. Then, with a probability of 0.95, the E_b/N_0 is below

0 dB. Assume the quality of service requirement is that users have word error rate less than 10^{-2} for more than 95% of the time. Then a worst case design requires the use of FEC code of length at least 1000 and rate below 0.35 according to the sphere packing bound, such that it may provides $P_e < 10^{-2}$ for $E_b/N_0 > 0$ dB. If the code rate can be chosen intelligently depending on the channel condition, however, FEC codes with much smaller block lengths and high code rates can be used. For instance, when the channel is known to have an E_b/N_0 greater than 2 dB, a length 100 and rate 0.7 code would satisfy the sphere-packing bound as shown in Figure 1.1. Therefore, in theory we can potentially shorten the delay by an order of magnitude and double the code rate by exploiting the channel side information and utilizing a rate-adaptive coding system.

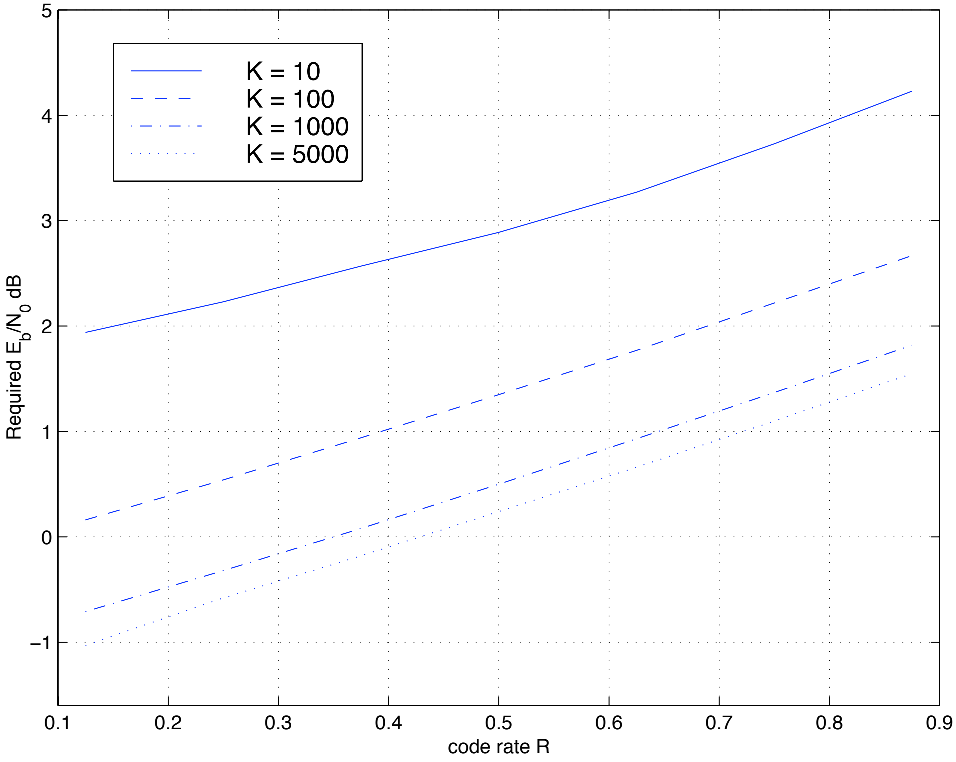


Figure 1.1: Sphere-packing lower bound on required E_b/N_0 for word error rate 10^{-2} versus the code rate R for information block lengths from 10 to 5000 bits

1.1.2 Data Link Layer

The data link layer provides higher layer protocols a virtual link free of undetected errors between two nodes with physical communications connections. For wireless networks, the access control to the shared channel is a challenging task. Medium access control (MAC) protocols are introduced to deal with this problem. Time-division multiple access (TDMA) and code-division multiple access (CDMA) have been widely used in the second generation cellular systems. In a TDMA system, each user is assigned a unique time slot, and only one user can transmit at one time. Enhancement with adaptive coding to the current TDMA systems have been introduced in the third generation (3G) IS-136 protocols. In a CDMA system, each user's narrow-band signal is spread to the entire frequency spectrum by scrambling with a noise-like spreading sequence. A conventional receiver recovers the message by correlating the received signal with the same spreading sequence. CDMA has been shown to provide robust performance against multi-path fading and multi-user interference given an appropriate control of user transmitting power. In this thesis, applications of adaptive coding techniques in a CDMA network are investigated.

Another central piece of data link control protocols is the automatic repeat request (ARQ) retransmission protocol. In a basic ARQ protocol, the information sequence is encoded with an error detecting code. If the receiver decodes the received frame and finds no errors, it sends an acknowledgment (ACK) to the transmitter, and the transmitter clears the frame out of the buffer. If the receiver detects error(s) in the received frame, it sends a negative acknowledgment (NAK) to the transmitter, and the transmitter resends the frame. In this thesis, NAK's and ACK's are assumed to be error free.

ARQ schemes are usually divided into three types: selective-repeat, Go-back-N, and Stop-and-Go according to the windowing techniques [13]. The ARQ schemes considered in this thesis are of selective-repeat type, i.e., the source keeps transmitting a stream of frames and only retransmits those frames that are in error. Go-back-N and Stop-and-Go schemes are not considered here because of their low throughput and large delay. For selective-repeat protocols, the receiver must have a large buffer in which it can store out-of-order frames and frames waiting for retransmission. In this thesis, we assume that the buffer has infinite capacity.

For wired communications, ARQ schemes use an error detecting code to detect errors in the data. Although these schemes may not employ FEC coding, they still provide reliable services because of the benign nature of the channel. In a fading channel, there are often deep fades which attenuate signals way below the average signal strength and cause errors. For even moderate frame lengths, the probability of not having deep fades in a frame duration is usually small, and the probability of frame error will be close to one. Thus selective-repeat ARQ schemes without FEC coding are ineffective and wasteful of energy and bandwidth resources over wireless channels.

In this thesis, we are concerned with the joint design and analysis of ARQ protocols and adaptive FEC coding schemes in wireless systems. With powerful FEC codes, the frame errors over wireless channels can be reduced such that ARQ protocols will function properly. With ARQ protocols, multiple transmissions for an information frame provide a very natural way of rate adaptation for FEC coding.

1.2 Previous Work

1.2.1 Advances in Coding

In the last decade, the coding community has benefited tremendously from the discovery of turbo codes and the iterative decoding principle. In 1993, Berrou *et al* proposed the turbo codes which achieve performance close to the Shannon channel capacity over AWGN channels [12]. A turbo code is a parallel concatenated code whose encoder is formed by two constituent systematic convolutional encoders joined by an interleaver. The basic concept of turbo decoding is that for each constituent code the corresponding decoder estimates the likelihood of the message based on a fragment of the received signal, then through sophisticated algorithms, the decoders exchange information. This process repeats iteratively and the estimation of the message becomes more and more reliable. After a number of iterations, a decision for each information bit is made based on the likelihood of that bit calculated by both decoding modules.

Inspired by the excellent performance of turbo coding, various iterative decoding algorithms have been investigated for interleaved codes. Serial concatenated convolutional and block codes [11] are constructed by concatenating codes in a serial fashion. Gallager's low density parity check (LDPC) codes [34], after being dormant for more than twenty years, were rediscovered independently by MacKay and Neal [54] and Wiberg [84]. Product codes [40, 67], a special case of block turbo codes, are two dimensional codes constructed from small component codes. High rate product codes have been constructed and shown to perform close to the Shannon channel capacity. In [57], a rate 0.98 Hamming product code $(1023, 1013)^2$ was shown to achieve a BER of 10^{-5} within 0.27 dB of capacity.

Designing rate compatible codes is of great practical interest. A rate compatible code family is a collection of codes such that the code bits of any codeword from a high rate code are contained in a codeword of a lower rate code. Hence, a stronger code can be derived from a weaker code by adding incremental redundancy. Rate compatible codes also facilitate the use of a single decoder for the decoding of codes of all rates. Rate compatible codes can be obtained by puncturing a low rate code. Puncturing is a process of increasing the code rate by deleting some of the redundant parity check symbols before transmission.

The earliest rate compatible codes trace back to Mandelbaum's rate compatible Reed-Solomon codes [55]. An (N, K) Reed-Solomon code has K information symbols and N coded symbols. This code is capable of correcting t errors plus e erasures, as long as $2t + e \leq N - K$. Applications of Reed-Solomon (RS) codes range from audio CD, NASA deep space communications systems [59, 60], SINGARS military radios [46], to FEC for wavelength division multiplexing (WDM) systems at 2.5 Gb/s [83]. Rate compatible RS codes can be obtained by puncturing the codewords of the (N, K) code to reduce the number of code symbols to N'. The resulting code is capable of correcting t errors plus e erasures, as long as $2t + e \leq N' - K$.

Punctured convolutional codes are widely adopted in 3G cellular standards for rate matching and unequal error protection. A convolutional code can be considered as linear filtering over the binary field which introduces redundancy in the original input sequence. A convolutional encoder is implemented with a shift register, where one information bit is taken as the input every shift register clock cycle and more than one bit is generated as the output. Unlike block codes, convolutional codes map an arbitrarily long input message sequence to an arbitrarily long code stream without block structure. Punctured convolutional codes are obtained by periodically puncturing code symbols from the output of a low rate $R = 1/N$ convolutional code, and the decoding complexity using Viterbi algorithm is about the same as the decoding of the original code [21]. Good punctured convolutional codes have been reported in [20] for rate matching and unequal error protection. Rate compatible punctured convolutional codes were discovered by Hagenauer in [39] and extended by Lee in [51] for applications in type-II hybrid ARQ schemes.

Similar to punctured convolutional codes, punctured turbo codes have also been studied for rate matching and unequal error protection [1, 7, 45]. In [7], turbo codes were punctured to provide two level of error protection. In [45], a few puncturing patterns were compared for turbo codes, and the resulting codes were simulated over AWGN and Rayleigh fading channels. In [1], several 16-state punctured turbo codes of length 10000 bits and rate above 1/2 were designed to achieve a BER of 10^{-5} within 0.75 dB from the channel capacity limits. Punctured turbo codes with BPSK modulation do not have phase tracking difficulties, as opposed to higher order modulation schemes that achieve similar spectral efficiency. Rate compatible punctured turbo codes for type-II hybrid ARQ schemes were investigated by Rowitch and Milstein in [71, 73].

The substantial variation of channel conditions in wireless systems has given rise to a large interest in adaptive coding techniques based on channel estimation [2, 3, 4, 35, 36, 42]. The basic idea is to estimate the channel condition at the *transmitter*, then to adapt the code rate, modulation size, and signal power accordingly. Although the well known result of information theory [33] states that knowledge of the channel condition at the transmitter does not improve single-user, memoryless channel capacity, substantial gains over fixed-rate systems have been demonstrated under the assumption of perfect or very reliable channel estimation [2, 3, 42]. However, engineers in practice are concerned that accurate channel estimates are usually hard to obtain, and poor estimates may substantially degrade system performance. Most recently, the robustness of adaptive trellis coded modulation with outdated estimates has been taken into consideration [35] and up to 1.5 dB gain was achieved over fixed-rate systems of the same complexity. When the correlation between each estimate updates decreases, the outdated estimate becomes useless and this adaptive system fails to provide any gain.

1.2.2 Advances in ARQ

To improve the performance over wireless channels, the basic ARQ schemes are enhanced by first encode the information sequence with an error detecting code, then encode the sequence again with an FEC code [85, 90]. These type of ARQ schemes are called Type-I hybrid ARQ schemes. The receiver first tries to correct the errors in the received frame, then checks if there are any uncorrectable (but detectable) errors in the decoded frame with the error detecting code. If errors are detected, the receiver throws away the erroneous frame and sends a NAK to the transmitter to request a retransmission. In severe channel conditions, however, as poor conditions persist, even very powerful FEC codes could fail repeatedly. More robust and efficient modified ARQ protocols which adapt the code rate according to the channel condition need to be developed. If the channel condition can be easily obtained at the transmitter, the adaptive coding techniques described in the previous section can be applied to an ARQ system.

When good channel estimations are not available, rate adaptation can be effectively achieved by metric combining of fixed-rate codes [23]. In metric combining type-I hybrid ARQ schemes, when a decoding error occurs, the frame is retransmitted. The receiver then averages the demodulator output metrics from all received copies of the frame and attempts to decode with the combined frame. The basic idea behind metric combining is to improve the effective channel SNR by combining multiple copies of a frame. The effectiveness of code combining schemes with Reed-Solomon, BCH, and convolutional codes has been investigated over fading channels in numerous papers [5, 14, 41]. Since the same frame is sent over the channel multiple times, a metric combining type-I hybrid ARQ is equivalent to a rate adaptive repetition coding scheme from the coding point of view.

In Figure 1.2, the block diagram of an incremental redundancy retransmission ARQ system is shown. These type of ARQ schemes are called type-II hybrid ARQ schemes. The transmitter is composed of a cyclic redundancy check (CRC) error detecting code encoder, a rate compatible FEC encoder, an interleaver, and a modulator. The receiver consists of a demodulator, a deinterleaver, a rate compatible FEC decoder, and a CRC decoder.

The source bits are grouped into sequences of length L_i, and each sequence is encoded by a CRC encoder to form a length $L_i + L_{CRC}$ bit sequence with L_{CRC} parity check bits. The encoding algorithm and examples can be found in [79]. A well designed CRC code can guarantee very small undetected error probability with only a small number of parity check bits (16 bits in ANSI and CCITT standards) [87, 89]. In our analysis, we assume that the undetected error probability is negligible. The overhead due to CRC is also neglected because L_{CRC} is usually relatively small compared with L_i and it is fixed for all ARQ schemes under consideration.

The CRC encoded sequence is further encoded with the lowest rate code from a family of rate compatible FEC codes. The number of information bits of the rate compatible code family equals $L_i + L_{CRC}$. In the initial transmission, a fragment of the codeword, which forms a codeword of the highest rate (weakest) code in the

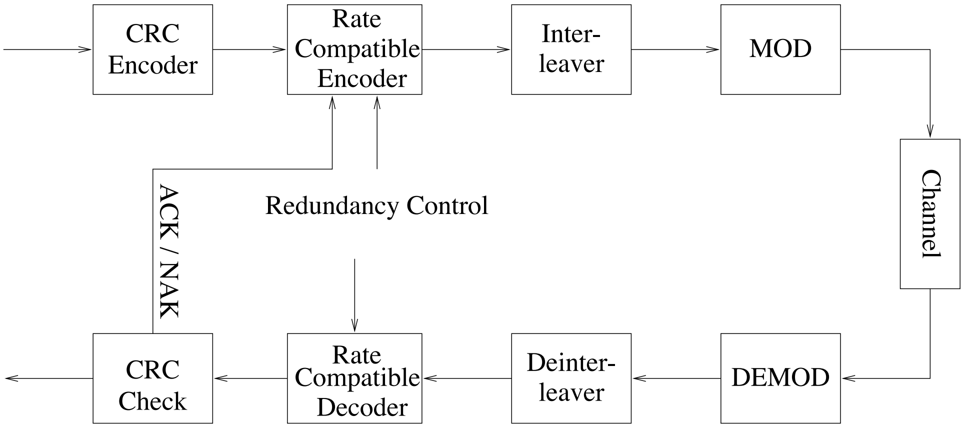


Figure 1.2: Block diagram of an incremental redundancy retransmission system

code family, is transmitted through the channel. The receiver first performs error correcting with the rate compatible FEC decoder, then performs the CRC check to detect uncorrectable but detectable errors. When error is detected, the receiver sends a NAK to the transmitter. Upon the receipt of NAK from the receiver, incremental redundancy is transmitted to aid decoding at the receiver. After each transmission, the effective code rate is lowered and thus a stronger code is obtained. This procedure is repeated until a successful decoding occurs or decoding fails after all code bits are received. In the second case, the receiver throws away all previous bits and the protocol restarts. The receiver also has the option of restarting the protocol and combining multiple copies of code sequences with metric combining.

An interleaver is used to break bursts of errors caused by the channel to enhance the performance of FEC codes, which can effectively correct random errors. An input sequence to the interleaver is reordered so that adjacent positions in the input sequence are far apart in the output sequence. For a well designed interleaver, a burst of errors in the channel will be spread into random errors throughout the deinterleaved sequence.

In [55], Mandelbaum suggested that a punctured Reed-Solomon codeword be transmitted as the primary codeword and the remaining parity bits be used in retransmissions as incremental redundancy blocks. In the original convolutional code based type-II hybrid ARQ scheme proposed by Lin and Yu [52], a rate 1/2 convolutional code of length $2K$ is decomposed into two rate 1 code sequences. The transmitter sends K bits each time by alternating between the two rate 1 code sequences, and the receiver attempts to recover the information from each sequence separately, and on failure decodes the combined codeword jointly. ARQ schemes based on rate compatible punctured convolutional codes have been further studied for AWGN channels [58] and Rayleigh fading channels [32, 27, 53]. To achieve even higher bandwidth efficiency, rate compatible punctured trellis coded modulation was proposed [31] and applied to a type-II hybrid ARQ scheme. It was observed to have substantial gain over a fixed-rate system.

Rate compatible punctured turbo codes were proposed by Rowitch and Milstein [71, 73] for type-II hybrid ARQ schemes. The resulting ARQ scheme is observed to have superior throughput performance compared to ARQ schemes based on rate compatible punctured convolutional codes. The spectral efficiency is shown to be above the computational cutoff rate bound.

The performance of modified ARQ schemes has been analyzed for limited channel models. For ARQ schemes based on convolutional and turbo codes, since the analysis of exact packet error rate is still unknown, error bounds on the packet error rate have to be used. For convolutional codes, the error bounds are relatively tight, and the performance analysis gives a relatively good approximation [16]. For turbo codes, the union bounds and the Gallager bounds diverge at low signal-to-noise ratio [9, 30], the tangential sphere bounds do not diverge but are not valid bounds for any specific codes [74]. Hence, simulation is the only effective way to study turbo code based

ARQ systems.

For block code based ARQ schemes, modified ARQ schemes have been analyzed for memoryless channels because the calculation of the exact error probability is possible. Type-II hybrid ARQ schemes based on maximum distance separable (MDS) codes have been studied for AWGN channels in [86], for synchronous frequency-hopped spread spectrum multiple access systems in [17], for frequency-hopped partial-band interference channels in [25], and for meteor-burst channels in [64, 65]. An (N, K) MDS code has minimum distance N-K+1. The most popular MDS codes used in industry are Reed-Solomon codes. In [17, 25, 86], the analysis is limited to systems with a small number of redundancy blocks; otherwise, it is numerically intractable.

1.3 Main Contributions and Overview of Thesis

In this thesis, we offer an analytical framework for the performance evaluation of general type-I and type-II hybrid ARQ protocols over finite-state channels. As an example, an analysis of Reed-Solomon code based systems is shown to yield insight into the effect of the protocols, channel parameters, design parameters, and decoder implementations on the system performance. We also investigate the construction of powerful rate compatible product codes for use in adaptive coding. We then evaluate the performance of adaptive turbo coded ARQ schemes over a multi-user DS-CDMA network. The thesis consists of three self-contained chapters focusing on different aspects of adaptive coding with ARQ.

In Chapter II, we present the analysis of MDS codes based ARQ protocols over correlated fading channels. We first review previous work and point out the lack of analytical methods to evaluate ARQ protocols with sophisticated rate adaptive

FEC codes. Then, we introduce the system and accompanying models for coding, signaling, and fading channels. The fading process is modeled as a fast Rayleigh fading random process superimposed on a slowly varying shadowing process. The shadowing process is modeled as a finite-state Markov process for the convenience of analysis.

Error recursion techniques are applied to three ARQ protocols based on Reed-Solomon codes. A fixed-rate type-I hybrid ARQ protocol is used as the baseline system for comparison. Then, we define a genie-aided rate-adaptive type-I hybrid ARQ protocol where a genie provides perfect channel information to the transmitter, and the transmitter chooses the optimal code from a family of rate compatible FEC codes based on the channel condition. The performance of this scheme provides an upper bound for practical systems with imperfect channel estimation. The third candidate protocol is a type-II hybrid ARQ schemes with incremental redundancy retransmission.

An error recursion approach is proposed to analyze the throughput, delay, and energy efficiency performance of ARQ schemes. Following the derivation of the error recursion algorithm, numerical results are shown with a few practical examples. As expected, the rate-adaptive type-I hybrid ARQ scheme offers significant performance improvement over the fixed-rate type-I hybrid ARQ. More interestingly, the type-II hybrid ARQ, without any channel information besides NAK/ACK feedback for each frame, is shown to outperform the rate-adaptive type-I hybrid ARQ with perfect channel information. The effect of channel memory and soft decision decoding on the performance of ARQ schemes is also accurately predicted in our analysis.

The numerical results can be interpreted first as a demonstration of the power of the error recursion approach in solving adaptive coding performance evaluation problems. Furthermore, the results directly answer a very interesting question: how does event-triggered rate adaptation (type-II hybrid ARQ) compare with measurement-triggered rate adaptation (rate-adaptive type-I hybrid ARQ with perfect channel side information)? The answer is that the inherent rate adaptation mechanism of type-II hybrid ARQ offers better throughput and energy efficiency than measurement-based rate-adaptive type-I ARQ at the expense of a larger buffer size requirement.

In Chapter III, a family of rate compatible multiple product codes[1] are constructed. Product codes are two dimensional codes constructed from small component systematic block codes. Recent developments in iterative decoding show that high rate product codes can achieve performance close to Shannon capacity. In Chapter III, we first review the literature and introduce the advances in iterative decoding algorithms. The need and difficulty of decoding low rate product codes are also discussed. Preliminaries of product codes and an introduction to the iterative decoding principle are discussed. In the discussion of decoding algorithms, we show that the complexity of a MAP decoding algorithm based on parity check trellis grows exponentially with the number of parity check bits in a block code. Hence, lowering the code rate of a product code by extending the component code is not a practical option.

We develop a simple construction of low rate *multiple* product codes based on the original two dimensional product code. In an M-dimensional multiple product code each information bit is encoded by M high rate component codes. Because high rate component codes are used, the decoding complexity of the new codes only grows linearly with the length of the code and the dimensionality of the code. The

[1] These codes are not product codes in strict sense but instead are parallel concatenated block codes. However, since they are constructed from the original product codes and share many features of product codes, they will be referred to as rate compatible multiple product codes or simply rate compatible product codes in this thesis

weight enumerators and union bounds are derived for multiple product codes based on the uniform random interleaver argument [9].

Numerical evaluation of the union bounds and simulations results are provided to assess the performance of product codes over AWGN and Rayleigh fading channels. The union bounds are shown to be tight at high SNR, which implies that the iterative decoding algorithm is a good approximation of the maximum likelihood decoding algorithm. The resulting product codes of rates 0.72, 0.63, and 0.57 are shown to outperform the S-random interleaved turbo codes of the same rate, length, and comparable complexity for a target BER less than or equal to 10^{-4}. Finally, the rate compatible codes are put into the framework of an adaptive ARQ scheme and the throughput performance is evaluated via simulations.

Since the adaptive coding schemes discussed in this thesis are most likely to be used in a multi-user system, in Chapter IV, we apply a few rate adaptive ARQ protocols to a more practical wireless network setting and examine their performance. The performance metrics include overall network throughput, average delay, and energy efficiency. We also modified some published rate compatible punctured turbo codes to make them more suitable for practical applications.

For multi-user systems, the interference level increases with the number of users when non-orthogonal multiple access is employed. More users implies larger offered load to the network, but also means a larger interference level. Intuitively, there exists an optimal number of users, which maximizes the total network throughput for a fixed-rate ARQ system. For rate adaptive systems, we would also like to exploit the optimal call admission policy to maximize the total network throughput. Since delay is one of the major concerns in data networks, it will be considered as a critical benchmark in the performance evaluation. For a mobile unit depending on batteries, energy consumption determines the lifetime of the unit, hence energy efficiency is also evaluated for low-power communication systems.

In Chapter IV, we first introduce the system model of an asynchronous direct-sequence CDMA network and the mathematical model for the signaling scheme. Then, we present the structure of punctured turbo codes, the retransmission protocol, and the complexity analysis. The ARQ protocols under consideration are turbo coded type-I hybrid ARQ, metric combining type-I hybrid ARQ, and type-II hybrid ARQ schemes.

Simulation results over lognormal shadowing and frequency-selective fading channels are presented. Compared with fixed-rate $\frac{1}{3}$ type-I hybrid ARQ, and metric combining type-I hybrid ARQ schemes, the type-II hybrid ARQ schemes are shown to provide a significant increase in normalized throughput at the cost of additional storage requirements and larger delay. The redundancy block size of the type-II hybrid ARQ protocol is shown to be a critical design parameter for delay performance. The maximum achievable throughput under energy constraints is also investigated. The optimal call admission policy and signal power to meet the quality of service can be made based on the performance evaluation in this chapter.

In Chapter V, we summarize the findings and discuss future topics.

CHAPTER II

Rate Compatible Reed-Solomon Codes

Severe channel impairments in wireless communication systems make coding and retransmission schemes a necessity for data communications networks. In this chapter, we propose an error recursion approach to analyze the throughput, delay, and energy efficiency performance of Reed-Solomon coded ARQ schemes over correlated fading channels. The performance of type-II hybrid ARQ protocols with incremental redundancy retransmission (IRR) is compared with the performance of conventional type-I hybrid ARQ and rate-adaptive (RA) type-I hybrid ARQ schemes. As expected, the RA type-I hybrid ARQ scheme optimized for throughput yields significant performance improvement over the fixed-rate type-I hybrid ARQ. More interestingly, the IRR type-II hybrid ARQ, without any channel information besides NAK/ACK feedback for each frame, is shown to outperform the RA type-I hybrid ARQ with perfect channel information. The effect of channel memory and soft decision decoding on the performance of ARQ schemes is also accurately predicted in our analysis.

2.1 Introduction

Reed-Solomon codes have been widely used as error correcting codes because of their burst error correcting capability and fast algebraic decoding which yields efficient decoder implementation. Applications of Reed-Solomon codes range from audio CD, NASA deep space communications systems [59, 60], SINGARS military radios [46], to FEC for wavelength division multiplexing (WDM) systems at 2.5 Gb/s [83].

Hybrid ARQ schemes based on Reed-Solomon codes have been studied intensively during the last two decades [50, 44, 66, 25, 64, 69, 86, 85]. In a standard ARQ protocol, if errors are detected in a received frame, the packet is discarded, and a retransmission request is sent to the transmitter. Type-I hybrid ARQ schemes provide more protection than basic ARQ schemes against poor channel conditions by encoding the information sequence with both error detecting and error correcting codes [90, 85]. The receiver first tries to correct the errors in the frame, then checks if there are any uncorrectable (but detectable) errors in the decoded frame with the error detecting code. The transmitter is not restricted to encode the frame with the same FEC code for every transmission. In a rate-adaptive type-I hybrid ARQ system, the code rate varies according to the channel quality [65, 69].

In type-II ARQ schemes, an FEC codeword is composed of a few complementary blocks, which are sent through the channel in multiple transmissions upon the receipt of NAK's from the receiver. In [55], Mandelbaum suggested that a punctured Reed-Solomon codeword be transmitted as the primary codeword and the remaining parity symbols be used in retransmissions as incremental redundancy blocks. Both Reed-Solomon codes and punctured Reed-Solomon codes are MDS codes. As described in the previous chapter, an (N, K) MDS code can correct up to $\lfloor \frac{N-K}{2} \rfloor$ errors, where $\lfloor a \rfloor$ is the largest integer that is no greater than a.

Performance of Reed-Solomon code based type-II hybrid ARQ schemes has been analyzed over a variety of memoryless channels. In [86], a MDS code based type-II hybrid ARQ scheme over AWGN channels has been analyzed by Wicker and Bartz using the generating function method. An (n, k) MDS code was decomposed into a pair of $(n/2, k)$ punctured MDS codes. The original code and the two derived codes were used to form a type-II ARQ protocol. The ARQ protocol can be represented by a graph, which can be analyzed with the generating function method. However, this approach is numerically intractable for graphs with a large number of nodes. The generating function approach has also been used to analyze a frequency-hopped partial-band interference channel by Daraiseh *et al* in [25]. A frequency-hopped spread spectrum multiple access communication system with type-II hybrid ARQ was analyzed by Bigloo *et al* in [17]. In [17], the authors derived the performance for the case where a Reed-Solomon codeword is broken into three higher rate subcodes for successive transmissions. For more than three subcodes, this approach is also numerically intractable.

In [64, 65], the performance of a type-II hybrid ARQ, a fixed-rate type-I hybrid ARQ, and a rate-adaptive type-I hybrid ARQ were analyzed for meteor-burst communications. The analysis has a similar flavor to the work presented here in the sense that an error recursion is also used to analyze the probability of frame error for the type-II ARQ protocol. However, the meteor-burst channel is memoryless and has an exponentially decaying signal-to-noise ratio (SNR), which makes the error recursion significantly different.

The traditional approach of analyzing the performance of a rate adaptive system over channels with memory usually involves solving for the steady state probability of the joint channel state and code rate pairs. In [66], an adaptive-rate FEC (without ARQ) scheme for slow frequency-hop communications systems was investigated. In [69], a rate-adaptive type-I hybrid ARQ protocol was analyzed for slowly varying channels. The channel was assumed to remain the same for a time period that is one or more orders of magnitude greater than the frame transmission time and acknowledgment delay. Under this assumption, the throughput can be obtained by simply averaging the well known selective-repeat ARQ throughput for a fixed-rate code over the joint distribution of the channel state and code rate.

In this chapter, we use an error recursion technique to analyze the throughput, delay, and energy efficiency performance of punctured Reed-Solomon coded ARQ schemes. Error recursion techniques have been used for the study of RS coded systems in non-interleaved Rayleigh channels in [18], and for hybrid power control FEC systems over finite-state channels in [61, 24]. Applied to the problem of hybrid ARQ protocols, an error recursion helps to capture the dynamics of the *rate adaptation* and *channel variation* simultaneously. The accuracy of this model also allows us to quantitatively study the effect of block size and channel correlation on system performance.

The rest of the chapter is organized as follows. In Section 2.2, the system and finite-state channel model are introduced. In Section 2.3, a genie-aided rate-adaptive type-I hybrid ARQ over channels with lognormal shadowing is analyzed. This is followed by the description of an error recursion technique for an IRR type-II hybrid ARQ scheme. Then this error recursion technique is extended to the analysis of ARQ schemes based on soft-decision decoding. In Section 2.4, numerical results are presented and conclusions are given in Section 2.5.

2.2 System and Models

In this section the models for the transmitter, channel and receiver are described. As shown in Figure 1.2, the transmitter is composed of a CRC encoder, a rate compatible Reed-Solomon encoder, an interleaver, and a binary frequency-shift-keying (BFSK) modulator. Note that a symbol interleaver that matches the size of the Reed-Solomon code symbol is used instead of a bit interleaver. The receiver consists of a BFSK demodulator, a deinterleaver, a Reed-Solomon decoder, and a CRC decoder.

2.2.1 Coding and ARQ Systems

The source bit sequences are encoded with a CRC code for error detection. The transmitter is assumed to have infinite supply of source bits. The source bits are grouped into sequences of length L_i, and each sequence is encoded by a CRC encoder to form a length L_i+L_{CRC} bit sequence with L_{CRC} parity check bits. The CRC coded sequences are encoded by Reed-Solomon codes for forward error correcting. When a Reed-Solomon code with alphabet size $N = 2^m$ is used, the information sequence length $L_i + L_{CRC}$ bits is required to be a multiple of m, say, $L_i + L_{CRC} = Km$. Every m information bits are grouped into an RS symbol. Then each length K symbol information sequence is encoded into an extended Reed-Solomon codeword of length N. This codeword is punctured to form a family of rate compatible punctured RS codes with rates from K/N to 1.

For fixed-rate type-I hybrid ARQ protocols, one puncturing scheme with a certain code rate is selected and used by both the encoder and the decoder regardless of the channel condition. For rate-adaptive type-I hybrid ARQ protocols, a punctured code with an appropriate rate is selected from a family of (N_i, K) codes for each transmission based on either channel observation or acknowledgments. For type-II hybrid ARQ, the RS codeword is divided into a length K block and $\frac{N-K}{\Delta}$ size Δ redundancy blocks for successive transmissions.

The resulting punctured RS codewords are interleaved before transmission to achieve good burst-error correcting performance. We are going to use the fixed-rate type-I hybrid ARQ protocol as an example to explain the interleaving scheme. Consider an (N, K) RS code used with a block interleaver of N rows and D columns. A sequence of D codewords (DN code symbols) are written into the interleaver column by column. Then the code symbols are read out row by row. Thus, the j-th input symbol is written into the interleaver at the $j \bmod N$-th row and $\lfloor j/N \rfloor$-th column. Subsequently, this element is read out as the $(j \bmod N)D + \lfloor j/N \rfloor$-th element in the output sequence. It is easily shown that any two positions within distance N in the input sequence will be at least D positions apart in the output sequence. The interleaving depth, D, is designed to be long enough relative to the fading rate such that any two fading levels that are D symbol periods apart are essentially uncorrelated. The same interleaving scheme also applies to rate-adaptive type-I ARQ and type-II ARQ. The requirement on the interleaving depth, D, remains the same, and the number of rows in the interleaver, N, is required to be greater than or equal to the maximum number of symbols in each transmission for one information sequence.

The retransmission protocol for type-II hybrid ARQ is more complex and requires additional overhead compared to type-I ARQ schemes. For each information sequence, the transmitter first sends K information symbols. If the CRC determines that the word is received correctly, an ACK is sent back to the transmitter, and the transmitter discards the rest of the code symbols. Upon the receipt of a NAK from the receiver, the transmitter transmits a redundancy block with Δ new code symbols. For convenience of analysis, we assume that Δ is even and Δ divides $N - K$. In order for the receiver to put the incremental redundancy blocks into the correct position within a codeword, a subsequence number needs to be sent with each incremental redundancy block. This procedure is repeated until a successful decoding occurs or decoding fails after all N symbols are received. In the second case, the receiver throws away all previous symbols and restarts the protocol.

It is well known that for Reed-Solomon codes, the increment of the error correcting capability due to Δ more redundancy symbols is $\lfloor \Delta/2 \rfloor$. Upon the receipt of the j-th transmission, a $(K + (j-1)\Delta, K)$ punctured Reed-Solomon code is formed at the receiver and up to $(j-1)\Delta/2$ errors can be corrected. When a Reed-Solomon decoder operates in error-and-erasure mode, it corrects up to t errors and e erasures, as long as $2t + e \leq (j-1)\Delta$.

The modulator maps each RS symbol onto m binary FSK signals according to the natural binary mapping. Binary FSK is an effective modulation scheme for communications over fading channels, as used in SINGARS radios. In binary FSK, a pair of orthogonal sinusoidal waveforms are used to represent the data 0 and 1 as follows

$$\begin{aligned} s_0(t) &= \sqrt{2P} \sin 2\pi f_0 t \quad 0 \leq t \leq T_s \\ s_1(t) &= \sqrt{2P} \sin 2\pi f_1 t \quad 0 \leq t \leq T_s, \end{aligned}$$

where T_s is the symbol duration, and P is the signal power. The frequency f_0 and f_1 are chosen to satisfy the orthogonality between the two signals as follows

$$\int_0^{T_s} s_0(t) s_1(t) = 0.$$

Let $s(t)$ denote the transmitted signal. The received signal $r(t)$ is given by

$$r(t) = K(d)\beta(t)\alpha(t)s(t) + n(t), \tag{2.1}$$

where $K(d)$ is the path-loss due to the distance, d, between the transmitter and the receiver, $\beta(t)$ is a lognormal shadowing process, $\alpha(t)$ is a fast fading process, and $n(t)$

is an additive white Gaussian noise process of double-sided power spectrum density $N_0/2$. The detailed models for these parameters are discussed in the next section. The channel signal-to-noise ratio is given by

$$\gamma(t) = E[\beta^2(t)\alpha^2(t)]K^2(d)P/(N_0/2). \tag{2.2}$$

2.2.2 Shadowing Channel Model

In the fading model considered here, the transmitter receiver pair suffers from both fast fading, $\alpha(t)$, and lognormal shadowing, $\beta(t)$. A random variable is said to follow lognormal distribution if the logarithm of the random variable follows a Gaussian distribution. The mean of the shadowing process is the path loss $K(d)$, which is assumed to be a constant in our study. Since the shadowing process is usually very slow compared to the data rate, the shadowing is assumed to be constant over each transmission (frame) $\beta(t) = \beta$. The correlation between the logarithm of the shadowing levels of adjacent frames is modeled as an exponentially decaying function [37]. Suppose the mobile velocity is v m/s, the normalized correlation, $R(T)$, of the shadowing levels measured at T seconds apart is given by

$$R(T) = \sigma_s^2 \epsilon_D^{vT/D}$$

where σ_s is the standard deviation of the shadowing level, and ϵ_D is the correlation between two points separated by distance D. According to the 1700 MHz measurement in an urban area [37], σ_s was estimated to be 4.3 dB, $\epsilon_{10} = 0.3$. Our numerical analysis uses the parameters measured in [37].

Under the assumption that the acknowledgment delay is constant and much longer than the frame transmission time, as it is in most practical systems, the distance that a mobile travels between the transmission of a frame and its immediate retransmission is roughly constant. Hence, the correlation of the shadowing process between retransmissions is constant regardless of the redundancy block size.

The fast fading process, $\alpha(t)$, is modeled as a constant flat Rayleigh fading process over each symbol and independent from symbol to symbol. This model is accurate under the assumption of narrowband signaling and sufficiently large interleaver size. The pdf of α follows a Rayleigh distribution, which is given by

$$p_\alpha(a) = \frac{a}{\sigma^2} e^{-a^2/2\sigma^2}, \tag{2.3}$$

where σ^2 is usually normalized to be 0.5 so that $E(\alpha^2) = 1$. For the channel model described in this section, the signal-to-noise ratio, γ, as defined in (2.2) can be rewritten as

$$\gamma = \frac{2K^2(d)P}{N_0} \beta^2. \tag{2.4}$$

2.2.3 First-order Markov Model for Shadowing Channel

The time varying nature of the shadowing process can be captured by Markov models. If we quantize the channel gain, β in dB, into L levels $\{u_1, u_2, \cdots, u_L\}$, we can approximate the shadowing channel with an L-state first-order Markov model. Since β follows a Gaussian distribution in dB, a Lloyd-Max quantizer for Gaussian source is employed to determine the reconstruction levels $\{u_l\}$, and decision regions $\{c_l\}$. The quantization function $Q(x)$ is given by

$$Q(x) = u_l \quad x \in c_l.$$

Lloyd-Max quantizers are designed to minimize the mean squared quantization error [75].

The steady state probability of the channel be in state u_l can be obtained by integrating the Gaussian density $p(x)$ over c_l,

$$P_U(u_l) = \int_{c_l} p(x)dx \tag{2.5}$$

where $p(x)$ is given by

$$p(x) = \frac{1}{\sqrt{2\pi\sigma_s^2}} \exp(-\frac{x^2}{2\sigma_s^2})$$

where σ_s^2 is specified in [37].

Consider two random variables corresponding to the values of the logarithm of the shadowing process at time t and $t-T$, say, Y and Z. The joint pdf of correlated Gaussian random variables is

$$p_{Y,Z}(y,z) = \frac{1}{2\pi\sigma_s^2\sqrt{1-\rho^2}} \exp\{-\frac{y^2 - 2\rho yz + z^2}{2(1-\rho^2)\sigma_s^2}\} \tag{2.6}$$

where ρ is defined as

$$\rho \stackrel{\triangle}{=} R(T)/\sigma_s^2. \tag{2.7}$$

The state transition matrix can then be easily obtained using

$$P_U(u_l|u_n) = \frac{\int_{c_l}\int_{c_n} p_{Y,Z}(y,z)dzdy}{\int_{c_n} p(z)dz}. \tag{2.8}$$

2.2.4 Demodulation and Symbol Error Rate

In this section, the demodulator and the corresponding RS code symbol error performance are described. When the signal is modulated with a carrier signal, the carrier phase is often unknown at the receiver. The phase ambiguity is either due to the phase change during the propagation or because the oscillators at the receiver and transmitter are not synchronized. Noncoherent demodulators offer robust error performance with no need for carrier phase estimation. For BFSK, the noncoherent demodulator can be implemented with either square-law detector or envelope detector [19]. For a Reed-Solomon symbol that consists of m BFSK signals, all m signals undergo the same channel attenuation due to the symbol interleaver. The fading process is assumed to be a constant α over one symbol period, and independent from symbol to symbol. The BFSK demodulation bit error probability conditioned on the fading level α is given by [63]

$$p_b(\alpha) = \frac{1}{2} e^{-\frac{\alpha^2 E}{2N_0}}. \tag{2.9}$$

The probability of symbol error conditioned on the fading level α is given by

$$\begin{aligned} p_s(\alpha) &= 1 - (1 - p_b(\alpha))^m \\ &= 1 - \sum_{l=0}^{m} \binom{m}{l} 1^{m-l} (-p_b(\alpha))^l \\ &= \sum_{l=1}^{m} \binom{m}{l} (-1)^{l+1} 2^{-l} e^{-\alpha^2 \frac{lE}{2N_0}}. \end{aligned}$$

When we average $p_e(\alpha)$ over the distribution of α, as given by (2.3), the resulting symbol error probability is

$$p_s = \sum_{l=1}^{m} \binom{m}{l} \frac{(-1)^{l+1} 2^{-l+1}}{2 + \frac{lE}{N_0}}. \tag{2.10}$$

As mentioned before, a Reed-Solomon decoder is more powerful when the demodulator could erase unreliable symbols. In this chapter, the parity-check method [66] is used to decide which symbol should be erased. For a Reed-Solomon symbol composed of m BFSK symbols, one even parity-check bit is appended to it. If the demodulated $m+1$ bits do not have even parity, the symbol is erased. Other methods of generating erasures include Viterbi's RTT (ratio threshold test) [81] and Baysian detection method [8], etc.. The analysis developed in this work can also be extended for these erasure generating techniques.

For systems generating symbol erasures, we first calculate the probability of erasure and symbol error conditioned on channel attenuation α, then average them over the fading distribution. The resulting probability of erasures p_e, and probability of symbol errors p_s are given by

$$p_e = \int \sum_{l=1;\ l \text{ odd}}^{m+1} \binom{m+1}{l} p_b(a)^l (1 - p_b(a))^{m+1-l} p_\alpha(a) da, \tag{2.11}$$

$$p_s = \int \sum_{l=2;\ l \text{ even}}^{m+1} \binom{m+1}{l} p_b(a)^l (1 - p_b(a))^{m+1-l} p_\alpha(a) da. \tag{2.12}$$

The closed-form solution for p_e and p_s exist, but since the solution is a summation of an alternating sequence, numerical evaluation would lead to precision problems. Therefore, we numerically integrate (2.11) and (2.12) to obtain more accurate results.

2.3 Performance Analysis

In this section, we will analyze the throughput, delay, and energy efficiency performance of three ARQ protocols based on RS codes. We first derive the performance for hard decision decoders, then extend the results to error-and-erasure decoders.

2.3.1 Rate-Adaptive Type-I Hybrid ARQ

Consider a rate-adaptive type-I hybrid ARQ scheme where a genie tells the transmitter the channel condition for each transmission. The transmitter then chooses an appropriate code to maximize the instantaneous throughput based on the channel condition. The performance of this scheme provides an upper bound for practical systems with imperfect channel estimation. We will first describe the code selection algorithm, then derive the performance over correlated shadowing channels.

2.3.1.1 Optimal Code Selection

The code selection algorithm is designed to maximize the throughput for each transmission. The code candidates are chosen from a family of codes with rate compatibility in the sense that a higher rate weaker code is a prefix of a lower rate stronger code. Rate compatible codes facilitate the use of a single decoder for the decoding of codes of all rates. Assume the information frame has K symbols over $GF(2^m)$, then the code family is given by $\{\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_f\}$, where $\mathcal{C}_i$ is an (N_i, K) punctured Reed-Solomon code, and $N_1 = K, N_2 = K + 2, \cdots, N_i = K + 2 * (i - 1), \cdots, N_f = 2^m$. For the convenience of analysis, K is assumed to be an even number. In the following example, a family of rate compatible Reed-Solomon codes is obtained by puncturing an (64,16) extended Reed-Solomon code over $GF(2^6)$.

Consider the transmission of an (N_i, K) code $\mathcal{C}_i$ over a channel in state $U = u_l$. The throughput, $\eta(u_l, \mathcal{C}_i)$, is defined as the expected number of correctly received information bits per channel use as given by

$$\eta(u_l, \mathcal{C}_i) = \frac{K}{N_i} P_c(i, m), \tag{2.13}$$

where $P_c(i, m)$ is the probability of correctly decoding a codeword of code $\mathcal{C}_i$ over channel in state u_l. A larger code rate, $\frac{K}{N_i}$, usually implies smaller probability of correct decoding $P_c(i, m)$. The trade off between the code rate, $\frac{K}{N_i}$, and the probability of correct decoding, $P_c(i, m)$, is evident in (2.13).

The probability of correctly decoding a codeword of $\mathcal{C}_i$ depends on the error correcting capability of the code, $t_i = (N_i - K)/2$, and the symbol error probability of the demodulator output. For the channel models described in the previous section, the symbol errors are independent, identically distributed within a frame. Let $p_{s|l}$ denote the symbol error probability over a channel in state u_l, which can be obtained from (2.10) for flat Rayleigh fading channels. Then $P_c(i, m)$ can be written as

$$P_c(i, m) = \sum_{l=0}^{t_i} \binom{N_i}{l} p_{s|l}^l (1 - p_{s|l})^{N_i - l}.$$

The optimal code $\mathcal{C}_l^*$ for channel state u_l is given by

$$\mathcal{C}_l^* = \arg \max_{\mathcal{C}_i : i \in \{1,2,\cdots,f\}} \eta(u_l, \mathcal{C}_i). \tag{2.14}$$

For a given channel state u_l, the code selection algorithm is described by (2.14).

To illustrate the code selection algorithm, the code lengths of the optimal $(N_i, 16)$ RS codes are shown in Figure 2.1 for a range of channel SNR's. The resulting throughput of the type-I hybrid ARQ protocol with optimal rate over flat Rayleigh fading channels is shown in Figure 2.2.

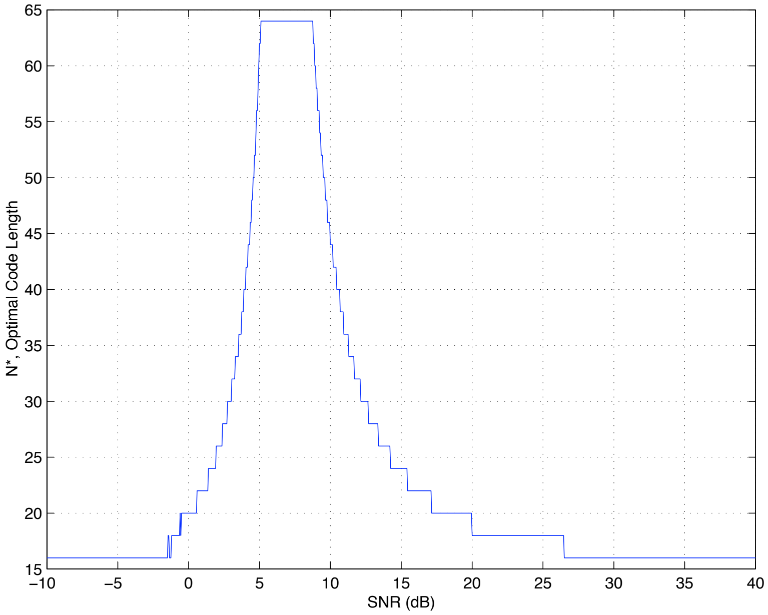


Figure 2.1: Optimal code lengths of the punctured $(N_i, 16)$ Reed-Solomon codes over flat Rayleigh fading channels

For channel SNR above 27 dB, the rate-1 code is selected since no error protection is needed as shown in 2.1. As the channel quality degrades, more and more parity checks are added to the codeword to protect the transmission from channel corruption. When the channel SNR drops to 6.5 dB, the lowest rate code becomes the optimal code. When the channel SNR goes below 5 dB, the additional error protection provided by the lowest rate code, i.e., the (N, K) code, over the next lowest rate code, i.e., the $(N-2, K)$ code, can no longer offset the smaller code rate of the (N, K) code. Thus, the optimal code rate starts to increase. When the channel SNR goes below -2 dB, the rate 1 code again becomes the optimal code selection because

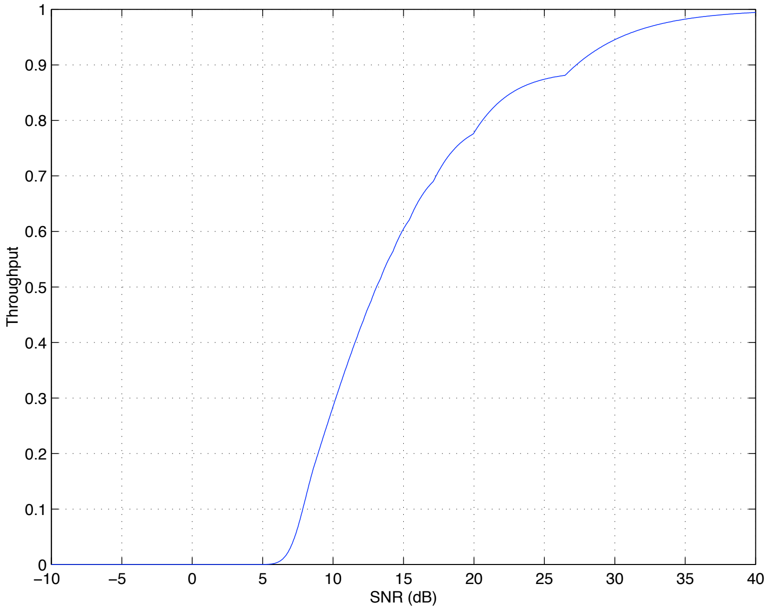


Figure 2.2: Throughput of type-I hybrid ARQ schemes with the optimal $(N_i, 16)$ punctured Reed-Solomon codes over flat Rayleigh fading channels

codes of all rates result in equally poor error performance.

2.3.1.2 Error Recursion

Now we proceed to analyze the delay, throughput, and energy efficiency performance of the rate adaptive type-I ARQ protocol with perfect channel information over correlated fading channels. The optimal rate adaption algorithm is described in the preceding section, and the first-order Markov model for the shadowing channel is described in Section 2.2.3. For a given set of parameters (pass-loss $K(d)$, transmitted signal power P, and background noise power spectrum density $N_0/2$), the channel SNR for each transmission varies with the state of the shadowing process according to (2.4).

The decoding error probability is recursive computed for all transmissions associated with the same information sequence. Let X_j denote the number of symbol errors in the j-th transmission, let t_j denote the error correcting capability of the code selected for the j-th transmission, and let U_j denote the channel state for the j-th transmission. Define F_j as the event of decoding failures of all of the first j transmissions. The joint probability of channel being in state u_l during the j-th transmission and event F_j is true, $P_l(F_j)$, is defined as

$$\begin{aligned} P_l(F_j) &\triangleq P(X_j > t_j, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1, U_j = u_l) \\ &= P(X_j > t_j | X_{j-1} > t_{j-1}, \cdots, X_1 > t_1, U_j = u_l) \\ &\cdot P(X_{j-1} > t_{j-1}, \cdots, X_1 > t_1, U_j = u_l) \end{aligned}$$

Note that the number of symbol errors in the j-th transmission, X_j, is independent of the previous $j-1$ transmissions conditioned on the channel state of the j-th transmission. The channel state of the j-th transmission, U_j, is also independent of the success or failure of the previous $j-1$ transmissions conditioned on the channel state of the $j-1$-th transmission. It follows that

$$P_l(F_j) = P(X_j > t_j | U_j = u_l) \sum_{n=1}^{L} P(U_j = u_l | U_{j-1} = u_n) P(F_{j-1}, U_{j-1} = u_n). \quad (2.15)$$

For a channel state u_l, the optimal code $\mathcal{C}_l^*$ is selected by the transmitter according to (2.14). The corresponding code length and error correcting capability are N_l^* and t_l^*, respectively. It follows

$$P(X_j > t_j | U_j = u_l) = P(X > t_l^* | U = u_l)$$

Denoting $P(X > t_l^* | U = u_l)$ by Q_l, it follows

$$Q_l = \sum_{i=t_l^*+1}^{N_l^*} \binom{N_l^*}{i} p_{s|l}^i (1 - p_{s|l})^{N_l^* - i} \quad (2.16)$$

where the symbol error probability over channel in state m, $p_{s|l}$, is given by (2.10). With this simplification, (2.15) can be rewritten as an error recursion for $P_l(F_j)$ with respect to j

$$P_l(F_j) = Q_l \sum_{n=1}^{L} P_n(F_{j-1}) P_U(u_l | u_n), \quad (2.17)$$

where the state transition probability $P_U(u_l|u_n)$ is given by (2.8).

The initial value of $P_l(F_j)$ is given by

$$P(F_1, u_l) = Q_l P_U(u_l),$$

where $P_U(u_l)$, the steady state probability, is given by (2.5).

Once $P_l(F_j)$ is known, the performance of the rate-adaptive ARQ protocol can be evaluated. Let $I(F_i)$ be the indicator function of F_i. The average number of retransmissions D is given by

$$\begin{aligned} D &= E[\sum_{j=1}^{\infty} I(F_j)] && (2.18) \\ &= \sum_{j=1}^{\infty} P(F_j) \\ &= \sum_{j=1}^{\infty}\sum_{l=1}^{L} P_l(F_j). && (2.19) \end{aligned}$$

The throughput for the selective-repeat ARQ schemes is defined as the average number of information bits per transmitted channel symbol,

$$\eta \triangleq \frac{K}{E(\sum_{j=1}^{\infty} \Omega_j)}, \qquad (2.20)$$

where Ω_j is the number of code symbols transmitted during the j-th transmission. For the first transmission, $\Omega_1 = N_l^*$, where l is the channel state of the first transmission. For $j \geq 2$, Ω_j is given by

$$\Omega_j = \begin{cases} N_l^*, & F_{j-1} \\ 0, & \overline{F}_{j-1}, \end{cases}$$

where l is the channel state of the j-th transmission. The expectation of Ω_j is given by

$$E[\Omega_1] \;=\; \sum_{l=1}^{L} P(U_1 = u_l) E[\Omega_1 | U_1 = u_l]$$

$$
\begin{aligned}
&= \sum_{l=1}^{L} P_U(u_l) N_l^*, && (2.21)\\
E[\Omega_j] &= \sum_{l=1}^{L} P(F_{j-1}, U_j = u_l) E[\Omega_j | F_{j-1}, U_j = u_l]\\
&\quad + \sum_{l=1}^{L} P(\overline{F}_{j-1}, U_j = u_l) E[\Omega_j | \overline{F}_{j-1}, U_j = u_l]\\
&= \sum_{l=1}^{L} P(F_{j-1}, U_j = u_l) N_l^*\\
&= \sum_{l=1}^{L} N_l^* \sum_{n=1}^{L} P(F_{j-1}, U_{j-1} = u_n) P(U_j = u_l | F_{j-1}, U_{j-1} = u_n)\\
&= \sum_{l=1}^{L} N_l^* \sum_{n=1}^{L} P_n(F_{j-1}) P_U(u_l | u_n) \quad \text{for } j \geq 2. && (2.22)
\end{aligned}
$$

Substituting (2.21) and (2.22) into (2.20), we have

$$
\begin{aligned}
\eta &= \frac{K}{\sum_{l=1}^{L} P_U(u_l) N_l^* + \sum_{j=2}^{\infty} \sum_{l=1}^{L} N_l^* \sum_{n=1}^{L} P_n(F_{j-1}) P_U(u_l|u_n)}\\
&= \frac{K}{\sum_{l=1}^{L} N_l^* [P_U(u_l) + \sum_{j=1}^{\infty} \sum_{n=1}^{L} P_n(F_j) P_U(u_l|u_n)]}. && (2.23)
\end{aligned}
$$

In the numerical evaluation of (2.19) and (2.23), finite terms were taken as an approximation of the infinite sum.

In this chapter, energy efficiency is calculated taking into account only the transmitted power, because in most radio systems the transmitter radiated power accounts for more than 50% of the total system power. The energy efficiency is defined as the reciprocal of the the total energy dissipated for each correctly received bit, $\tilde{E}_b$ Joule/bit. To generalize the results for any background noise power spectrum density N_0, the normalized energy efficiency is defined as

$$
\begin{aligned}
\lambda &\triangleq \frac{N_0}{\tilde{E}_b}\\
&= \frac{N_0}{E_s} \eta. && (2.24)
\end{aligned}
$$

2.3.2 Fixed-Rate Type-I Hybrid ARQ

In this section, the throughput, delay, and energy efficiency performance for a fixed-rate type-I Hybrid ARQ scheme is analyzed. In a fixed-rate type-I hybrid ARQ scheme, an (N, K) code is used for error correction under all channel conditions. As a special case of rate-adaptive type-I hybrid ARQ scheme, the formulas derived in last section can also be used for fixed-rate ARQ protocol with minor changes.

The error-recursion algorithm for $P_l(F_j)$ is the same as given in (2.17) for the rate-adaptive case with Q_l modified as

$$Q_l = \sum_{i=t+1}^{N} \binom{N}{i} p_{s|l}^{i} (1 - p_{s|l})^{N-i}. \tag{2.25}$$

The number of retransmissions D is given by (2.19). The throughput η is given by

$$\eta = \frac{K}{N \sum_{l=1}^{L} [P_U(u_l) + \sum_{j=1}^{\infty} \sum_{n=1}^{L} P_n(F_j) P_U(u_l|u_n)]}. \tag{2.26}$$

The energy efficiency λ is given by (2.24).

2.3.3 IRR Type-II Hybrid ARQ

In this section, we analyze type-II hybrid ARQ schemes with incremental redundancy retransmission. In this protocol, the information sequence is encoded with an (N, K) extended Reed-Solomon code. During the first transmission, only K information symbols are transmitted. Upon the receipt of a NAK from the receiver, the transmitter transmits a redundancy block with Δ new code symbols. This procedure is repeated until a successful decoding occurs or decoding fails after all N symbols are received. In the case of decoding failure, the receiver throws away all previous symbols and restarts the protocol. As discussed in Section 2.2, the error correcting capability of the RS codeword increases with the number of retransmissions. This complication requires a completely different error recursion to be developed for IRR type-II ARQ protocols.

Let $P_c(j)$ denote the probability of first $j-1$ transmissions fail and the j-th transmission leads to a successful decoding. We write the expected number of retransmissions D for IRR type-II ARQ as a function of $P_c(j)$,

$$D = \sum_{j=2}^{\infty}(j-1)P_c(j), \tag{2.27}$$

and the throughput η as

$$\eta = \frac{K}{\sum_{j=1}^{\infty} P_c(j) \sum_{i=1}^{j} \Delta_i}, \tag{2.28}$$

where Δ_i is the number of code symbols in the i-th transmitted block as given by

$$\Delta_i \quad = \begin{cases} K, & i \bmod V = 1 \\ \Delta, & i \bmod V \neq 1. \end{cases}$$

where

$$V = (N-K)/\Delta + 1 \tag{2.29}$$

is the total number of code blocks within one codeword.

Let X_j denote the number of errors up to and including j-th transmission since the last restart of the protocol. From the definition, $P_c(j)$ can be written as

$$\begin{aligned} P_c(j) &\triangleq \sum_{i=t_{j-1}+1}^{t_j} P(X_j = i, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1) \\ &= \sum_{i=t_{j-1}+1}^{t_j} \sum_{l=1}^{L} P(X_j = i, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1, U_j = u_l). \end{aligned} \tag{2.30}$$

Define

$$P_X(j,i,u_l) \triangleq P(X_j = i, X_{j-1} > t_{j-1}, \cdots, X_1 > t_1, U_j = u_l).$$

The task of analyzing the system performance boils down to calculating $P_X(j,i,u_l)$. The recursive algorithm for calculating $P_X(j,i,u_l)$ can be described by the following theorem.

Theorem: For incremental redundancy retransmission schemes with incremental block size Δ, the probability $P_X(j,i,u_l)$ is given by

$$P_X(j,i,u_l) = \begin{cases} \sum_{k=1}^{\min(i-t_{j-1},\Delta)} Q_l(k) \sum_n P_X(j-1,i-k,u_n)P_U(u_l|u_n), & j \bmod V \neq 1 \\ P_X(1,i,u_l)(1-\sum_{s=1}^{j-1}\sum_n \mathrm{Cov}_{|j-s|}(l,n)\sum_{k=0}^{t_s} P_X(s,k,u_n)), & j \bmod V = 1 \end{cases} \tag{2.31}$$

where V is defined by (2.29), and

$$\begin{aligned} P_X(1,i,u_l) &= \binom{K}{i} p_{s|l}^i (1-p_{s|l})^{K-i} P_U(u_l), & (2.32) \\ Q_l(k) &= \binom{\Delta}{k} p_{s|l}^k (1-p_{s|l})^{\Delta-k}, & (2.33) \\ \mathrm{Cov}_{|j-s|}(l,n) &= \frac{P(U_s = u_n, U_j = u_l)}{P(U_j = u_l)P(U_s = u_n)}. & (2.34) \end{aligned}$$

Proof:

When the j-th frame is not the information frame, i.e., $j \bmod V \neq 1$, the probability of successful decoding depends on the total number of symbol errors since the last restart of the protocol and S_j, the number of symbol errors in the current transmission. The recursion is given by

$$\begin{aligned} P_X(j,i,u_l) &= \sum_{k=1}^{\min(i-t_{j-1},\Delta)} P(S_j = k, X_{j-1} = i-k, \cdots, X_1 > t_1, U_j = u_l) \\ &= \sum_k P(S_j = k|U_j = u_l) P(X_{j-1} = i-k, \cdots, X_1 > t_1, U_j = u_l), \end{aligned} \tag{2.35}$$

where $P(S_j = k|U_j = u_l)$ is denoted as $Q_l(k)$ in (2.33), $P(X_{j-1} = i-k, \cdots, X_1 > t_1, U_j = u_l)$ is given by

$$\begin{aligned} & P(X_{j-1} = i-k, \cdots, X_1 > t_1, U_j = u_l) \\ &= \sum_n P(X_{j-1} = i-k, \cdots, X_1 > t_1, U_j = u_l, U_{j-1} = u_n) \\ &= \sum_n P_X(j-1,i-k,u_n) P(U_j = u_l|U_{j-1} = u_n). \end{aligned} \tag{2.36}$$

Combining (2.35) and (2.36), we get the first half of (2.31) as follows

$$P_X(j,i,u_l) = \sum_{k=1}^{\min(i-t_{j-1},\Delta)} Q_l(k) \sum_n P_X(j-1,i-k,u_n) P_U(u_l|u_n) \text{ for } j \bmod V \neq 1.$$

When j mod $V = 1$, say, $j = wV + 1$, $P_X(j,i,u_l)$ can be written as

$$\begin{aligned} & P_X(wV+1,i,u_l) \\ = \; & P(X_{wV+1} = i, X_{wV} > t_{wV}, \cdots, X_1 > t_1, U_{wV+1} = u_l) \\ = \; & P(X_{wV+1} = i, U_{wV+1} = u_l) \\ & \cdot P(X_{wV} > t_{wV}, \cdots, X_1 > t_1 | U_{wV+1} = u_l, X_{wV+1}). \end{aligned} \tag{2.37}$$

At the $wV + 1$-th transmission, the protocol restarts, thus the error event at the $wV + 1$-th transmission is independent of previous error events given the channel state U_{wV+1}. It follows that

$$\begin{aligned} & P_X(wV+1,i,u_l) \\ = \; & P(X_{wV+1} = i, U_{wV+1} = u_l) P(X_{wV} > t_{wV}, \cdots, X_1 > t_1 | U_{wV+1} = u_l) \\ = \; & P(X_{wV+1} = i, U_{wV+1} = u_l) \\ & (1 - \sum_{s=1}^{wV} P(X_s \leq t_s, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1 | U_{wV+1} = u_l), \end{aligned} \tag{2.38}$$

where $P(X_s \leq t_s, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1 | U_{wV+1} = u_l)$ is the probability of the protocol success at the s-th transmission given U_{wV+1}. This probability can be written as

$$\begin{aligned} & P(X_s \leq t_s, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1 | U_{wV+1} = u_l) \\ = \; & \sum_{k=0}^{t_s} \sum_n P(X_s = k, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1, U_s = u_n | U_{wV+1} = u_l) \\ = \; & \sum_{k=0}^{t_s} \sum_n P(X_s = k, X_{s-1} > t_{s-1}, \cdots, X_1 > t_1, U_s = u_n) \\ & \cdot \frac{P(U_s = u_n, U_{wV+1} = u_l)}{P(U_{wV+1} = u_l) P(U_s = u_n)} \end{aligned} \tag{2.39}$$

Note that $P(X_{wV+1} = i, U_{wV+1} = u_l) = P_X(1, i, u_l)$ is independent of w. Combining (2.38) and (2.39), the recursion can be written as

$$P_X(j, i, u_l) = P_X(1, i, u_l)(1 - \sum_{s=1}^{j-1} \sum_{k=0}^{t_s} \sum_{n} P_X(s, k, u_n)\mathrm{Cov}_{|j-s|}(m, n)) \quad \text{for } j \bmod V = 1$$

Q.E.D.

With the knowledge of $P_X(j, i, u_l)$, $P_c(j)$ can be obtained via (2.30). Then the throughput, delay, and energy efficiency are computed according to (2.27), (2.28), and (2.24).

2.3.4 Soft-Decision Decoding

The performance of ARQ schemes with soft-decision Reed-Solomon decoders is discussed in this section. The error counting method for the soft-decision decoding algorithm is very similar to the methods described above for hard-decision decoding. The difference is that instead of counting the number of errors in a codeword, X_j, the new algorithm keeps track of a new quantity, X'_j, which is the sum of the number of erasures and twice the number of errors. In addition, the error correcting capability, t'_j, is redefined as $N-K$. After the first j transmissions, the RS code can be correctly decoded as long as $X'_j \leq t'_j$. Note that, in order to generate an erasure, the parity-check method adds one parity-check bit to each m-bit Reed-Solomon code symbol, which results in $\frac{1}{m+1}$ throughput reduction.

Given the above definition, all equations derived for a hard-decision system can also be used for a soft-decision system with X'_j and t'_j replacing X_j and t_j, respectively, with the exception of (2.16), (2.25), (2.32), and (2.33).

For rate-adaptive type-I hybrid ARQ schemes, let $\mathcal{C}'_l$ denote the optimal code for channel state u_l with soft-decision decoding. The corresponding code length and error correcting capability are denoted by N'_l and t'_l. The probability of frame error rate conditioned on channel state $U = u_l$, previously calculated with (2.16), is now given by

$$Q_l = \sum_{i=t_l'+1}^{N_l'} \sum_{r=0}^{\lfloor i/2 \rfloor} \binom{N_l'}{r} \binom{N_l' - r}{i - 2r} p_{s|l}^r p_{e|l}^{i-2r} (1 - p_{s|l} - p_{e|l})^{N_l' - i + r}, \tag{2.40}$$

where $p_{e|l}$ and $p_{s|l}$ are the probability of erasure and symbol error conditioned on channel state $U = u_l$ as given by (2.11) and (2.12).

For fixed-rate type-I hybrid ARQ schemes, Q_l, previously calculated with (2.25), is now given by

$$Q_l = \sum_{i=t'+1}^{N} \sum_{r=0}^{\lfloor i/2 \rfloor} \binom{N}{r} \binom{N - r}{i - 2r} p_{s|l}^r p_{e|l}^{i-2r} (1 - p_{s|l} - p_{e|l})^{N - i + r}, \tag{2.41}$$

For type-II hybrid ARQ schemes, the initial value of the recursion in (2.31), $P_X(1, i, u_l)$, previously defined in (2.32), is now given by

$$P_X(1, i, u_l) = \sum_{r=0}^{\lfloor i/2 \rfloor} \binom{K}{r} \binom{K - r}{i - 2r} p_{s|l}^r p_{e|l}^{i-2r} (1 - p_{e|l} - p_{s|l})^{K - i + r} P_U(u_l), \tag{2.42}$$

and the incremental error probability $Q_l(k)$, previously defined in (2.33), is now given by

$$Q_l(k) = \sum_{r=0}^{\lfloor k/2 \rfloor} \binom{\Delta}{r} \binom{\Delta - r}{k - 2r} p_{s|l}^r p_{e|l}^{k-2r} (1 - p_{e|l} - p_{s|l})^{\Delta - k + r}. \tag{2.43}$$

2.4 Numerical Results

The performance of fixed-rate type-I hybrid ARQ schemes, a genie-aided rate-adaptive type-I hybrid ARQ scheme, and IRR type-II hybrid ARQ schemes are numerically evaluated over fading channels with lognormal shadowing. Comprehensive comparisons are made between ARQ schemes, design parameters, decoding schemes, and channel parameters. The rate-adaptive type-I hybrid ARQ scheme presented here is optimal in the sense that perfect channel estimate is used to select the code rate to maximize throughput over each transmission. The performance of this scheme can serve as the upper bound on the performance of any practical variable-rate or fixed-rate scheme based on the same Reed-Solomon code family.

The first-order Markov model used in our analysis is fully described by the number of states, the steady state distribution, and the state transition probabilities. Both the accuracy of the model and the complexity of the numerical evaluation algorithm increase with the number of states. In this section, a Markov model of 32 states is used for numerical evaluation. The steady state distribution and the state transition probabilities are functions of ρ and σ_s as given by (2.5) and (2.8), where ρ is the normalized correlation of two shadowing levels between successive retransmissions, and σ_s is the standard deviation of the shadowing process. In [37], Gudmundson showed that σ_s depends on the propagation model. In this chapter, without loss of generality, the urban model is used, and the corresponding standard deviation is 4.3 dB. As described in Section II, ρ depends on both the propagation environment and the distance that a mobile travels between the successive retransmissions. In this chapter, system performance is evaluated at two channel correlation values: $\rho = 0.99$ and $\rho = 0.50$, where ρ is defined in (2.7).

In Table 2.1, the corresponding space separations in different propagation environments for the two correlation factors are tabulated. The correlation factor 0.99 corresponds to a distance separation of 0.084 meters in urban areas, which may results from a mobile unit traveling at 1 m/s with an acknowledgment delay of 84 millisecond. The same correlation factor also corresponds to a distance separation of 5 meters in suburban area, which may results from a mobile unit traveling at 30 m/s with an acknowledgment delay of 167 milliseconds. The correlation factor 0.5 corresponds to much larger space separation between retransmissions, as shown in

Table 2.1. This may occur in the scenarios of much higher mobile speed or much longer acknowledgment delay.

	Urban (m)	Suburban (m)
$\rho = 0.99$	0.084	5
$\rho = 0.5$	7.8	350

Table 2.1: Correlation of the shadowing process for adjacent frames and the corresponding propagation models and space separation

Note that the throughput and delay performances defined in previous section require the sum of items from infinite number of retransmissions. In our numerical evaluation of (2.19) and (2.20), 1000 retransmissions are recursively calculated for rate-adaptive type-I hybrid ARQ and fixed-rate type-I ARQ schemes. For type-II hybrid ARQ schemes, 200 retransmissions are sufficient for accurate evaluation of (2.27) and (2.28). The energy efficiency is derived from the throughput according to (2.24).

In Sections 2.4.1 to 2.4.3, the performance of the three types of ARQ protocols using hard-decision Reed-Solomon decoding is evaluated. In Section 2.4.4, the performance of ARQ protocols with error-and-erasure decoders is compared with the performance with errors-only hard-decision decoders.

2.4.1 Rate-Adaptive Type-I Hybrid ARQ and Fixed-Rate Type-I Hybrid ARQ

In this section, the throughput, delay, and energy efficiency of a genie-aided optimal rate-adaptive type-I hybrid ARQ scheme and fixed-rate type-I hybrid ARQ schemes are compared. The results presented in Figures 2.3 to 2.5 are based on the $(N_i, 16)$ Reed-Solomon code family over $GF(2^6)$. The channel correlation considered here is 0.99.

In Figure. 2.3, the throughput performance is plotted versus the channel SNR. For fixed-rate protocols, higher rate codes are shown to have better throughput at higher channel SNR, while the lower rate codes are shown to have better throughput at lower channel SNR. The rate-adaptive protocol is shown to uniformly achieve higher throughput than all fixed-rate codes for all SNR's.

In Figure. 2.4, the average numbers of retransmissions for the rate-adaptive and fixed-rate type-I hybrid ARQ schemes are plotted against the channel SNR. Since the rate-adaptive scheme adapts code rate to maximize the throughput, the delay performance is not guaranteed to be superior than the fixed-rate schemes. Because the rate 1/4 code has the best error correcting capability in the code family, the fixed-rate 1/4 coded scheme is shown to have the smallest delay among all ARQ schemes considered. Surprisingly, the rate-adaptive scheme has virtually the same delay performance as the rate 1/4 coded system in the region where the corresponding throughput is larger than 0.1. The rate 4/5 coded protocol is shown to have significantly larger delay than the rate 1/4 and 1/2 codes.

In Figure 2.5, the normalized energy efficiency is compared between fixed-rate schemes of different rates and the rate-adaptive scheme. The rate-adaptive scheme is shown to achieve a maximum energy efficiency of 0.009 and to outperform all fixed-rate schemes. Among the fixed-rate schemes, higher rate codes achieve better energy efficiency at high SNR, and lower rate codes achieve better energy efficiency at low SNR.

2.4.2 Type-II hybrid ARQ and Rate-Adaptive Type-I Hybrid ARQ

In this section, we present the answer to the most important problem investigated in this study: How do the type-II hybrid ARQ schemes compare with the best

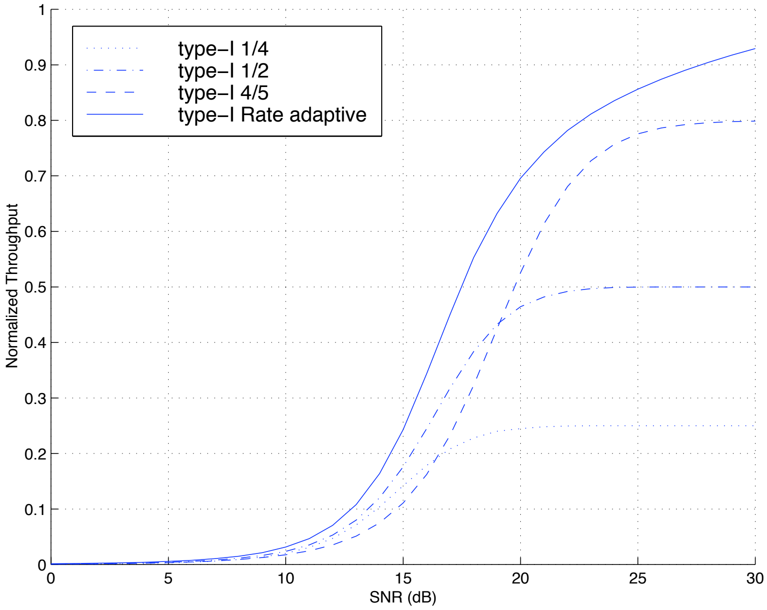


Figure 2.3: Throughput of rate-adaptive type-I hybrid ARQ schemes and fixed-rate type-I hybrid ARQ schemes: $\rho = 0.99$, $(N_i, 16)$ codes

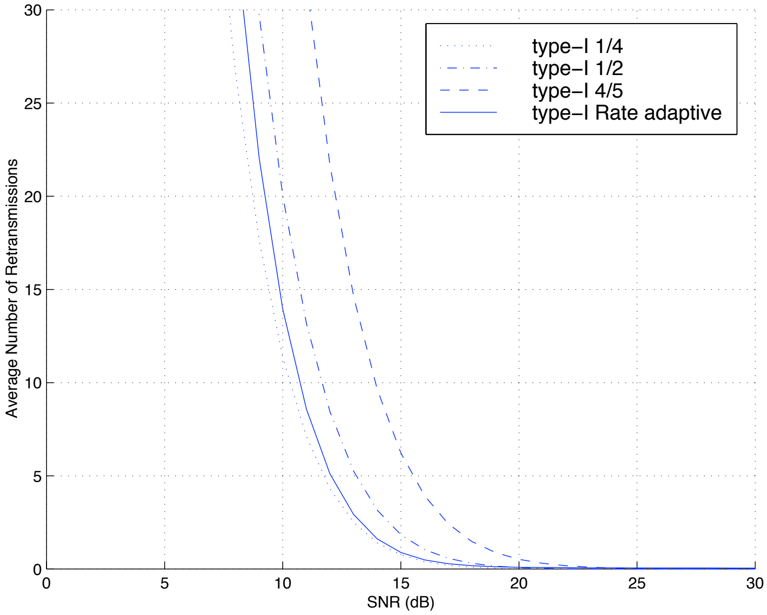


Figure 2.4: Average number of retransmissions of rate-adaptive type-I hybrid ARQ schemes and fixed-rate type-I hybrid ARQ schemes: $\rho = 0.99$, $(N_i, 16)$ codes

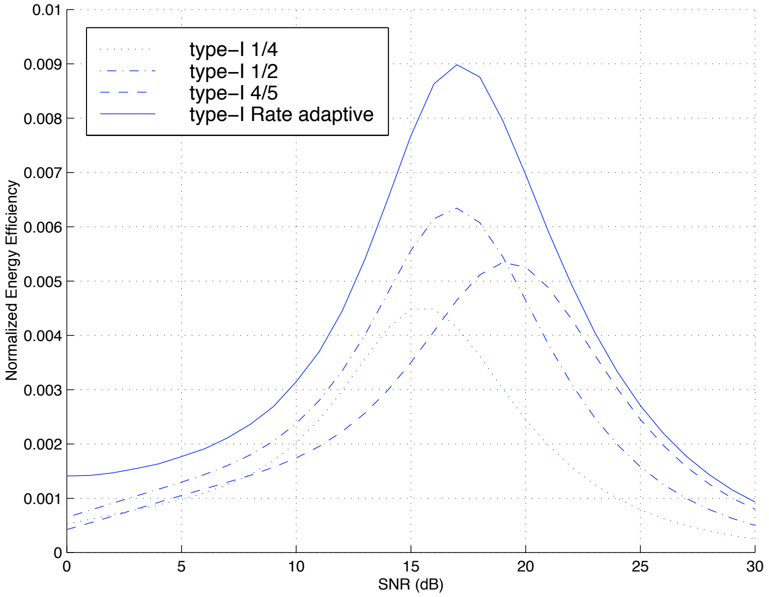


Figure 2.5: Normalized energy efficiency of rate-adaptive type-I hybrid ARQ schemes and fixed-rate type-I hybrid ARQ schemes: $\rho = 0.99$, $(N_i, 16)$ codes

possible type-I hybrid ARQ scheme? Incremental redundancy retransmission type-II hybrid ARQ schemes inherently adapt the code rate according to channel condition through a binary NAK or ACK feedback from the receiver. Rate-adaptive type-I hybrid ARQ adapts code rate based on channel estimate obtained either explicitly from channel measurement or implicitly from feedback of the error events at the receiver. The results presented in Figures 2.6 to 2.8 are based on the $(N_i, 16)$ Reed-Solomon code family over $GF(2^6)$. The channel correlation considered here is 0.99.

In Figure 2.6, the throughput performance is presented for the optimal rate-adaptive type-I ARQ scheme and IRR type-II hybrid ARQ schemes with redundancy blocks of size 2,4, and 8 symbols/block. It is observed that the throughput of type-II ARQ schemes with all three block sizes are notably larger than that of the best possible type-I ARQ scheme. Among the type-II schemes, the scheme with smaller block size shows better throughput than the schemes with larger block size. Note that, in the analysis of the type-II ARQ schemes, the overhead of a subsequence number for each redundancy block is neglected for simplicity. For very small redundancy blocks, the throughput presented here is somewhat optimistic.

In Figure 2.7, the rate-adaptive type-I ARQ scheme is shown to have better delay performance than all type-II ARQ schemes. For type-II ARQ schemes, the smaller the block size, the larger the delay. Consider the SNR range of interests, from 13 dB and above, where all four ARQ schemes achieve at least a throughput of 0.1. For the type-II scheme with block sizes 8 symbols, the average number of retransmissions is below 6, while the rate-adaptive type-I system has the number of retransmissions below 3. The longer delay of the type-II ARQ protocol also implies larger buffer requirements at both the transmitter and the receiver.

In Figure 2.8, the rate-adaptive type-I ARQ scheme is shown to have the worst energy efficiency among the schemes under comparison. The maximum energy efficiency of the type-I ARQ scheme is about 0.009, which is achieved at SNR around 15 dB with a throughput of 0.5. At the same SNR, the type-II ARQ with block size 8 symbols achieves energy efficiency of around 0.11, and throughput of 0.6.

From the discussion above, the optimal rate-adaptive type-I ARQ scheme is shown to have excellent delay performance and a smaller buffer requirement, both of which are desirable for high speed data links. However, the throughput and energy efficiency of the type-I ARQ scheme is notably inferior to IRR type-II ARQ schemes. In addition, the rate-adaptive type-I ARQ scheme considered here is a genie-aided scheme with perfect channel information. In practice, rate-adaptive systems require additional hardware for channel estimation, and imperfect channel estimation may significantly degrade the system performance [35].

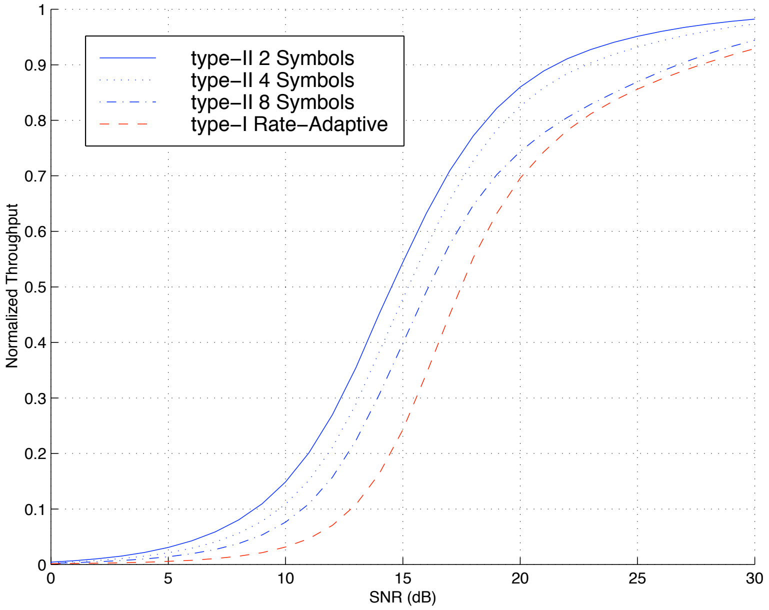


Figure 2.6: Throughput of incremental redundancy retransmission type-II hybrid ARQ schemes and rate-adaptive type-I hybrid ARQ schemes: $\rho = 0.99$, $(N_i, 16)$ codes

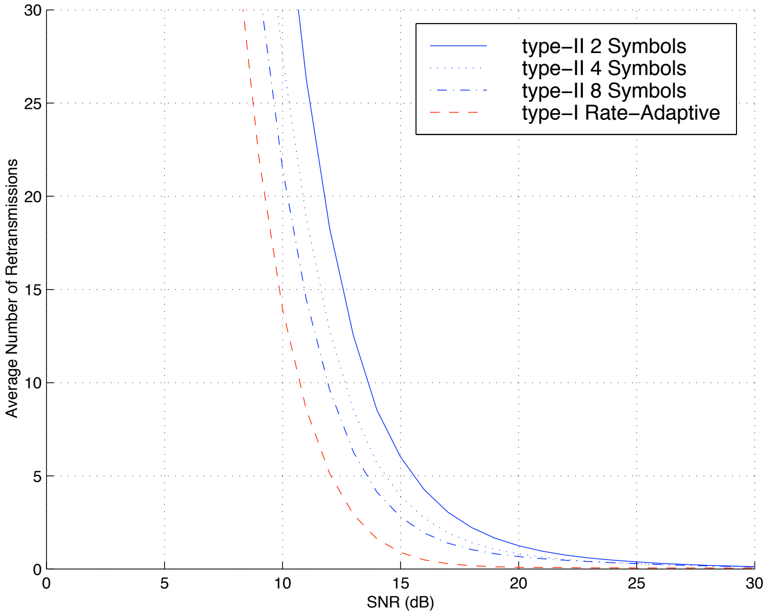


Figure 2.7: Average number of retransmissions of incremental redundancy retransmission type-II hybrid ARQ schemes and rate-adaptive type-I hybrid ARQ schemes: $\rho = 0.99$, $(N_i, 16)$ codes

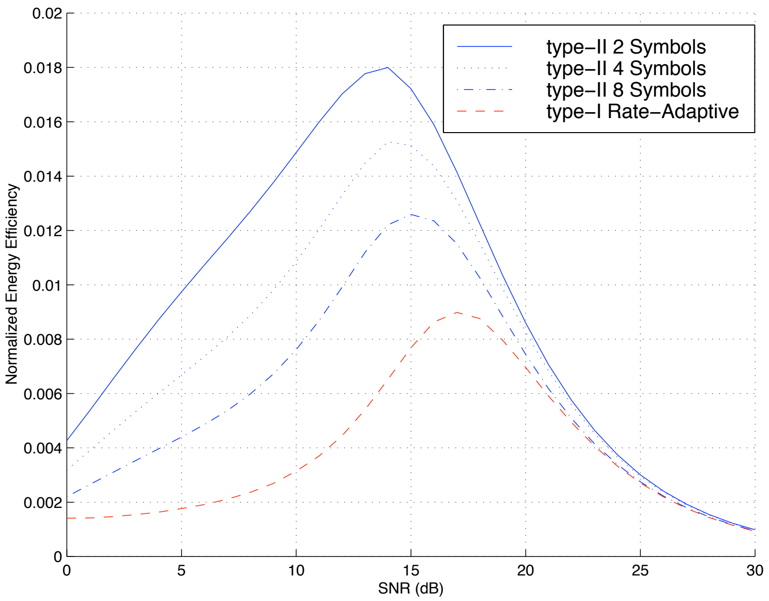


Figure 2.8: Normalized energy efficiency of incremental redundancy retransmission type-II hybrid ARQ schemes and rate-adaptive type-I hybrid ARQ schemes: $\rho = 0.99$, $(N_i, 16)$ codes

2.4.3 Hybrid ARQ over Channels of Different Correlation

The effect of channel correlation on the rate-adaptive type-I ARQ scheme and an IRR type-II ARQ scheme with 8 symbols/block is shown in Figures 2.9 to 2.14. Channel correlations, ρ, from 0.01 to 0.99 are considered. The figures show that there is not much performance difference between channels with correlation 0.50 and 0.01. The highly correlated channel with correlation 0.99 is observed to have large performance degradation at moderate SNR's.

For rate-adaptive type-I ARQ scheme over channels of correlation 0.99, the throughput achieves 0.1 at SNR around 13 dB according to Figure 2.9. The same protocol achieves a throughput of 0.1 at SNR around 7 dB for channel correlation 0.50. The throughput difference diminishes as SNR increases. For SNR 20 dB and higher, the protocol has slight performance gain in the correlation 0.99 channel over the correlation 0.50 channel.

The delay performance of rate-adaptive type-I ARQ shown in Figure 2.10 indicates around 6 dB loss for channels with $\rho = 0.99$ compared with channels with $\rho = 0.50$. For instance, SNR of 12 dB or higher is required to achieve average number of retransmissions less than 5 over a channel with correlation 0.99, while the average number of retransmission is below 5 for SNR 6 dB and above over a channel with correlation 0.50. The maximum energy efficiency of rate-adaptive type-I ARQ scheme over channels with correlation 0.50 is shown to be 0.019 compared with 0.09 over channels with correlation 0.99 (Figure 2.11).

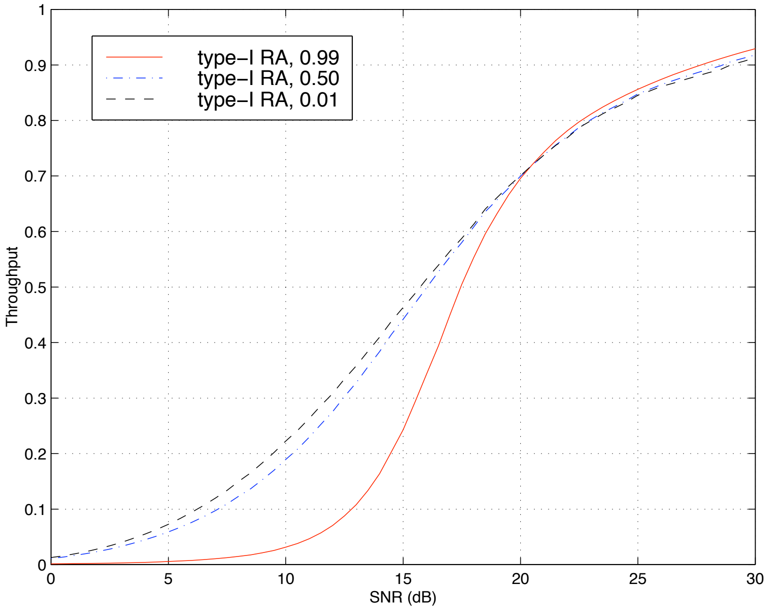


Figure 2.9: Throughput of rate-adaptive type-I hybrid ARQ schemes over channels with different correlations: $(N_i, 16)$ codes

For IRR type-II ARQ protocols, we observe similar performance degradation due to higher channel correlation. However, the degradation is within 4 dB for the same throughput performance (Figure 2.12) and average number of retransmissions (Figure 2.13). The maximum energy efficiency over channels of correlation 0.50 is 0.028 versus 0.013 over channels of correlation 0.99.

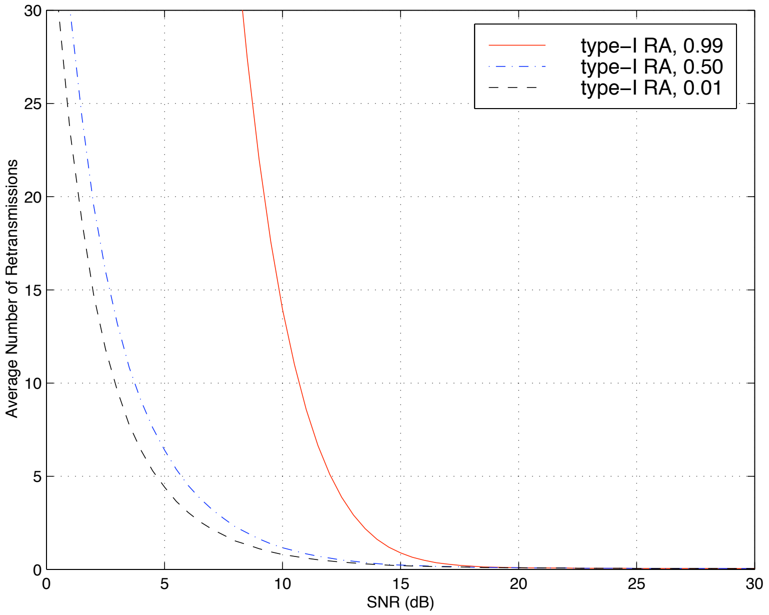


Figure 2.10: Average number of retransmissions of rate-adaptive type-I hybrid ARQ schemes over channels with different correlations: $(N_i, 16)$ codes

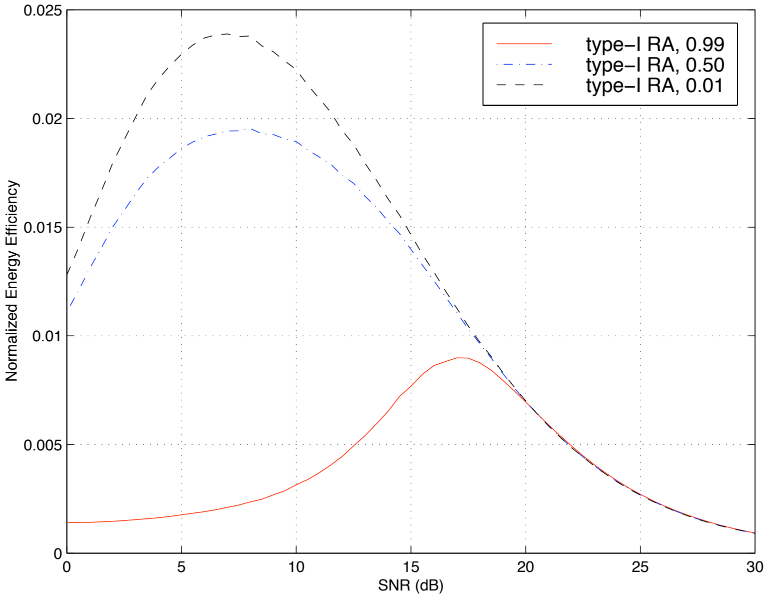


Figure 2.11: Normalized energy efficiency of rate-adaptive type-I hybrid ARQ schemes over channels with different correlations: $(N_i, 16)$ codes

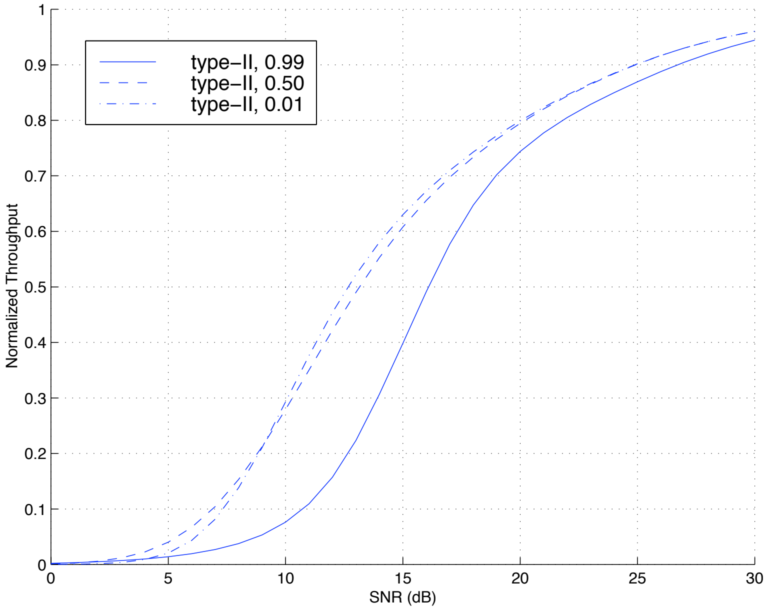


Figure 2.12: Throughput of incremental redundancy retransmission type-II hybrid ARQ schemes over channels with different correlations: $\Delta = 8$ symbols, $(N_i, 16)$ codes

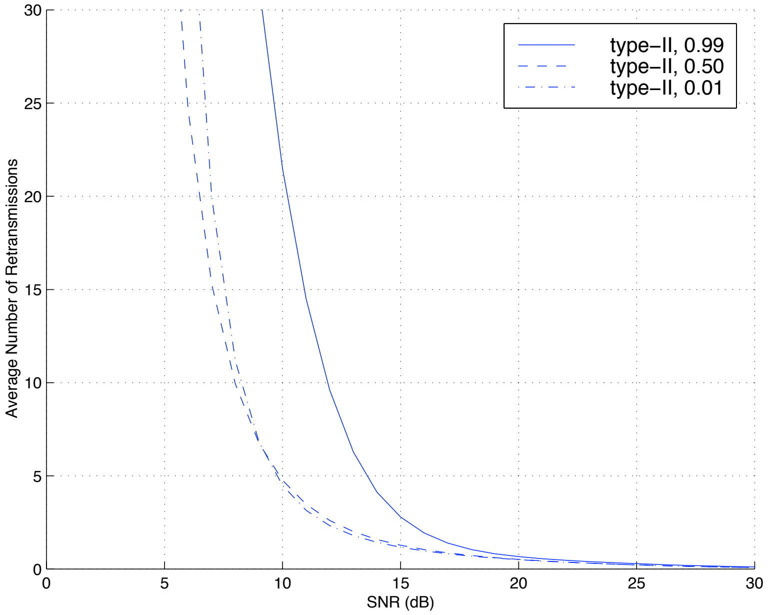


Figure 2.13: Average number of retransmissions of incremental redundancy retransmission type-II hybrid ARQ schemes over channels with different correlations: $\Delta = 8$ symbols, $(N_i, 16)$ codes

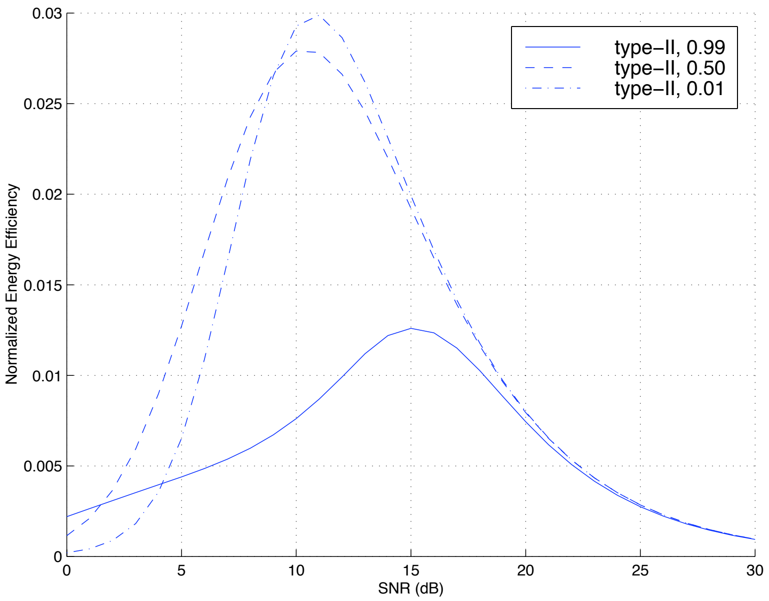


Figure 2.14: Normalized energy efficiency of incremental redundancy retransmission type-II hybrid ARQ schemes over channels with different correlations: $\Delta = 8$ symbols, $(N_i, 16)$ codes

2.4.4 Hybrid ARQ with Soft-Decision and Hard-Decision Decoding

Rate-adaptive type-I hybrid ARQ scheme and IRR type-II hybrid ARQ scheme are analyzed with error-and-erasure decoding for the $(N_i, 64)$ Reed-Solomon code family over $GF(2^8)$. The channel correlation considered in this section is 0.99.

For the rate-adaptive type-I hybrid ARQ scheme, the error-and-erasure decoding has close to 2 dB gain over errors-only decoding for channel SNR below 20 dB (Figures 2.15 and 2.16), and has a loss of 11% throughput for high SNR due to the overhead of the parity check bit. The maximum energy efficiency achieved by error-and-erasure decoding is about 40% more than that of errors-only decoding (Figure. 2.17).

For type-II hybrid ARQ scheme with block size 16 symbols, the error-and-erasure decoding improves the system performance for SNR below 18 dB. As shown in Figure 2.18, the errors-only decoding scheme achieves a throughput of 0.37, an energy

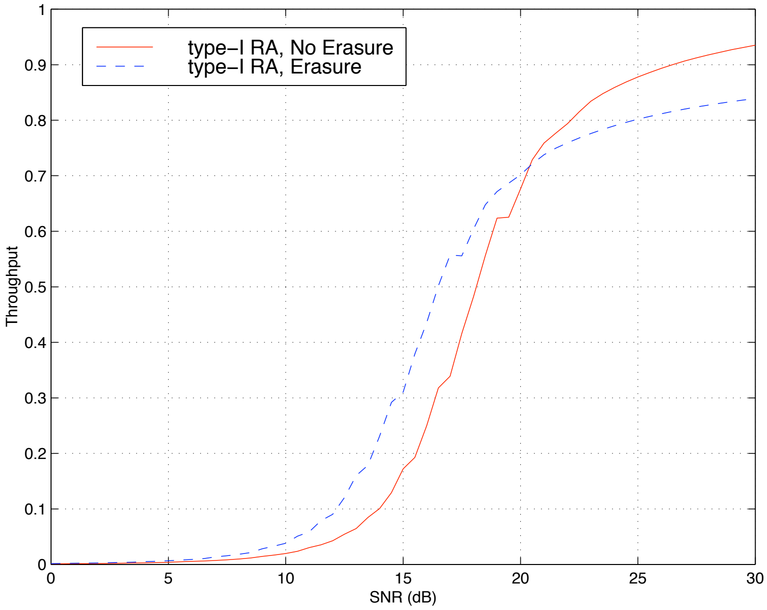


Figure 2.15: Throughput of rate-adaptive type-I hybrid ARQ schemes with errors-only and error-and-erasure decoding algorithms: $\rho = 0.99$, $(N_i, 64)$ codes

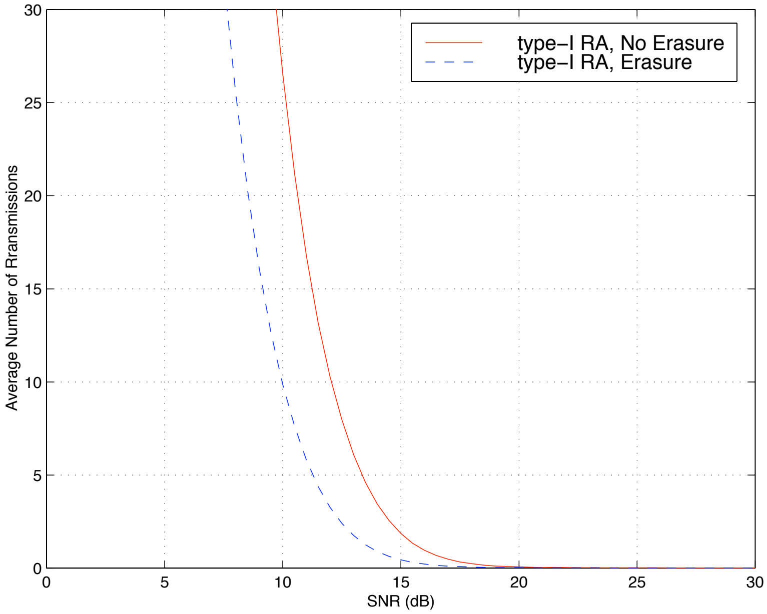


Figure 2.16: Average number of retransmissions of rate-adaptive type-I hybrid ARQ schemes with errors-only and error-and-erasure decoding algorithms: $\rho = 0.99$, $(N_i, 64)$ codes

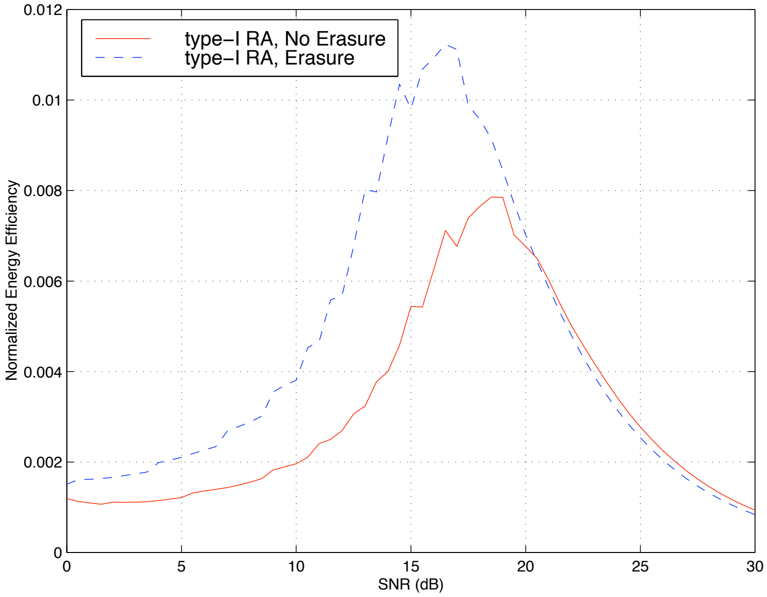


Figure 2.17: Normalized energy efficiency of rate-adaptive type-I hybrid ARQ schemes with errors-only and error-and-erasure decoding algorithms: $\rho = 0.99$, $(N_i, 64)$ codes

efficiency of 0.012, and an average number of retransmissions 5.8 at 15 dB, while the error-and-erasure decoding scheme achieves a throughput of 0.47, an energy efficiency of 0.015, and an average number of retransmissions 3.3. For SNR above 18 dB, the error-and-erasure decoding scheme has about 11% loss in throughput and energy-efficiency compared with the errors-only decoding scheme due to the additional parity check bit for each 8-bit RS code symbol.

2.5 Conclusions

Error recursion algorithms are developed for performance analysis of fixed-rate type-I hybrid ARQ schemes, a genie-aided rate-adaptive type-I hybrid ARQ scheme, and incremental redundancy retransmission type-II hybrid ARQ schemes over correlated fading channels. The central piece of this chapter is the analysis of IRR type-II hybrid ARQ schemes. The type-II ARQ protocols are robust and do not require any channel estimation while providing excellent performance. The performance of

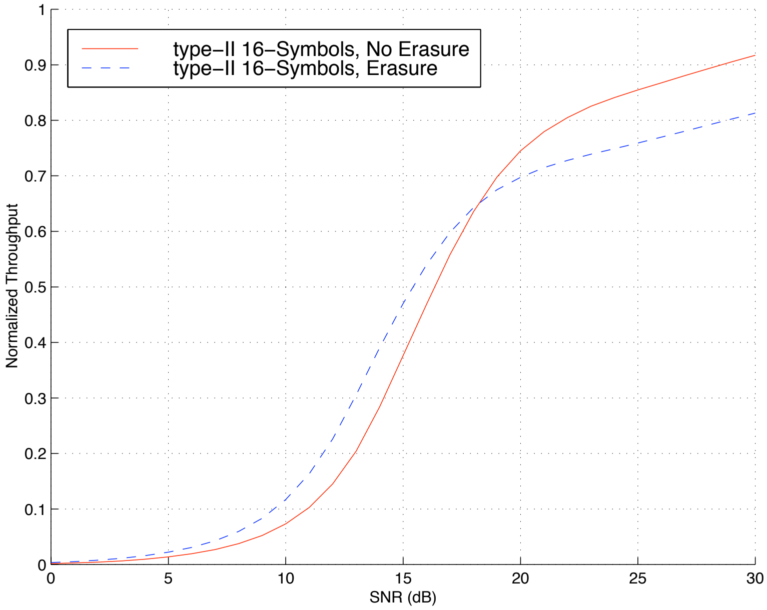


Figure 2.18: Throughput of incremental redundancy retransmission type-II hybrid ARQ schemes with errors-only and error-and-erasure decoding algorithms: $\rho = 0.99$, $\Delta = 16$ symbols, $(N_i, 64)$ codes

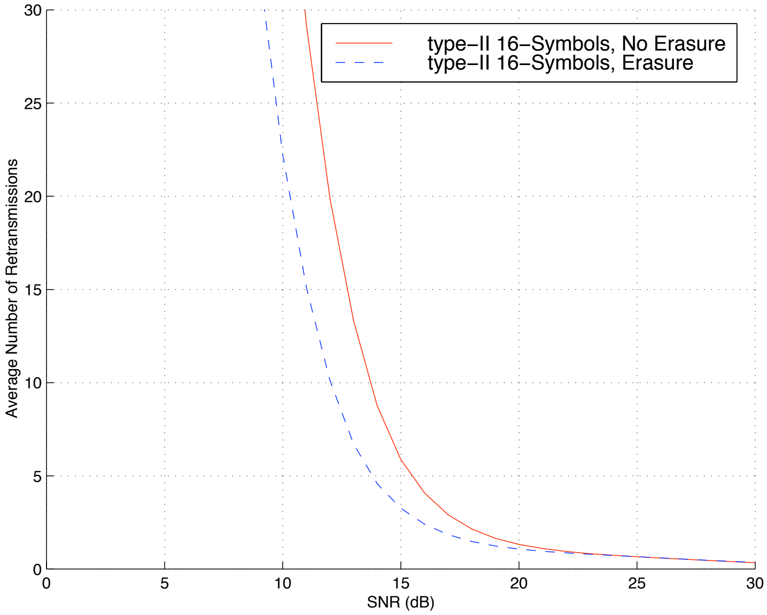


Figure 2.19: Average number of retransmissions of incremental redundancy retransmission type-II hybrid ARQ schemes with errors-only and error-and-erasure decoding algorithms: $\rho = 0.99$, $\Delta = 16$ symbols, $(N_i, 64)$ codes

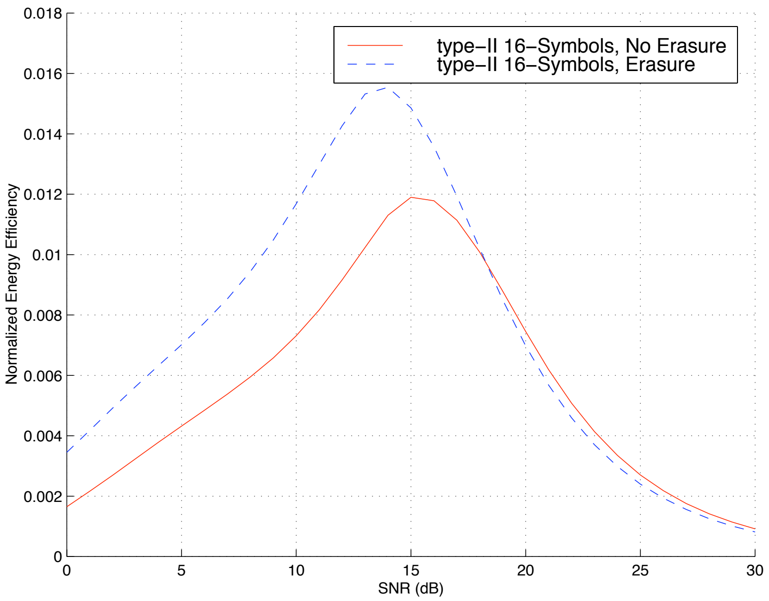


Figure 2.20: Normalized energy efficiency of incremental redundancy retransmission type-II with errors-only and error-and-erasure decoding algorithms: $\rho = 0.99$, $\Delta = 16$ symbols, $(N_i, 64)$ codes

optimal rate-adaptive type-I hybrid ARQ is also of great interests, since it defines the best possible achievable performance of any type-I hybrid ARQ scheme based on the same FEC code family.

Our numerical analysis shows that IRR type-II hybrid ARQ schemes achieve notably better throughput and energy efficiency than the optimal rate-adaptive type-I hybrid ARQ scheme, while the optimal rate-adaptive type-I hybrid ARQ scheme has smaller delay performance and requires less buffering at the transmitter and the receiver. High channel correlation is shown to degrade the system performance at moderate SNR for the ARQ schemes. Soft-decision decoding of Reed-Solomon code offers one to two dB gain compared with the errors-only hard-decision decoding algorithms in both type-I and type-II ARQ schemes. The recursive methodology proposed in this chapter can also be utilized to analyze other rate-adaptive block codes over general finite-state machine channels.

CHAPTER III

Rate Compatible Multiple Product Codes

Recently high rate product codes with large block lengths have been shown to achieve near Shannon capacity performance over AWGN channel with iterative decoding algorithms. In this chapter, we propose a simple construction of rate compatible low rate codes from relatively high rate product codes. One example is a code family with rates 0.72, 0.63, and 0.57 and a block length of 676 bits which outperforms S-random interleaved turbo codes of the same rates and comparable complexity for target BER $> 10^{-4}$ over AWGN and Rayleigh fading channels. An incremental redundancy retransmission type-II hybrid ARQ scheme based on rate compatible multiple product codes is described and the performance over an AWGN channel is evaluated through simulation.

3.1 Introduction

Product codes are two dimensional codes constructed from small component codes. Recent developments in iterative decoding show that high rate product codes can achieve performance close to Shannon capacity. In [57], a rate 0.98 Hamming product code of length 1026169 was shown to achieve a BER of 10^{-5} within 0.27 dB of Shannon's channel capacity with 2046 parallel 1024-state iterative decoders.

In contrast to high rate product codes, low rate product codes have been largely ignored due to high decoding complexity. Iterative decoding of product codes can be derived from the maximum likelihood decoding of component block codes. Maximum likelihood decoding of a linear block code can be implemented with a trellis search algorithm over a decoding trellis constructed from its parity check matrix [88]. The number of nodes (states) in the trellis grows exponentially with the number of parity check bits of the code. The large number of parity check bits of a low rate block component code results in prohibitive decoding complexity for a full trellis search algorithm. Suboptimal trellis search algorithms, such as the Chase algorithm [22] and Kaneko algorithm [47, 48], have been proposed to be used in product code decoding to reduce the decoding complexity[26, 67].

Many people also believe that turbo codes provide better performance than product codes for low code rates. In [40], iterative decoding algorithms for product codes were derived and the error performance was compared between product codes and turbo codes. Based on the simulation results in [40], a claim was made that for a given BER, there is a threshold rate, above which product codes are better and below which turbo codes are better. However, we note that the turbo codes used in the simulations in [40] had 4 states and a block length of 1000 bits, while their product code counterparts had 8 to 64 states in the decoding trellis and block lengths from 16 to 3249 bits. A fair comparison in terms of block lengths and decoding complexity between turbo and product codes has not been conducted.

In Benedetto and Montorsi's celebrated paper "Unveiling Turbo Codes" [9], the analysis of both parallel concatenated block codes and turbo codes are presented. The authors pointed out the similarity between parallel concatenated block codes and product codes, but no performance analysis for product codes was given. In other published literatures on product codes, only simulation results have been presented.

In this chapter, the construction of rate compatible multiple product codes is investigated. Powerful low rate codes are constructed from the original product code with the decoding complexity growing linearly with the code length. After the results presented in this chapter were published, an interesting construction of single parity check product code was brought to our attention. In [62], Ping *et at* introduced a sub-optimal soft-input-soft-output decoding rule for single parity check, and offered an example utilizing the new decoding rule in practice. In that example, a size 500×20 rate $5/6$ product code was constructed from a rate $20/21$ single parity check code. Despite the fact that the component codes, interleaver design, decoding algorithms, error performance, and design objective of the rate compatible product codes in our work are substantially different from that in [62], we find the underlying methodologies of both code constructions are similar.

Union bounds and simulations are provided to evaluate the performance of product codes over AWGN and Rayleigh fading channels. The resulting codes are then compared with turbo codes of the same rates, block lengths, and similar complexity. Finally, the rate compatible product codes are applied to an incremental redundancy retransmission (IRR) type-II hybrid ARQ scheme and the throughput performance of the ARQ scheme is evaluated via simulations.

The remainder of the chapter is organized as follows. In Section 3.2, the structure of product codes and an iterative decoding algorithm for product codes are discussed. In Section 3.3, we propose the construction of low rate *multiple* product codes and apply the resulting rate compatible codes in an IRR type-II hybrid ARQ scheme. In Section 3.4, the weight enumerators and union bounds are derived for product codes. In Section 3.5, simulation results and numerical evaluation of the union bounds are presented and compared with turbo codes. We conclude the chapter with a few remarks on the findings in Section 3.6.

3.2 Product Codes

Product codes are two dimensional codes constructed from small component codes, and represent a special class of parallel concatenated block codes. In a standard product code, the information bits are arranged into a $K_1 \times K_2$ array (Figure 3.1). Then each row is encoded by an $(N_2,\ K_2)$ systematic block code and each column is encoded by an $(N_1,\ K_1)$ systematic block code. A code is systematic if the original information bits are included in its codewords. The parity check bits of the horizontal code and vertical code are written into array H and V, respectively. In a more canonical form of a product code, the horizontal parity check bits were also encoded by the $(N_1,\ K_1)$ code to generate check-on-check parity bits. In this chapter, no check-on-check bits are considered.

3.2.1 Iterative Decoding Algorithm

The block diagram of an iterative decoder of the product code is illustrated in Figure 3.2. During each iteration, a soft-input-soft-output (SISO) module estimates the log-likelihood ratio (LLR) of the information bits based on the channel observations and *extrinsic* information provided by the other SISO module. The updated estimates of the information bits are then fed to the other SISO module as extrinsic information for the next iteration. A detailed description of the iterative decoding algorithm for product codes can be found in [40].

In Figure 3.2, L_I, L_H, and L_V denote the initial LLR generated by the demodulator for the information bits, horizontal parity checks, and vertical parity checks, respectively. Let $L_{e,H}$ and $L_{e,V}$ denote the *extrinsic* likelihood estimates by the hor-

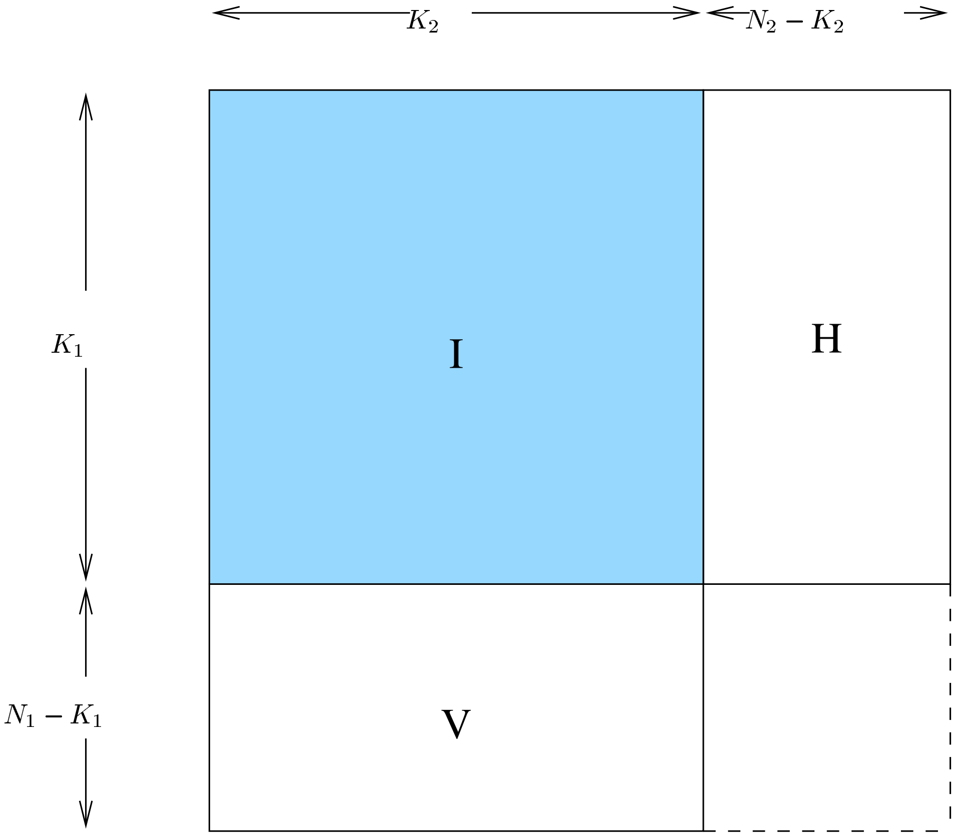


Figure 3.1: Information bits and parity bits array of a standard product code

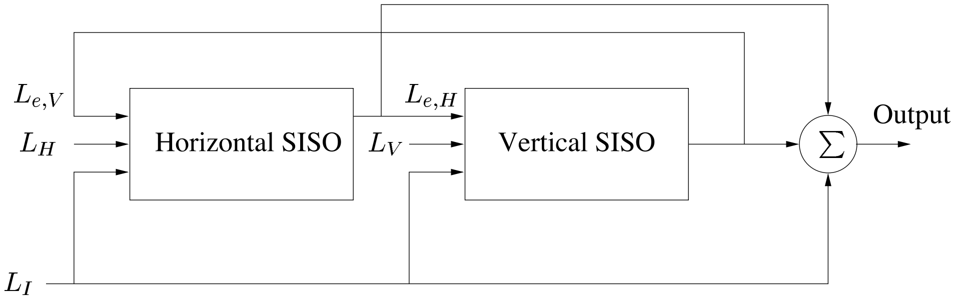


Figure 3.2: Block diagram of an iterative decoder for product code

izontal and vertical SISO modules. After a few iterations, the decoder outputs the final estimate of the LLR for information bit u as given by

$$L_u = L_I + L_{e,V} + L_{e,H}.$$

The algorithm for the SISO modules is discussed in Section 3.2.2.

The inputs to the product code decoder, L_I, L_H, and L_V, are determined by the modulation scheme and the channel model. In this chapter, we consider a binary phase shift keying (BPSK) modulated system with coherent demodulation. The low-pass equivalent of the transmitted signal for a bit u is given by

$$s(t) = \sqrt{2P} u p_T(t), \quad u \in [+1, -1],$$

where P is the transmitted signal power, and $p_T(t)$ is a rectangular pulse given by

$$p_T(t) = \begin{cases} 1 & t \in [0, T] \\ 0 & \text{otherwise.} \end{cases}$$

AWGN channels and frequency nonselective slow fading channels are considered in this chapter for performance evaluation. Perfect channel side information about the fading level is assumed to be available at the decoder. An interleaver is used to break burst errors caused by the slow fading process. With sufficient interleaving depth, the fading level can be modeled as constant over each bit duration and independent from bit to bit.

For Rayleigh fading channels, the lowpass equivalent of the received signal $r(t)$ is given by

$$r(t) = \alpha e^{j\varphi} s(t) + n(t),$$

where α is a random variable following a Rayleigh distribution as given by (2.3), and φ is a random variable following a uniform distribution over $[0, 2\pi]$. The background noise $n(t)$ is a white complex Gaussian random process with double-side power spectrum density N_0. The coherent demodulator output, y, is given by [19]

$$y = \mathrm{Re}\left\{\int_0^T r(t)e^{-j\varphi}dt\right\}. \tag{3.1}$$

For AWGN channel, the received signal $r(t) = s(t) + n(t)$, and the demodulator output y is given by $y = \mathrm{Re}\left\{\int_0^T r(t)dt\right\}$.

The LLR of a binary input u conditioned on the demodulator output y over AWGN channel is given by

$$\begin{aligned} L_{U|Y}(y) &\triangleq \log\frac{P(u=+1|y)}{P(u=-1|y)} && (3.2)\\ &= 4\sqrt{2P}y/N_0. && (3.3)\end{aligned}$$

The LLR of a binary input u conditioned on the demodulator output y and the channel side information α over a Rayleigh fading channel is given by

$$\begin{aligned} L_{U|Y}(y,\alpha) &\triangleq \log\frac{P(u=+1|y,\alpha)}{P(u=-1|y,\alpha)} && (3.4)\\ &= 4\alpha\sqrt{2P}y/N_0. && (3.5)\end{aligned}$$

3.2.2 Soft-Input-Soft-Output Decoder

Consider an $(N,K)^2$ product code. The SISO module in Figure 3.2 is composed of K parallel SISO decoders for each component linear block code. A symbol-by-symbol maximum *a posteriori* (MAP) decoder can serve as the optimal SISO decoder for each component code. For an (N,K) linear block code, a decoding trellis can be generated from the parity check matrix $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_N\}$, where $\mathbf{h}_k$ is a column vector of $N-K$ elements [88]. If $\mathbf{X} = \{x_1, x_2, \cdots, x_N\}^T$ is a codeword of the code defined by $\mathbf{H}$, then $\mathbf{X}$ satisfies the parity check equation, i.e.,

$$\mathbf{HX} = \mathbf{0}. \tag{3.6}$$

A decoding trellis has a maximum of 2^{N-K} states at each stage, while a minimum trellis of fewer states exists for many codes [49]. In the decoding trellis, let $\mathbf{s}_k$ denote a state at the k-th decoding stage, for $k = 1, \cdots, N+1$. State $\mathbf{s}_i$ is an $(N-K)$-tuple given by

$$\mathbf{s}_k = \begin{cases} \mathbf{s}_{k-1} + \mathbf{h}_{i-1}x_{i-1} \quad \mod 2, & k \geq 2 \\ (0, \cdots 0)^T, & k = 1 \end{cases}$$

where x_{k-1} is the $(k-1)$-th code bit. Since a codeword must satisfy (3.6), the corresponding decoding path always ends in the all zero state.

In the rest of this section, the symbol-by-symbol MAP algorithm over AWGN channel is briefly described. The LLR of information bit u_k conditioned on demodulator output sequence $\mathbf{y}$ is given by

$$\begin{aligned} L(u_k) &\triangleq \log \frac{P(u_k = +1|\mathbf{y})}{P(u_k = -1|\mathbf{y})} \\ &= \log \frac{\sum_{(s',s)|u_k=+1} P((s',s)|\mathbf{y})}{\sum_{(s',s)|u_k=-1} P((s',s)|\mathbf{y})}, \end{aligned}$$

where the sum is taken over all state transitions (s', s) from depth k to $k+1$ corresponding to the input value u_k. As shown in [40], $L(u_k)$ can be written as

$$\begin{aligned} L(u_k) &= L_{U|Y}(y_k) + \hat{L}(u_k) \\ &+ \log \frac{\sum_{(s',s)|u_k=+1} \gamma_k^{(e)}(s',s) \cdot \alpha_{k-1}(s')\beta_k(s)}{\sum_{(s',s)|u_k=-1} \gamma_k^{(e)}(s',s) \cdot \alpha_{k-1}(s')\beta_k(s)}, \end{aligned} \tag{3.7}$$

where $L_{U|Y}(y_k)$ is given by (3.2), and $\hat{L}(u_k)$ is the *a priori* LLR of u_k. The variables $\gamma^{(e)}$, α, and β are computed with BCJR algorithm which requires one forward recursion and one backward recursion of the trellis [6].

In the first iteration, $\hat{L}^{(1)}(u_k)$ is set to zero. In the n-th iteration, for $n > 1$, $\hat{L}^{(n)}(u_k)$ is given by

$$\hat{L}^{(n)}(u_k) = L_e^{(n-1)}(u_k)' \tag{3.8}$$

where $L_e^{(n-1)}(u_k)'$ is the extrinsic LLR for u_k generated by the other decoder at the $(n-1)$-th iteration. Recall that each information bit u_k is encoded by one horizontal component code and one vertical component code. $L_e^{(n-1)}(u_k)'$ is given by

$$L_e^{(n-1)}(u_k)' = L^{(n-1)}(u_k)' - L_{U|Y}(y_k) - \hat{L}^{(n-1)}(u_k)'. \tag{3.9}$$

Suppose the algorithm terminates at the n-th iteration, the LLR estimate of the bit u_k is given by

$$L_f(u_k) = L_{U|Y}(y_k) + L_{e,h}^{(n)}(u_k) + L_{e,v}^{(n)}(u_k).$$

There are a few alternative implementations of the SISO module, such as decoders based on the trellis of dual code [40], the modified Chase algorithm [67], and the modified Kaneko algorithm [26]. In our simulation, the standard Log-MAP BCJR algorithm is used. Details of this algorithm can be found in [6, 40].

For decoding over Rayleigh fading channels, the MAP algorithm is identical to that for the AWGN channel, except $L_{U|Y}(y_k)$ in (3.7) and (3.9) is replaced with $L_{U|Y}(y_k, \alpha)$ as defined by (3.4).

3.3 Rate Compatible Codes

In this section, a simple construction of rate compatible codes of higher and lower rates from an original product code is presented. Rate compatible codes are highly desirable for both rate adaptation systems based on channel estimation and incremental redundancy retransmission type-II hybrid ARQ schemes.

Consider an $(N, K)^2$ Hamming product code. This code has rate $\frac{K^2}{N^2-(N-K)^2}$. If higher rates are desired, we can reduce the redundancy by perforating the parity check bits from each component code. Due to the symmetry of the Hamming code, bit positions in the puncturing pattern do not affect the weight distribution of the resulting component code. Hence, no optimization of puncturing patterns is necessary. We observe that puncturing a product code may substantially weaken the error correcting capability of the code. For example, the $(15, 11)$ Hamming code has weight enumerator $A(Z) = 1 + 35Z^3 + 105Z^4 + \cdots$; the punctured $(12, 11)$ code has $A(Z) = 1 + 4Z + 34Z^2 + \cdots$; and the punctured $(13, 11)$ code has $A(Z) = 1 + Z + 18Z^2 + \cdots$. It is obvious that there are at least one pair of codewords at distance 1 which means there are unprotected bits in both punctured Hamming codes.

When codes with rates lower than the original product code are desired, we need to find a proper way to add more redundancy. A straightforward method is to extend the component (N, K) block code to an $(N + \Delta, K)$ code. However, as stated before, the decoding complexity grows exponentially with the number of parity check bits in the component code, hence, for every additional bit added to the component code, the decoding complexity doubles. Apparently, this approach is too complex for practical purpose. In this chapter, we propose to lower the code rate of product codes by introducing *multiple* product codes.

We demonstrate the procedure of lowering the rate of an product code from $\frac{K}{K+2(N-K)}$ to $\frac{K}{K+M(N-K)}$ in Figure 3.3. The information bits are first encoded row by row to form the horizontal code. When an (N, K) systematic block code is used, the redundancy parity bits are written into a K by $N - K$ block, denoted by P_1. The information array is then interleaved (or permutated), and encoded again with the (N, K) code on each row to form a parity block P_2. When block interleaving is used, P_1 and P_2 are equivalent to the H and V parity block shown in Figure 3.1. The process of adding parity checks is repeated until the desired code rate $\frac{K}{K+M(N-K)}$ is obtained with M parity blocks[1]. Multiple product codes are "M-dimensional" codes

[1] The resulting codes are no longer product codes in strict sense but instead are parallel concatenated block codes. However, since they are constructed from the original product codes and share many features of product codes, they will be referred to as rate compatible multiple product codes or simply rate compatible product codes in this thesis.

where each bit is encoded by M component codes, as compared with the original product codes where each information bit is encoded twice by one horizontal and one vertical block codes.

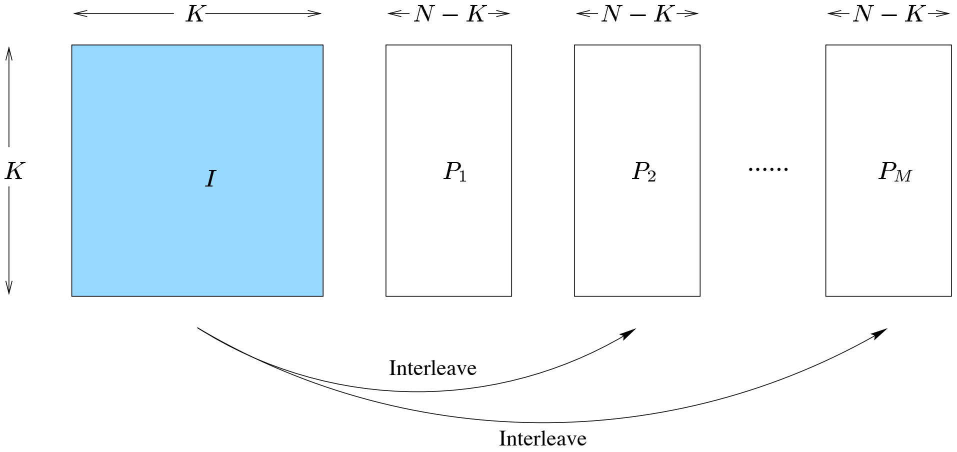
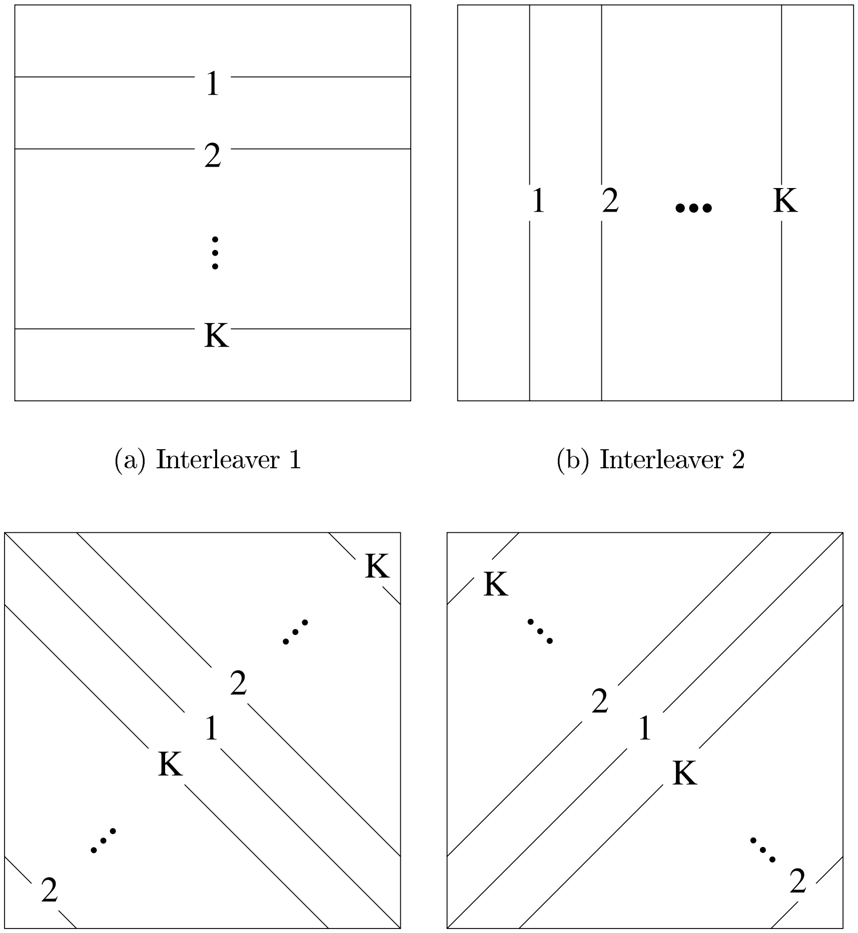


Figure 3.3: Construction of an M-dimensional multiple product code

In Figure 3.4, four block interleavers for a four dimensional product code are illustrated. Let the information bits fill a $K \times K$ array. The horizontal and vertical encoders encode the information bits in each row and column, respectively; a third encoder encodes the information bits along the upper left to lower right diagonals; and the fourth encoder encodes the information bits along the upper right to lower left diagonals.

Interleavers that are generated by pseudo-random number generators can also be employed in multiple turbo codes and offer more flexibility in encoder and decoder design. Rank and Gulliver showed that random interleavers work well with single parity check product codes on three or higher dimensional information arrays [68]. To avoid random interleavers that result in poor message passing between the component codes, the S-random interleavers are adopted in our work. An S-random interleaver is generated randomly with the property that any two bit positions within distance S in the input sequence are spread beyond distance S in the output sequence [29].



(a) Interleaver 1

(b) Interleaver 2

(c) Interleaver 3

(d) Interleaver 4

Figure 3.4: Block interleavers for 4-D product codes

The bit error rates of product codes with block interleaver and S-random interleaver over AWGN channels are compared in Figure 3.5. For both $(31, 26)^2$ and $(15, 11)^2$ product Hamming codes, the performance difference is shown to be negligible for the two different interleaver implementations.

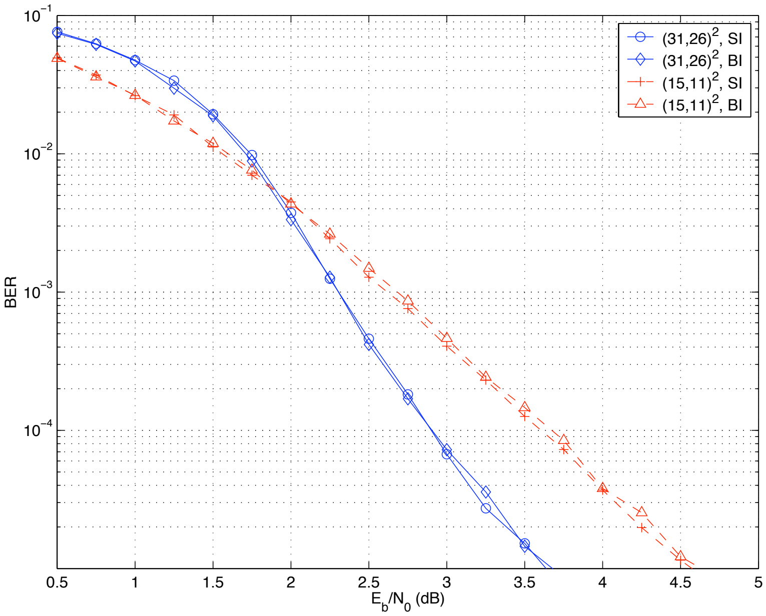


Figure 3.5: BER performance of $(31, 26)^2$ and $(15, 11)^2$ Hamming product codes with S-random interleaver and block interleaver over AWGN channels

A decoder for an M-dimensional product code consists of M SISO modules. Each module is identical to the decoding module in the original two dimensional product codes. The message passing among the SISO modules is similar to that illustrated in Figure 3.2 for two dimensional product codes. The difference is that the *a prior* LLR $\hat{L}_i^{(n)}(u_k)$ of the i-th decoder at the n-th iteration is given by

$$\hat{L}_i^{(n)}(u_k) = \sum_{j \neq i} L_{e,j}^{(n-1)}(u_k) \quad \text{for } n \neq 1. \tag{3.10}$$

The decoding algorithm is mathematically described by (3.7), (3.9), and (3.10). If a decision is made after n iterations for u_k, the final LLR is given by

$$L_f(u_k) = L_{U|Y}(y_k) + \sum_{i=1}^{M} L_{e,i}^{(n)}(u_k).$$

It is obvious that the decoding complexity grows linearly with the number of SISO modules in the decoder, or equivalently, the length of the multiple product codes.

3.4 Weight Enumerators and Union Bounds

In this section, the weight enumerators and union bounds are derived for multiple product codes. Consider an M-dimensional multiple product code with K^2 information bits. Let $\mathcal{B}$ denote the component (N, K) systematic linear block code. Let $\mathcal{C}$ denote the block code for the first dimension. Each codeword of $\mathcal{C}$ consists of K codewords form the component codes $\mathcal{B}$. Since the information array is a square, the same (N, K) block code can also be used on the other $M - 1$ dimensions.

The input-output-weight-enumerator (IOWE) of $\mathcal{B}$, $A^{\mathcal{B}}(I, W)$, can be expressed as

$$A^{\mathcal{B}}(I, W) \triangleq \sum_{i=0}^{K} \sum_{w=0}^{N-K} A^{\mathcal{B}}_{i,w} I^i W^w,$$

where $A^{\mathcal{B}}_{i,w}$ is the number of codewords in $\mathcal{B}$ with information weight i and parity weight w. It is usually straightforward to obtain $A^{\mathcal{B}}(I, W)$, because $\mathcal{B}$ is often small and has very regular algebraic structure.

The IOWE of the horizontal code $\mathcal{C}$, $A^{\mathcal{C}}(I, W)$, is defined as

$$A^{\mathcal{C}}(I, W) \triangleq \sum_{i=0}^{K^2} \sum_{w=0}^{K(N-K)} A^{\mathcal{C}}_{i,w} I^i W^w,$$

where $A^{\mathcal{C}}_{i,w}$ is the number of codewords in $\mathcal{C}$ with information weight i and parity weight w. It is easily shown that $A^{\mathcal{C}}(I, W)$ can be computed from $A^{\mathcal{B}}(I, W)$ as given by

$$A^{\mathcal{C}}(I, W) = (A^{\mathcal{B}}(I, W))^K. \tag{3.11}$$

Let $P^{\mathcal{C}}_{w|i}$ denote the probability of the encoder $\mathcal{C}$ generating a code sequence of parity weight w, given a random input sequence of weight i. Since there are a total of $\binom{K^2}{i}$

distinct weight i sequences, $P^{\mathcal{C}}_{w|i}$ can be written as

$$P^{\mathcal{C}}_{w|i} = A^{\mathcal{C}}_{i,w} / \binom{K^2}{i}. \tag{3.12}$$

In this analysis, a probabilistic device, a *uniform* random interleaver [9], is assumed to be used instead of a fixed interleaver. A uniform random interleaver of length k maps a given sequence of weight w onto all distinct $\binom{k}{w}$ permutations of it with equal probability $1/\binom{k}{w}$. The performance obtained under the uniform interleaver assumption is the ensemble average performance of product codes with randomly generated fixed interleavers.

The ensemble average IOWE of a product code can be derived with the knowledge of $P^{\mathcal{C}}_{w|i}$. From the definition of uniform random interleaver, a weight i input sequence is randomly mapped onto any distinct weight i sequence with probability $1/\binom{K^2}{i}$. Thus, the probability of one distinct weight i sequence being presented to an encoder on any dimension is the same. If we denote the code on the m-th dimension as $\mathcal{C}_m$, then it follows that $P^{\mathcal{C}_m}_{w|i}$ is the same for all m, and equals to $P^{\mathcal{C}}_{w|i}$. Let $P_{w_1,w_2,\cdots,w_M|i}$ denote the probability of a product code having parity weights $w_1, w_2, \cdots, w_M$ for codes on the M dimensions, respectively, given the input sequence weight i. Since the uniform random interleavers are assumed independent of each other, it follows that

$$P_{w_1,w_2,\cdots,w_M|i} = P^{\mathcal{C}}_{w_1|i} P^{\mathcal{C}}_{w_2|i} \cdots P^{\mathcal{C}}_{w_M|i}.$$

Let $P_{w|i}$ denote the probability of generating a product code of parity weight w from an input sequence of weight i. Then $P_{w|i}$ can be written as

$$P_{w|i} = \sum_{w_1=0}^{w} \sum_{w2=0}^{w-w_1} \cdots \sum_{w_M=0}^{w-\sum_{i=1}^{M-1} w_i} P_{w_1,\cdots,w_M|i}. \tag{3.13}$$

Define

$$P_i^{\mathcal{C}}(W) \triangleq \sum_{w=0}^{K(N-K)} P_{w|i}^{\mathcal{C}} W^w, \tag{3.14}$$

$$P_i(W) \triangleq \sum_{w=0}^{MK(N-K)} P_{w|i} W^w. \tag{3.15}$$

We can rewrite (3.13) as

$$P_i(W) = \overbrace{P_i^{\mathcal{C}}(W) * P_i^{\mathcal{C}}(W) * \cdots * P_i^{\mathcal{C}}(W),}^{M} \tag{3.16}$$

where $*$ denotes convolution.

The ensemble IOWE of a product code, $A(I,W)$, is defined as

$$A(I,W) \triangleq \sum_{i=0}^{K^2} \sum_{w=0}^{MK(N-K)} A_{i,w} I^i W^w,$$

where $A_{i,w}$ is the average number of product code codewords of input weight i and parity weight w. $A(I,W)$ can be written as a function of $P_i(W)$ as

$$A(I,W) = \sum_{i=0}^{K^2} \binom{K^2}{i} P_i(W) I^i. \tag{3.17}$$

The ensemble weight enumerator of a product code is defined as

$$A(Z) \triangleq \sum_{z=0}^{K(K+M(N-K))} A_z Z^z,$$

where A_z is the average number of codewords of weight z. By definition, $A(Z)$ can be written as

$$A(Z) = \sum_{z=0}^{K(K+M(N-K))} \sum_{i=0}^{\min\{K^2,z\}} A_{i,z-i} Z^z. \tag{3.18}$$

When $P_i(W)$ is known, the IOWE and weight enumerator of a product code can be obtained from (3.17) and (3.18), respectively.

The weight enumerators for the 2D, 3D and 4D product codes based on (31,26) Hamming code are shown in Figure 3.6(a). It is seen that there are more high weight codewords in codes of higher dimensions. For asymptotic performance at high SNR, we are more concerned about the number of low weight codewords. In Figure 3.6(b), the weight enumerators A_z for codewords of weight less than 20 is shown. The number of low weight codewords in 3D and 4D product codes are shown to be several orders of magnitude fewer than the original 2D product codes.

With the knowledge of IOWE of a product code, union bounds can be derived for the bit and packet error performance of the maximum likelihood decoder. Let $P_2(d, E_s/N_0)$ denote the pairwise error probability of two codewords of distance d apart with symbol energy-to-noise power spectrum density ratio E_s/N_0. The union bound is given by

$$P_b(\frac{E_b}{N_0}) \leq \sum_{i=0}^{K^2} \frac{i}{K^2} \sum_{w=0}^{MK(N-K)} A_{i,w} P_2(i+w, rE_b/N_0) \tag{3.19}$$

for the bit error rate, and

$$P_e(\frac{E_b}{N_0}) \leq \sum_{i=0}^{K^2} \sum_{w=0}^{MK(N-K)} A_{i,w} P_2(i+w, rE_b/N_0) \tag{3.20}$$

for the packet error rate, respectively, where $r = K/(K + M(N-K))$. The pairwise error probability is given by [63]

$$P_2(d, E_s/N_0) = Q(\sqrt{\frac{2dE_s}{N_0}}), \tag{3.21}$$

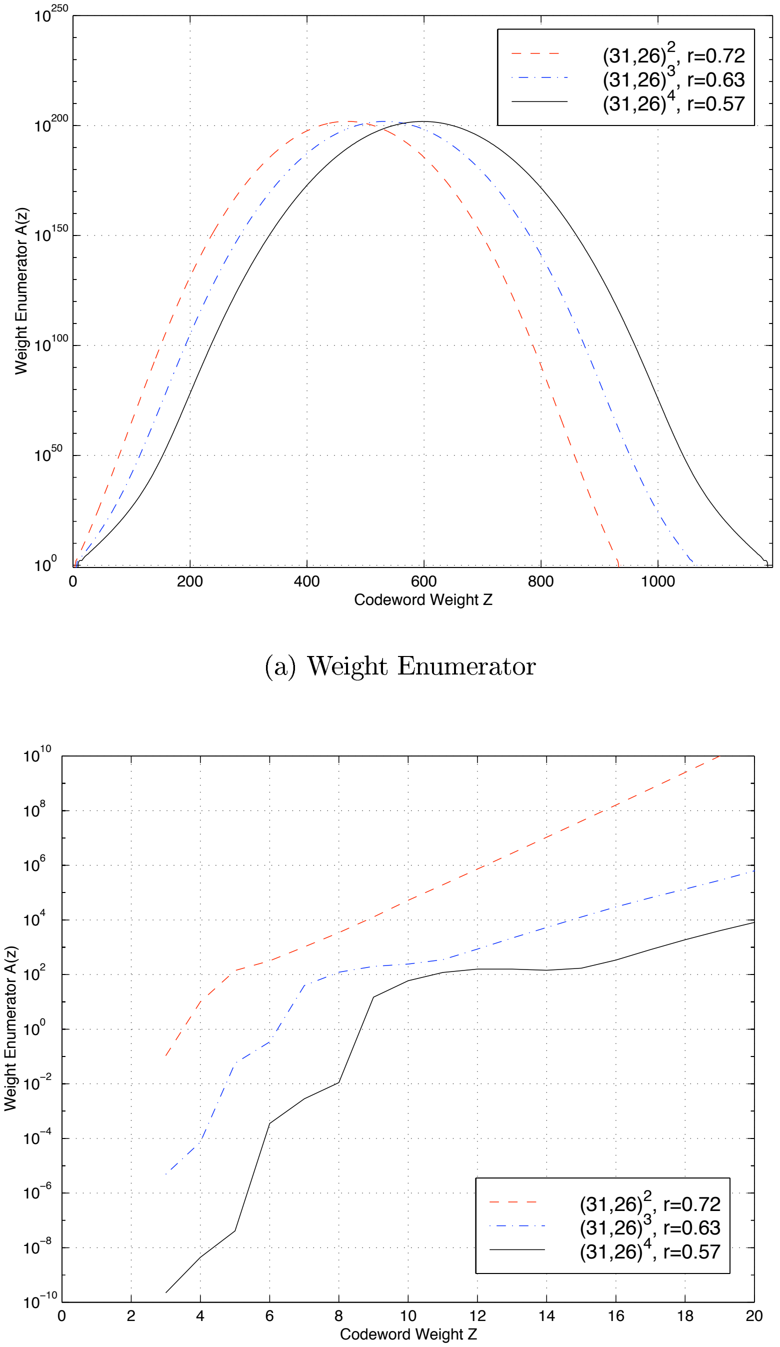
for AWGN channels, and

$$P_2(d, E_s/N_0) = [\frac{1}{2}(1-\mu)]^d \sum_{k=0}^{d-1} \binom{d-1+k}{k} [\frac{1}{2}(1+\mu)]^k \tag{3.22}$$

for flat Rayleigh fading channels, respectively, where μ is given by

$$\mu = \sqrt{\frac{E_s}{E_s + N_0}}.$$

Note that although a size K^2 product code is used as an example to explain the analysis in this chapter, the same analysis can also be applied to arbitrary information array size $K_1 \times K_2$.



(a) Weight Enumerator

(b) Weight Enumerator for Low Weight Codewords

Figure 3.6: Weight enumerators for $(31, 26)^M$ multiple product codes

3.5 Numerical Results

In this section, we compare the performance of a few rate compatible product codes and their turbo code counterparts over AWGN channels and Rayleigh fading channels. Both the product code decoders and the turbo code decoders are implemented with the Log-MAP BCJR algorithm [6, 10]. In the following example, we consider a two dimensional Hamming product code $(31, 26)^2$ with information block length 676 bits and rate 0.72. Lower rate codes $(31, 26)^3$ of rate 0.63 and $(31, 26)^4$ code of rate 0.57 are constructed with the S-random interleaving scheme described in Section 3.3.

Turbo codes of the same rates, same block sizes, and similar decoding complexity are constructed for performance comparison. The construction procedure will be described in Chapter IV. A family of rate compatible turbo codes can be easily obtained by periodically puncturing a low rate code, and the resulting codes can be decoded with a single decoder [71]. In this example, punctured turbo codes of rates 0.72, 0.63, and 0.57 are obtained by puncturing a rate 1/3 turbo code of 676 information bits.

The choice of turbo code constituent convolutional code is critical for both error performance and decoding complexity. Because BCJR algorithm is used in decoding both product codes and turbo codes, a good benchmark of complexity is the number of states in the decoding trellis. Since the $(31, 26)$ Hamming code has 32 states in the decoding trellis, the choice of turbo code constituent code is limited to constraint length 5 or less so that the decoding trellis has less than or equal to 32 states. It has been observed that for rate 1/3 turbo codes smaller constraint length results in better performance at low SNR, and larger constraint length results in better performance at high SNR [43]. However, for short frame length and high code rate, the statement is no longer true. In Figure 3.7, two rate 0.72 turbo codes of length 676 bits are simulated over AWGN channels. Apparently, the code with constraint length 5 outperforms the code with constraint length 4. The generator polynomial of the family of constraint length 5 turbo codes is chosen as $\{1, \frac{57}{75}, \frac{57}{75}\}$. This polynomial was shown to maximize the most critical design parameter of a turbo code, the *effective free-distance*, in [28]. The effective free-distance is the minimum Hamming weight of code sequences due to weight 2 input sequences [9]. In our simulation, 6 iterations are used for decoding product codes, and 10 iterations are used for decoding turbo codes.

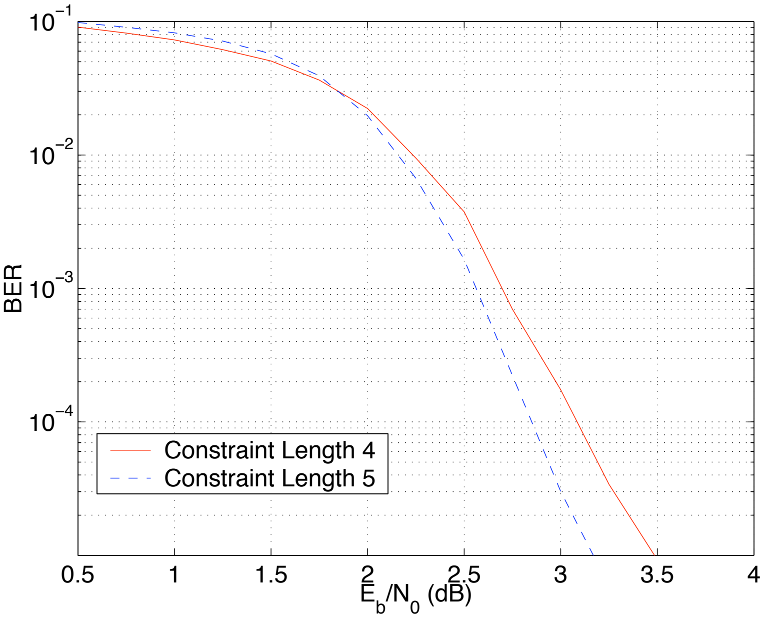


Figure 3.7: BER of rate 0.72 turbo codes with constraint length 4 and 5 over AWGN channels. (block length: 676 bits)

Simulation results for the 2D, 3D, and 4D product codes over AWGN channels are presented in Figures 3.8 and 3.9. The union bounds on BER and PER performance, as given by (3.19) and (3.20), are evaluated to verify the simulation results. The union bounds are observed to be tight at high SNR, and diverge when SNR goes below the cut-off rate threshold. To achieve a BER of 10^{-3}, the required E_b/N_0 for the 4D product code is 1.5 dB, which is 0.2 dB and 0.8 dB lower than the E_b/N_0 requirements for the 3D and 2D codes, respectively. For a target BER of 10^{-8}, the coding gain of the 4D product code increases to 0.8 dB and 2.3 dB, respectively. To achieve a PER of 10^{-2}, the required E_b/N_0 for the 4D product code is 1.7 dB, which is 0.3 dB and 1.5 dB better than the E_b/N_0 requirements for the 3D and 2D codes, respectively. For a target PER of 10^{-6}, the coding gain increases to 0.8 dB and 2.2 dB, respectively.

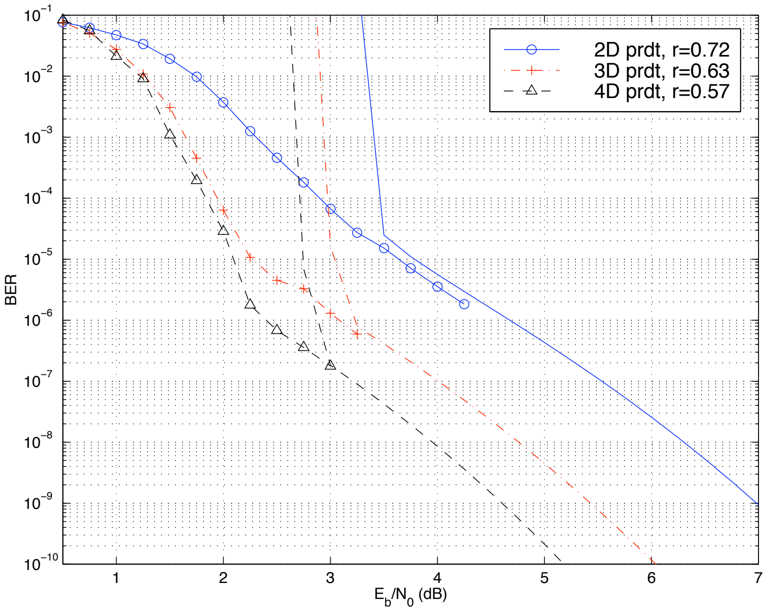


Figure 3.8: Union bounds and simulations for the bit error rate of multiple product codes over AWGN channels. (block length: 676 bits)

Simulation results for the product codes over flat Rayleigh fading channels are presented together with the union bounds in Figures 3.10 and 3.11. Similar to the results over AWGN channel, the union bounds are shown to be tight at high SNR, and diverge at SNR below the cut-off rate threshold. To achieve a BER of 10^{-3}, the required E_b/N_0 for the 4D product code is 4.0 dB, which is 0.4 dB and 1.7 dB lower than the E_b/N_0 requirements for the 3D and 2D codes, respectively. For a target

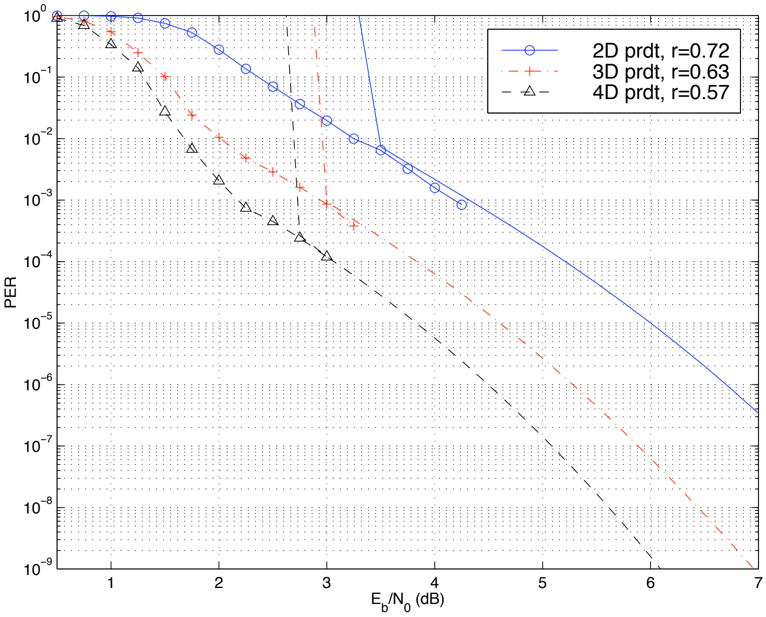


Figure 3.9: Union bounds and simulations for the packet error rate of multiple product codes over AWGN channels. (block length: 676 bits)

BER of 10^{-6}, the gain increases to 1.3 dB and 5.5 dB, respectively. To achieve a PER of 10^{-2}, the required E_b/N_0 for the 4D product code is 4.2 dB, which is 0.5 dB and 3.1 dB lower than the E_b/N_0 requirements for the 3D and 2D codes, respectively. For a target PER of 10^{-4}, the gain increases to 2.1 dB and 6.2 dB, respectively.

The BER performance of the product codes and the turbo codes over AWGN channels is shown in Figure 3.12. For the pair of rate 0.72 codes, the product code has up to 0.5 dB gain over the turbo codes at low SNR. The BER curves cross over at $E_b/N_0 = 3$ dB with a BER of 7×10^{-5}. For the rate 0.63 product and turbo codes, the product code has a consistent 0.4 dB gain over the turbo code. Since for many applications the BER requirement is greater than 10^{-4}, the performance of product codes is excellent for the block length we specified.

The PER performance for product codes and turbo codes over AWGN channels is presented in Figure 3.13. The product codes still outperform turbo codes at low SNR, but the crossovers of curves occur at higher error rates. The rate 0.72 product

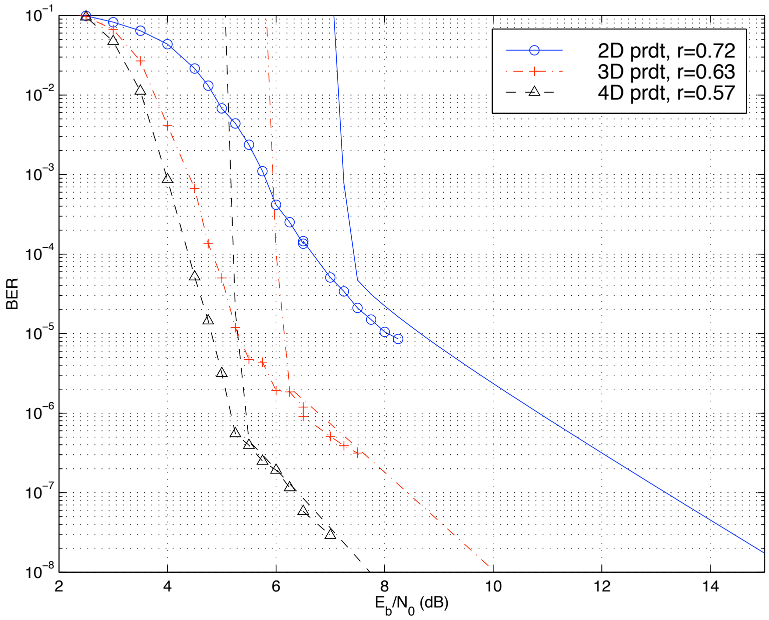


Figure 3.10: Union bounds and simulations for the bit error rate of multiple product codes over Rayleigh fading channels. (block length: 676 bits)

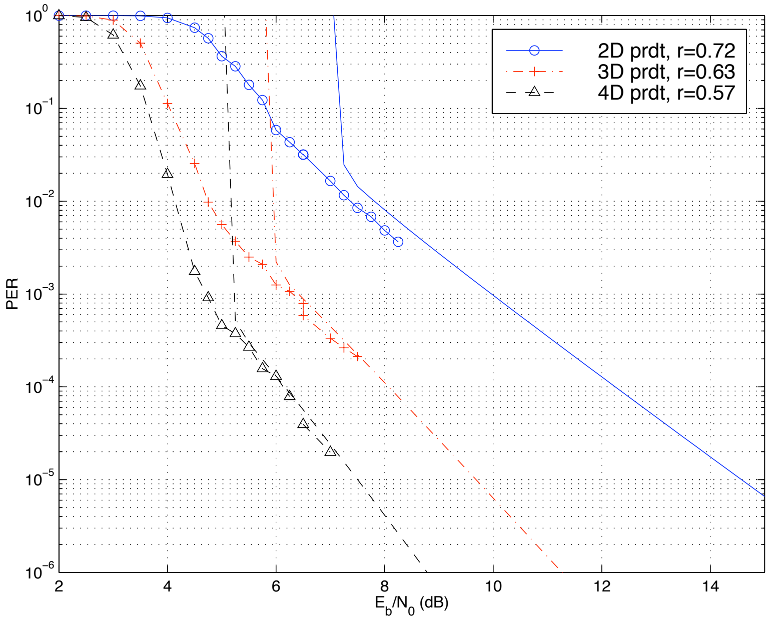


Figure 3.11: Union bounds and simulations for the packet error rate of multiple product codes over Rayleigh fading channels. (block length: 676 bits)

and turbo codes have performance crossover at $E_b/N_0 = 2.4$ dB with a PER of 8×10^{-2}. The rate 0.63 product and turbo codes have performance crossover at $E_b/N_0 = 2.3$ dB with a PER of 4×10^{-3}.

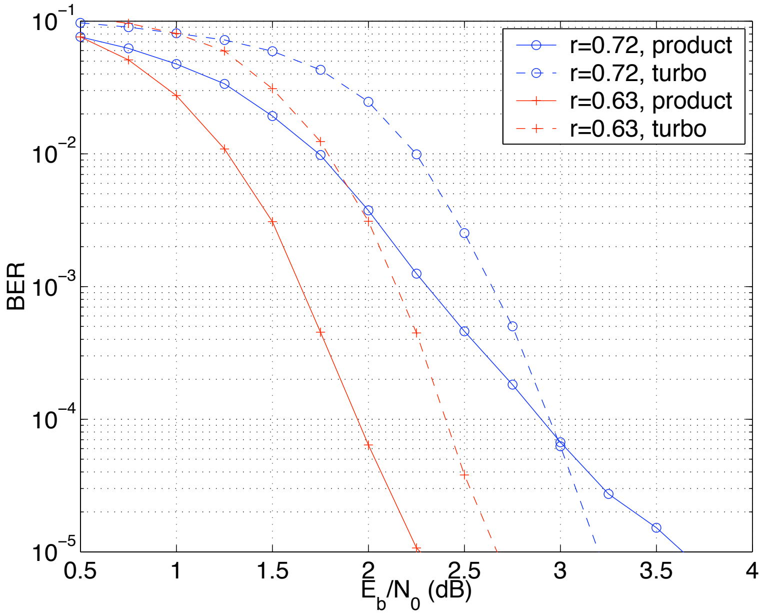


Figure 3.12: Packet error rates for turbo and multiple product codes over AWGN channels. (block length: 676 bits)

The BER performance of the product codes and the turbo codes over Rayleigh fading channels is shown in Figure 3.14. For the pair of rate 0.72 codes, the product code has up to 0.8 dB gain over the turbo codes at low SNR. The BER curves cross over at $E_b/N_0 = 7.3$ dB with a BER of 3×10^{-5}. For the rate 0.63 product and turbo codes, the product code has a consistent 0.7 dB gain over the turbo code. The BER performance of the product codes and the turbo codes over Rayleigh fading channels is shown in Figure 3.15. The product codes still outperform turbo codes for low SNR, but the crossovers of curves occur at higher error rates. The rate 0.72 codes have performance crossover at $E_b/N_0 = 6.4$ dB with a PER of 3.5×10^{-2}, the rate 0.63 codes have performance crossover at $E_b/N_0 = 5.7$ dB with a PER of 2.2×10^{-3}.

An incremental redundancy retransmission type-II hybrid ARQ scheme is imple-

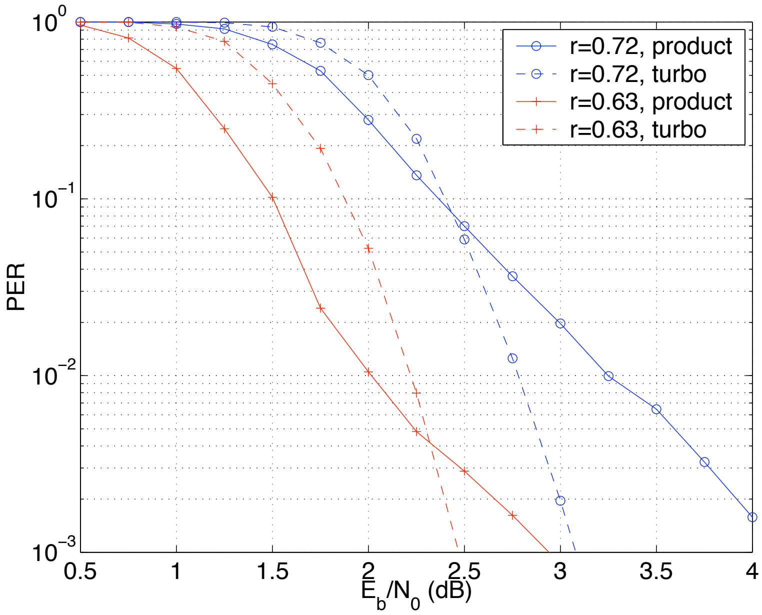


Figure 3.13: Bit error rates for turbo and multiple product codes over AWGN channels. (block length: 676 bits)

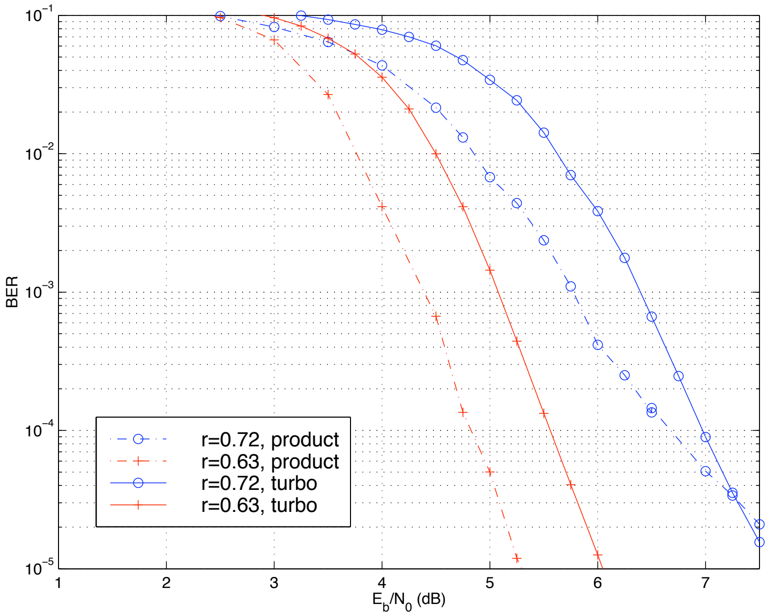


Figure 3.14: Packet error rates for turbo and multiple product codes over Rayleigh fading channels. (block length: 676 bits)

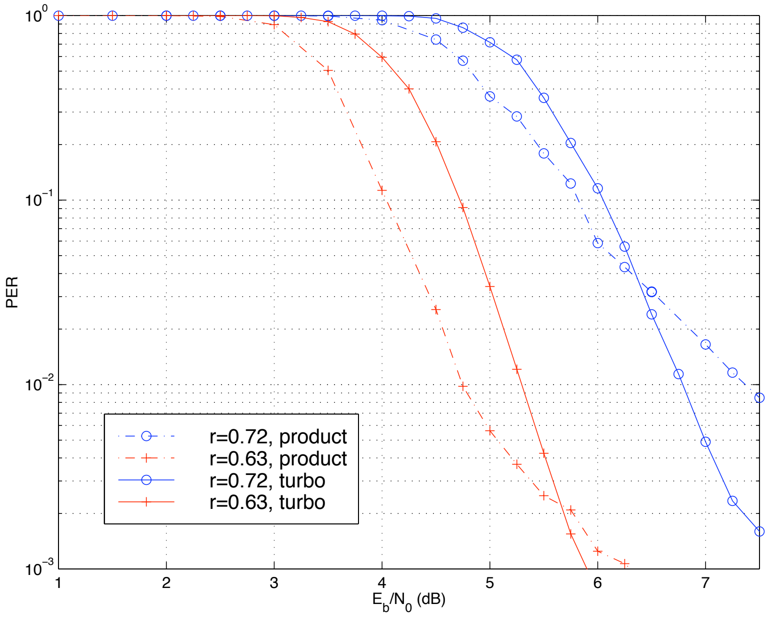


Figure 3.15: Packet error rates for turbo and multiple product codes over Rayleigh fading channels. (block length: 676 bits)

mented using a rate compatible multiple product (RCP) code family with rates {1, 0.93, 0.81, 0.72, 0.57, 0.36, 0.28}. In the first transmission, the 676 information bits are sent over the channel, with some number of these actually being CRC parity bits for error detection. The block diagram of the system is given in Figure 1.2. If a CRC detects an error, the receiver requests one parity bit from each Hamming code in the $(31, 26)^2$ product code, and combines the information and parity bits into a punctured product code of rate 0.93. One can also use alternative puncturing patterns, such as requesting two bits from half of the components codes, and none from the other half of the component codes. If the decoding of the punctured code is still incorrect, more redundant bits are requested which results in further reduction in the code rate. When the rate 0.28 code is not received correctly, the protocol is reset and restarts from the beginning. The simulation results of 50,000 packets of size 676 information bits over AWGN channels are shown in Figure 3.16. For E_s/N_0 below 2.5 dB, the normalized throughput, or spectral efficiency, of the ARQ scheme is larger than the computational cutoff rate bounds, and less than 2 dB from the channel capacity.

To make a comparison with punctured turbo coded ARQ, we also designed rate compatible punctured turbo (RCPT) codes of matching rates and simulated the throughput performance as shown in Figure 3.16. For E_s/N_0 above 1 dB, the RCPT/ARQ is observed to outperform RCP/ARQ. For lower SNR, the product codes are shown to have 0.1 dB to 0.5 dB gain over the competing turbo codes. The weakness of RCP at high rate is due to the punctured Hamming code, which has poor weight distribution, as shown in the previous section. However, the performance gap between RCP/ARQ and RCPT/ARQ should be discounted since high rate codes are not likely to be employed in any practical implementation of product coded ARQ. This becomes evident when considering the fact that the first redundancy block, which causes the rate to decrease from 1 to 0.93, contains only 52 bits. Taking into account the amount of overhead associated with each transmission, this small block size is very inefficient. Furthermore, the schemes considered here use seven transmissions to decrease to rate 0.3, which means a long delay will occur when the channel condition degrades. Thus, a more practical retransmission scheme will have a rate family $\{1, 0.72, 0.57, 0.28\}$, which corresponds to transmission block sizes $\{676, 260, 260, 1196\}$. In this case, product coded ARQ will have comparable performance to turbo coded ARQ.

3.6 Conclusions

In this chapter, we proposed an effective method of constructing multiple product codes of various rates. The decoding complexity increases linearly with the code length and dimensions. Weight enumerators and union bounds are derived for the

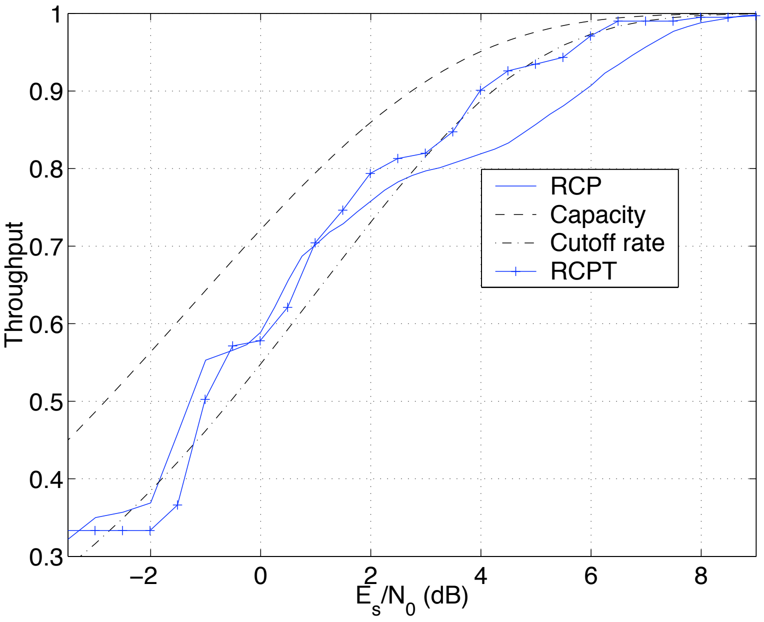


Figure 3.16: Normalized throughput for IRR type-II hybrid ARQ schemes based on rate compatible product codes and rate compatible punctured turbo codes. (block length: 676 bits)

proposed product codes. The resulting codes are shown to outperform S-random interleaved turbo codes of the same rate and comparable complexity for a block length of 676 bits when the target BER is greater than 10^{-4}. The claim that for a given BER, there is a threshold rate, *above* which product codes are better and *below* which turbo codes are better [40], is found to be not always true. For example, for a BER of 10^{-5}, the rate 0.72 turbo code is better than the $(31, 26)^2$ product code, but the rate 0.63 turbo code is about 0.5 dB worse than its product code counterpart over AWGN channels.

Furthermore, a family of rate compatible product codes are used to implement an IRR type-II hybrid ARQ scheme. This ARQ scheme achieves throughput above cut-off-rate bounds and within 2 dB of the channel capacity bound for SNR less than 5.5 dB over AWGN channels. Puncturing is used to obtain higher rate product codes but results in poor performance compared with turbo codes of the same rate.

This result suggests that although Hamming codes are good component codes for the original product codes, they are not robust under puncturing. Designing component codes with good high rate subcodes will be a key to the construction of high rate punctured product codes in the future.

CHAPTER IV

Rate Adaptive Turbo Coding in Multi-User Systems

In the previous two chapters, we studied ARQ protocols and rate adaptive coding for ARQ protocols in single user systems. Since the adaptive coding schemes discussed in this thesis are most likely to be used in a multi-user system, in this chapter, we apply a few rate adaptive ARQ protocols to a more practical wireless network setting and examine their performance. The multi-user wireless system considered here is an asynchronous direct-sequence code-division multiple-access (DS-CDMA) data network. The channel models considered include both shadowing and frequency selective fading.

The ARQ protocols and coding schemes investigated in this chapter are chosen such that they are of most practical value. Turbo codes have been specified in most third generation cellular systems for their powerful error correcting capability and availability on market. Hence, all systems considered in this chapter employ turbo codes as the FEC codes. As in previous chapters of this thesis, the conventional type-I hybrid ARQ schemes and the type-II hybrid ARQ schemes with incremental redundancy retransmission are compared. In addition, a metric combining type-I hybrid ARQ scheme is considered for the tradeoff between complexity and performance.

Besides the average delay and energy efficiency metrics used in the evaluation of single user systems, we introduce the total network throughput as a new performance metric. The total network throughput is defined as the product of the number of users in the network and the average throughput of each individual user.

4.1 Introduction

The advanced ARQ protocols and adaptive coding techniques investigated in previous chapters are indispensable in the next generation multi-user wireless data networks. Cellular systems designed for voice service typically require a frame error rate of 0.01 without retransmission and achieve spectral efficiency in the range of 0.03-0.05 b/s/Hz/sector [56]. One of the enabling techniques that allow the systems to achieve much higher capacity is ARQ. With ARQ schemes, the system requirement on frame error rate could be less stringent than 0.01, hence, the systems can operate at a lower signal-to-interference plus noise ratio (SINR) than that of the current cellular systems.

Metric combining type-I hybrid ARQ is a robust modified ARQ scheme that requires minimal additional complexity [15, 23]. In metric combining type-I hybrid ARQ schemes, when a decoding error occurs, the frame is retransmitted. The receiver then averages the demodulator output metrics from all received copies of the frame and attempts to decode with the combined frame. The basic idea behind metric combining is to improve the effective channel SINR by combining multiple copies of the received frame. The metric combining algorithms can also be viewed as concatenating a family of repetition codes of rates $\{1, \frac{1}{2}, \frac{1}{3}, \cdots\}$ and a fixed-rate FEC outer code. Instead of using repetition codes, there are usually more flexible ways to construct error correcting codes of various rates, such as rate compatible FEC codes.

A rate compatible FEC code family is a collection of codes such that all coded bits of higher rate codes are contained in lower rate codes. Upon each retransmission, the encoder transmits new redundant bits so that a stronger lower rate code is formed by combining frames from all transmissions.

In third generation DS-CDMA standards, wideband CDMA (WCDMA) and CDMA2000, type-II hybrid ARQ schemes based on rate compatible punctured turbo (RCPT) codes are being considered to improve the system bandwidth efficiency. In [1, 71], RCPT codes were constructed and evaluated for hybrid ARQ schemes in a single user communication system over AWGN channels. No multiple access interference (MAI) was considered and the ARQ scheme was evaluated with the local throughput. In [44], we investigated the performance of Reed-Solomon RCPT coded ARQ protocols in asynchronous DS-CDMA data networks over memoryless fading channels. The delay and network throughput performance were investigated for different ARQ schemes. In this chapter, the performance of turbo-coded ARQ protocols, including type-I hybrid ARQ, metric combining type-I hybrid ARQ, and RCPT type-II hybrid ARQ, is evaluated over correlated shadowing and fading channels.

Simulations are conducted to study the performance of the turbo coded ARQ protocols. For turbo codes, the union bounds and Gallager bounds diverge at low signal-to-noise ratio [9, 30]. The tangential sphere bounds do not diverge but are not valid bounds for any specific code [74]. Hence, simulation is the only effective way to study turbo code based ARQ systems.

The remainder of the chapter is organized as follows. In Section 4.2, the system model and the mathematical model for fading and shadowing processes are described. In Section 4.3, the structure of turbo codes and the construction of RCPT codes are discussed. In Section 4.4, the detailed retransmission protocols and the complexity comparison between different ARQ schemes are presented. In Section 4.5, we evaluate the performance of a few ARQ protocols for DS-CDMA data networks under fading channels. Section 4.6 provides a summary of this chapter.

4.2 System Model

The multiple-access network addressed in this chapter involves K mobile transmitters and K receivers at the base station. The baseband system block diagram of a transmitter-receiver pair is shown in Figure 1.2. The mobile transmitter is composed of a CRC encoder, a rate compatible punctured turbo code encoder, a bit interleaver, and a spread spectrum BPSK modulator. The information bit stream is first broken into sequences of length L_i. Then each information sequence is encoded by a CRC encoder for error detecting. The resulting sequence is of length $L_i + L_{CRC}$, where L_{CRC} equals 16 in ANSI and CCITT standards. The CRC protected sequence is encoded by a rate 1/3 turbo code and buffered for future transmission. The code bits are then punctured and transmitted in different manners depending on the ARQ protocols, which will be discussed in Sections 4.3 and 4.4.

The data sequence for the kth user, $b_k(t)$, consists of a sequence of coded bits of duration T of length L_c. Spreading is implemented by multiplying a binary sequence $a_k(t)$ with the data sequence $b_k(t)$,

$$\begin{aligned} b_k(t) &= \sum_{l=1}^{L_c} b_l^{(k)} p_T(t - lT), \quad b_l^{(k)} \in \{+1, -1\} \\ a_k(t) &= \sum_{l=-\infty}^{\infty} a_l^{(k)} p_{T_c}(t - lT_c), \quad a_l^{(k)} \in \{+1, -1\} \end{aligned}$$

where $T_c = T/N$, and $p_T(t)$ is a rectangular pulse given by

$$p_T(t) = \begin{cases} 1, & t \in [0, T] \\ 0, & \text{otherwise} \end{cases} .$$

In this chapter, we model the spreading sequence $\{a_l^k\}$ for each user k as an independent and identically distributed binary sequence. The low-pass equivalent of the transmitted signal is

$$s_k(t) = \sqrt{2P_k}a_k(t)b_k(t).$$

The transmitter receiver pair is assumed to suffer from both fast fading and log-normal shadowing. The shadowing model is described in Chapter II. The shadowing process is assumed to be constant over a frame duration $\beta^{(k)}(t) = \beta^{(k)}$ and the correlation between the logarithm of the shadowing levels of adjacent frames is modeled as an exponentially decaying function [37]. Mathematical model of the shadowing process can be found in Section 2.2.2. The fast fading process is modeled as a constant fading process over each symbol and independent from symbol to symbol. This model is accurate under the assumption of a sufficiently large interleaver size. In a wideband CDMA system, the signal bandwidth is larger than the coherent bandwidth of the fading channels so that the multipath components in the received signal can be resolved [82]. We assume that there are L_k resolvable, independent, equal strength paths for user k's channel. The low-pass equivalent of the received signal can then be written as

$$r(t) = \sum_{k=1}^{K}\sum_{j=1}^{L_k} \alpha_j^{(k)}\beta^{(k)}e^{j\varphi_j^{(k)}-\omega_c\tau_j^{(k)}}s_k(t-\tau_j^{(k)}) + n(t)$$

where ω_c is the carrier frequency, $\{\alpha_j^{(k)}\}$ are the fast fading components that follow the Rayleigh distribution as given by (2.3). The fading processes on different paths are assumed to be independent of each other. The random phases, $\{\varphi_j^{(k)}\}$, are independent random variables following the uniform distribution on $[0, 2\pi]$. For asynchronous DS-CDMA, the delays $\{\tau_j^{(k)}\}$ are independent uniformly distributed random variables over intervals of length $2T_c$, and the minimum separation between the delays $\{\tau_1^{(k)}, \tau_2^{(k)}, \cdots, \tau_{L_k}^{(k)}\}$ is also $2T_c$, such that the paths of the same user's channel are actually distinguishable. The background noise $n(t)$ is a white complex Gaussian random process with double-side power spectrum density N_0.

Two different power control assumptions are investigated. In the first case, we assume that *average* power control is employed over a frequency-selective fading channel. By average power control, we mean that the power control scheme can vary P_k to compensate for the effect of path loss and shadowing β_k so that the effective received power $P_k\beta_k$ is the same for all users. The performance of this system is equivalent to a system over a frequency selective fading channel without shadowing and power control. In the second case, we assume that power control is not available. In this chapter, we will refer to the first case as the fading channel case and the second one as the shadowing channel case, because with average power control the shadowing process has no effect on the system performance.

As shown in Figure 1.2, the CDMA receiver consists of a coherent demodulator, a bit deinterleaver, an RCPT decoder, and a CRC decoder. The receiver is provided with perfect channel side information, i.e., the Rayleigh fading level on each path. As in previous two chapters, we assume perfect error detection, error-free feedback channel, and negligible header compared with payload.

In the rest of the section, we derive the statistics and the effective signal-to-interference and noise ratio (SINR) at the demodulator output for the average power control scenario. For the case without power control, since the shadowing process is constant over each frame for one user, the SINR is simply β_k times the SINR in the previous case. The results enable us to simulate a multi-user system performance without actually simulating interference from each user. Note that the results presented here are for a single cell system, since no out of cell interference is considered.

The bit metric for the kth user's jth channel $Z_j^{(k)}$ is given by

$$Z_j^{(k)} = Re\left\{\int_{\tau_j^{(k)}}^{\tau_j^{(k)}+T} r(t)\sqrt{\frac{2}{T}}a_k(t-\tau_j^{(k)})e^{j(\omega_c\tau_j^{(k)}-\varphi_j^{(k)})}dt\right\}. \tag{4.1}$$

When a large number of users are present in the system, multiple-access interference can be approximated as Gaussian noise. It is straightforward to show that $Z_j^{(k)}$ is a Gaussian random variable conditioned on the fading level of the kth user's jth channel $\alpha_j^{(k)}e^{j\varphi_j^{(k)}}$, delay on that channel $\tau_j^{(k)}$, and the transmitted bit $b_j^{(k)}$. Without loss of generality, we only show the statistics of $Z_j^{(k)}$ in the case for $j=1$ and $k=1$ [78],

$$\begin{aligned}\overline{Z_1^{(1)}} = E[Z_1^{(1)}|\alpha_1^{(1)},\varphi_1^{(1)},\tau_1^{(1)},b_1^{(1)}=+1] &= \alpha_1^{(1)}\sqrt{E}\\ \sigma^2_{Z_1^{(1)}} = Var(Z_1^{(1)}|\alpha_1^{(1)},\varphi_1^{(1)},\tau_1^{(1)},b_1^{(1)}=+1) &= E\sum_{k=2}^{K}\sum_{j=1}^{L_k}\frac{\overline{(\alpha_j^{(k)})^2}}{3N}+E\sum_{j=2}^{L_k}\frac{\overline{(\alpha_j^{(1)})^2}}{3N}+N_0/2\end{aligned}$$

where E equals $P_k\beta_kT$, and $P_k\beta_k$ is a constant for all users. In this chapter, we assume that each user's channel has 3 equal strength paths, thus, $\sigma^2_{Z_1^{(1)}}$ can be simplified as

$$\sigma^2_{Z_1^{(1)}} = Var(Z_1^{(1)}|\alpha_1^{(1)},\varphi_1^{(k)},\tau_1^{(k)},b_1^{(1)}=+1) = (3K-1)\frac{2\sigma_1^2E}{3N}+N_0/2$$

The signal-to-noise ratio per branch of a rake receiver is

$$\gamma_1^{(1)} = \frac{\overline{Z_1^{(1)}}^2}{\sigma^2_{Z_1^{(1)}}}. \tag{4.2}$$

4.3 Rate Compatible Punctured Turbo Codes

Since the structure of turbo codes has been described in detail in many papers, only a brief description of turbo codes will be given here. The key component of the ARQ protocol, the rate compatible punctured turbo code, will be introduced.

4.3.1 Turbo Codes

With reference to Figure 4.1, we now describe the structure of the punctured turbo encoder. The original turbo codes [12] are parallel concatenated convolutional codes in which the information bits are first encoded by an RSC (recursive systematic convolutional) encoder, and then, after passing through an interleaver, are encoded by a second RSC encoder. A codeword of a turbo code is composed of an information sequence and parity check sequences from the two encoders. The RSC codes have the property of infinite impulse response so that even low-weight input sequences might generate a high-weight output if the distance between the 1's is far enough.

The interleavers employed in turbo encoders should have the capability of breaking up low-weight input sequences so that the permuted sequences have large distances between the 1's. Hence, even if the first encoder generates a low-weight output sequence due to a certain low-weight input pattern, the other encoders are likely to generate high-weight output sequences and the resulting overall codeword is still of high weight. The S parameter is an important parameter used to measure the capability of spreading low-weight sequences of an interleaver. An interleaver is said to have spreading parameter S if, wherever two bits are within distance S in the original sequence, their positions are at least separated by S in the output sequence. In our simulations, S-random interleavers [29] are employed. An S-random interleaver is designed by reordering a pseudo-random interleaver such that it achieves as high an S-parameter as possible. In the example we are going to show, the interleaver length is 424 bits with an S parameter equal to 14.

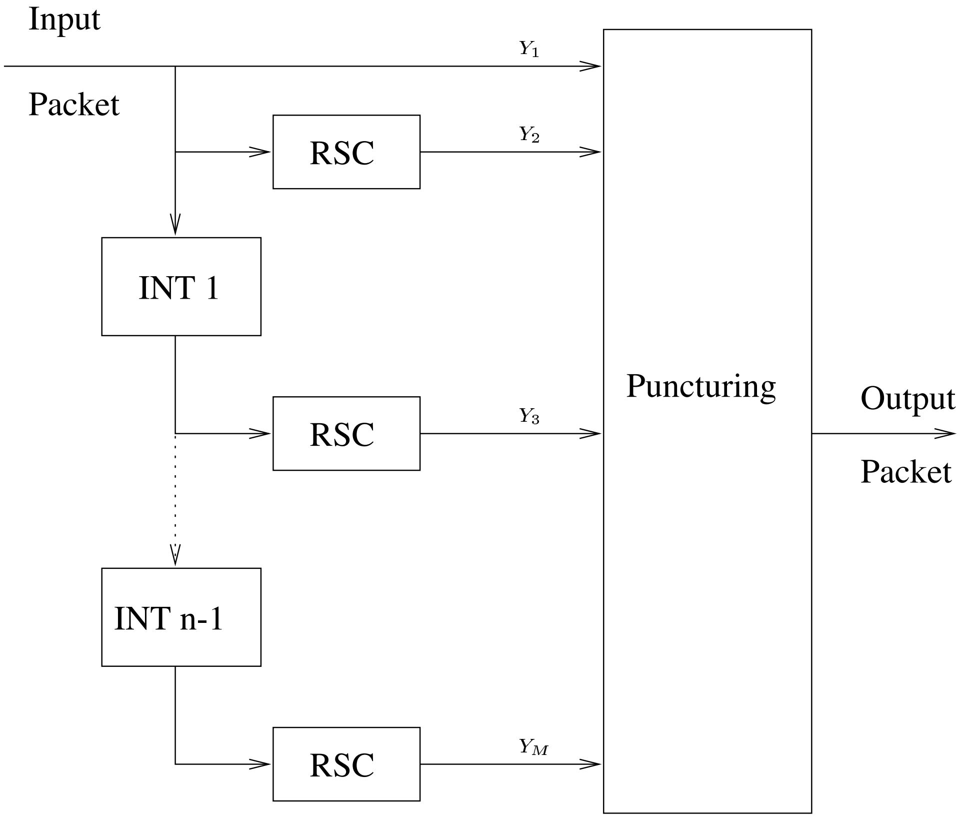


Figure 4.1: Punctured turbo code encoder

4.3.2 Puncturing Pattern

A family of RCPT codes can be obtained by puncturing the coded bits of a rate $\frac{1}{n}$ turbo code. Each output bit stream of the turbo code is punctured according to a puncturing pattern, which has period p. The puncturing patterns for n bit streams are represented by an $n \times p$ matrix $\mathbf{P}$, where the ith row of $\mathbf{P}$ denotes the puncturing pattern for the ith bit stream. A possible set of puncturing matrices $\{\mathbf{P}_0, \mathbf{P}_1, \cdots, \mathbf{P}_4\}$ corresponding to RCPT codes with puncturing period $P = 2$ and rates $\{1, \frac{2}{3}, \frac{1}{2}, \frac{2}{5}, \frac{1}{3}\}$

is given by

$$\overbrace{\begin{pmatrix}1 & 1\\ 0 & 0\\ 0 & 0\end{pmatrix}}^{\mathbf{P}_0}\ \overbrace{\begin{pmatrix}1 & 1\\ 1 & 0\\ 0 & 0\end{pmatrix}}^{\mathbf{P}_1}\ \overbrace{\begin{pmatrix}1 & 1\\ 1 & 0\\ 0 & 1\end{pmatrix}}^{\mathbf{P}_2}\ \overbrace{\begin{pmatrix}1 & 1\\ 1 & 1\\ 0 & 1\end{pmatrix}}^{\mathbf{P}_3}\ \overbrace{\begin{pmatrix}1 & 1\\ 1 & 1\\ 1 & 1\end{pmatrix}}^{\mathbf{P}_4}\ \begin{array}{l}\leftarrow\ Y_1\\ \leftarrow\ Y_2\\ \leftarrow\ Y_3\end{array}$$

Optimization of puncturing patterns for RCPT codes has been investigated by Acikel *et al* [1] and Rowitch *et al* [71, 72, 73]. In [1], the generator polynomial, interleaver, and puncturing pattern of RCPT codes are jointly optimized with an exhaustive search of patterns that maximize the distance spectra due to input-weight 2, 3, and 4 error events. The optimal patterns obtained by this method are strongly coupled with the interleaver design, and may not yield good results for other interleavers. The complexity of the search algorithm also limits the puncturing pattern and interleaver to be relatively simple. In [73], a computationally efficient algorithm was proposed to derive the IOWE of punctured turbo codes based on the uniform random interleaver assumption [9]. With this method, the optimal puncturing patterns can be designed to maximize the *free-distance*, *effective free-distance* [9], or *average distance spectrum slope* [72] of RCPT codes. The puncturing patterns used in the previous and current chapters are obtained with the algorithm proposed in [73].

4.3.3 Turbo Decoding

The decoding algorithm for turbo codes is a sub-optimal iterative algorithm [12]. The turbo decoder of a punctured turbo code consists of n soft-output decoding modules associated with the n constituent codes in the encoder. At the first iteration, each decoding block calculates the likelihood for each information bit from the demodulation output. At the next iteration, each decoding block recalculates these reliabilities with not only the demodulation outputs but also the *extrinsic* likelihood information generated by other decoders in the previous iteration. After a number of iterations, a decision for each information bit is made based on the likelihood of that bit calculated by all the decoding modules.

The heart of the turbo decoder is the soft-output decoding module that generates *extrinsic* likelihood information. For different sub-optimality and implementation complexity, several algorithms are available to generate this estimation. Some well known algorithms are soft-output Viterbi algorithm (SOVA) [38], BCJR [6], and log-BCJR algorithm [10]. We adopt the log-BCJR algorithm for its robustness and optimal performance. In our implementation, the initial loglikelihood ratio input to the turbo decoder for a bit of the kth user $L(z_k)$, is computed by implementing a rake receiver. That is,

$$L(z_k) = \sum_{j=1}^{L_k} Z_j^{(k)} \sqrt{\gamma_j^{(k)}}$$

where L_k is the number of channels of the kth user, $\gamma_j^{(k)}$ is specified in (4.2), and $Z_j^{(k)}$ is given by (4.1).

4.4 Retransmission Protocols

In this section, the type-II hybrid ARQ protocol based on RCPT codes are described. The type-I hybrid ARQ protocols will not be discussed, since they have been described in details in the previous chapters and sections. Efficient type-II hybrid ARQ schemes can be implemented with an appropriate choice of RCPT codes for each retransmission. Since our concerns are mainly the performance comparison between ARQ protocols using fixed-rate FEC codes and RCPT codes, the evaluation of selective-repeat ARQ with infinite buffer size is sufficient to demonstrate the performance differences. For the simplicity of analysis, an error-free feed-back channel, a perfect error detecting code, and correct decoding of the header information of every received frame are assumed. As an example, a type-II hybrid ARQ protocol with rate $\{1, 1/2, 1/3\}$ RCPT codes is given as follows:

Transmitter:

① Encode the ith frame with a rate $\frac{1}{3}$ turbo encoder, and buffer all coded bits. Transmit the bits specified in $\mathbf{P}_0$ for the first transmission.

② For each frame, if an ACK is received, remove the corresponding coded bits from the buffer. If a NAK is received, transmit additional bits in $\mathbf{P}_2$ which are not contained in $\mathbf{P}_0$.

③ If the second NAK is received, transmit additional bits contained in $\mathbf{P}_4$ but not in $\mathbf{P}_2$.

④ If another NAK is received, repeat Steps 1, 2, and 3 until an ACK is received.

Receiver:

① Assemble all the received bits according to the puncturing table and run the turbo decoding algorithm. If the CRC detects no errors in the turbo decoder output, send an ACK to the transmitter and return to step 1. However, if the CRC does detect errors, send a NAK to the transmitter and repeat this step. If three successive NAK's are returned, go to step 2.

② This step is only necessary if three consecutive unsuccessful decoding attempts have been made. At this point, the transmitter has received three consecutive NAK's and will retransmit the frame, starting with $\mathbf{P}_0$. Combine the newly received bits with the previously buffered bits, replace the old information in the buffer with the newly combined value, and decode. If the CRC detects no errors, then return an ACK and go back to step 1. Otherwise, assemble the received bits of the next transmission and decode. Repeat this procedure no more than three times.

③ If there are still errors after the transmitter has transmitted $\mathbf{P}_0$, $\mathbf{P}_2$, and $\mathbf{P}_4$ three times each, then discard all previously received code frames and go to step 1.

4.4.1 Complexity Comparison

As shown in the preceding section, the retransmission protocol of RCPT type-II hybrid ARQ schemes is more complicated than type-I hybrid ARQ and metric combining ARQ schemes. In type-I hybrid ARQ and metric combining type-I hybrid ARQ protocols, both the header and payload of a frame are encoded with the same FEC code. In RCPT type-II hybrid ARQ schemes, each sub-frame contains only a portion of a codeword, and the decoder needs to assemble the received bits from each sub-frame before decoding. Since the decoder can not proceed with decoding until the sub-frame number (such as $\mathbf{P}_0$, $\mathbf{P}_2$, and $\mathbf{P}_4$) is resolved, the header needs to be decoded separately. Thus, an additional pair of a powerful FEC encoder and decoder is required for the headers in RCPT type-II hybrid ARQ schemes.

The turbo encoder and decoder hardware complexity of RCPT type-II hybrid ARQ schemes is slightly higher than that of type-I hybrid ARQ and metric combining ARQ schemes. All protocols considered in this chapter share the same *base* turbo encoder and decoder. RCPT codes utilize binary masks at the turbo encoder output to select desired coded bits, and utilize the same masks to assemble the received word at the decoder. The complexity of this mask operation is negligible compared to the turbo decoder itself. Extra storage space is required by the receiver of RCPT type-II hybrid ARQ and metric combining ARQ to buffer the *soft* information of the erroneous sub-frames.

4.5 Numerical Results

Under the assumption that the number of users in the network is constant and each user has an infinite number of frames to transmit, we simulated 20,000 frames to obtain the performance of rate $\frac{1}{3}$ turbo-coded standard ARQ, turbo-coded metric combining ARQ, and RCPT type-II hybrid ARQ schemes. In the simulations, we use the size of a standard ATM cell (424 bits) as the size of the information frame. Moreover, we assume that the data rate is 1 Mbps, the mobile velocity is 20 m/s, and the feedback delay is 20 ms. The offered load is defined as

$$L = \frac{K}{N} \quad \text{users/chip},$$

where K is the number of users, and N is the spreading factor. For instance, in the second generation DS-CDMA standard the typical network load is about 0.25 user/chip [80]. For the time being, we assume the spreading gain is the same for all users in the cell. The overall normalized network throughput η is defined as the number of correctly decoded information bits per channel use, which is given by

$$\eta(\overline{\gamma}) = \frac{1}{J}\frac{K}{N},$$

where J is the average number of transmitted bits required to correctly decode one information bit, and $\overline{\gamma}$ is the signal-to-background noise ratio. Note that the throughput is equivalent to the system spectral efficiency given by

$$\eta = \frac{KR_b}{W} \qquad b/s/Hz,$$

where R_b is the information bit rate, and W is the signal bandwidth.

The energy efficiency is usually defined as the number of information bits successfully received at the cost of each Joule of energy consumed. Ignoring the power consumed in data processing, the energy efficiency is proportional to the reciprocal of the effective bit energy E_b Joules/bit. For a certain background noise power spectrum density N_0, we define the normalized energy efficiency as $\lambda(\overline{\gamma}) = \frac{N_0}{E_b}$. According to Shannon's information theory, the best achievable energy efficiency is 1.59 dB, i.e., 1.44 bits/Joule, with the corresponding spectral efficiency approaching zero.

Two families of RCPT codes are considered. The first code family, RCPT_{P8}, which was proposed in [71], has puncturing period 8. In RCPT_{P8}/ARQ, the transmitter transmits the uncoded information frame in the first transmission and sends incremental parity frames of size one-quarter of the information frame. For the RCPT_{P8}/ARQ scheme, the resulting code rates for successive transmissions are $\{1, 4/5, \cdots, 8/(8+2i), \cdots, 1/3\}$. The second code family with puncturing period 2, RCPT_{P2}, uses one uncoded transmission and two more parity check transmissions to reach the code rate $\frac{1}{3}$. Note that for RCPT_{P2} codes, the frame sizes of the second and third transmissions are identical to the information frame size, and the resulting code rates are $\{1, 1/2, 1/3\}$. Both code families employ a 4-state turbo code with generator polynomial $\frac{5}{7}$. Decoding decisions are made after ten iterations in the turbo decoder. The simulation parameters are summarized in Table 4.1.

We compare the performance of the four ARQ protocols listed above at a given average signal-to-background noise ratio, $\overline{\gamma} = 4.77$ dB, in Figures 4.2 to 4.7. For other signal-to-background noise ratios, similar results are obtained. The throughput under energy constraints, maximized over $\overline{\gamma}$ and offered network load, for fading and shadowing channels are shown in Figures 4.8 and 4.9, respectively.

Table 4.1: Simulation parameters for turbo coded ARQ schemes

	Channel Parameters	Primary Frame	Retransmission Frame
ARQ	Data rate = 1 Mpbs	1272 bits	1272 bits
MC/ARQ	Velocity = 20 m/s	1272 bits	1272 bits
RCPT$_{P2}$	Feedback delay = 20 ms	424 bits	424 bits
RCPT$_{P8}$	Shadowing = Urban [37]	424 bits	106 bits

Figure 4.2 demonstrates the throughput performance of the four ARQ schemes over the fading channel. The throughput of the type-I hybrid ARQ system achieves a maximum of 0.75 at $L = 2.5$, then decreases sharply with increasing values of L. The metric combining ARQ scheme has the same performance as the standard ARQ when the network load is less than 2.5, but is much more robust under severe interference. Both RCPT type-II hybrid ARQ schemes outperform type-I hybrid ARQ and metric combining type-I hybrid ARQ schemes for any offered load in terms of network throughput and robustness. Note that most current DS/CDMA cellular systems operate at $L \ll 1$, in which region RCPT type-II hybrid ARQ schemes have significantly larger throughput than the throughput of ARQ and metric combining type-I hybrid ARQ. Between the two RCPT schemes, the RCPT$_{P8}$/ARQ scheme has slightly better throughput performance than the RCPT$_{P2}$/ARQ scheme for network load $L < 3$, while the throughput of both RCPT type-II hybrid ARQ schemes are virtually the same for $L > 3$.

In Figure 4.3, the energy efficiency of the four protocols is compared. The energy efficiency, in general, decreases with increasing network load (increasing interference level). At $L = 0.4$, which is the minimum offered load we simulated, RCPT$_{P8}$ has $\lambda = 0.19$, and the energy efficiency of RCPT$_{P2}$, metric combining type-I hybrid ARQ, and type-I hybrid ARQ is 0.17, 0.11, and 0.11, respectively. When $L > 3$, the energy

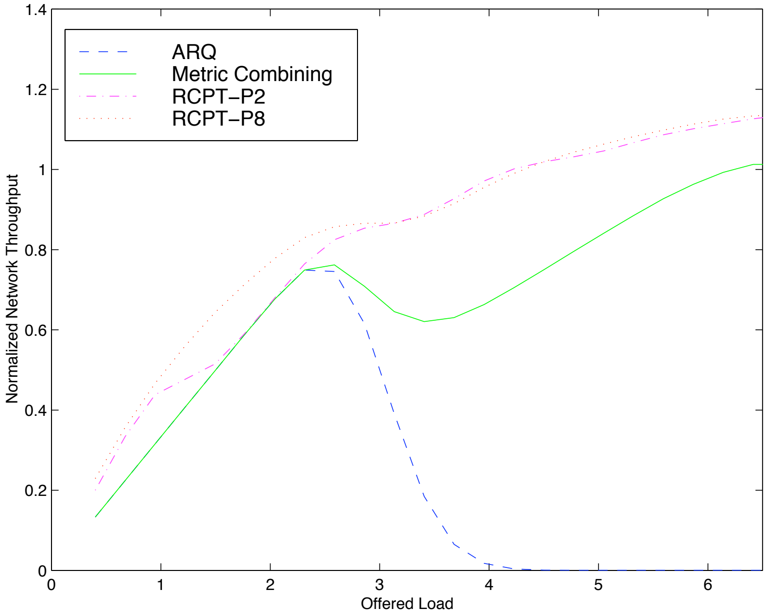


Figure 4.2: Total network throughput of turbo coded ARQ schemes for an asynchronous DS-CDMA network over fading channels

efficiency of type-I hybrid ARQ approaches 0 quickly, while all other three schemes maintain $\lambda > 0.05$. At higher signal-to-background noise ratios, the plateau of ARQ energy efficiency performance extends to higher load, but the relative position of all curves remains the same. For lower signal-to-background noise ratio, the right edge of the plateau moves to lower offered load until it disappears, which means that the type-I hybrid ARQ may fail to deliver any information if the noise level is too high even if there is no multiple access interference.

In Figure 4.4, the average delay of the ARQ schemes over fading channels is shown. We observe that the RCPT_{P2} scheme has only about one-third of the delay of the RCPT_{P8} scheme. This complies with our intuition that RCPT codes with shorter puncturing periods (and thus larger parity check frame) need fewer retransmissions to lower the code rate to $\frac{1}{3}$ than the codes with smaller parity check frame size. The delay of the standard ARQ scheme diverges to infinity when $L > 3$, and the delay of metric combining scheme is the lowest of all four schemes.

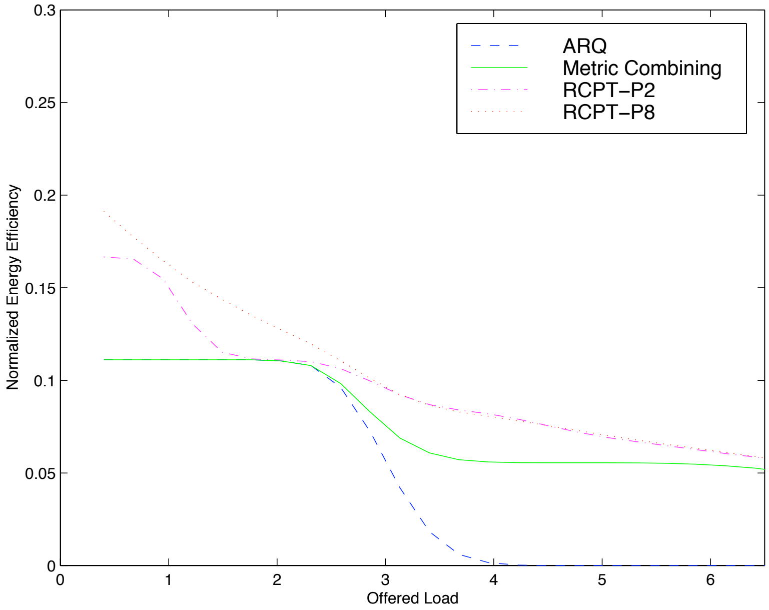


Figure 4.3: Energy efficiency of turbo coded ARQ schemes for an asynchronous DS-CDMA network over fading channels

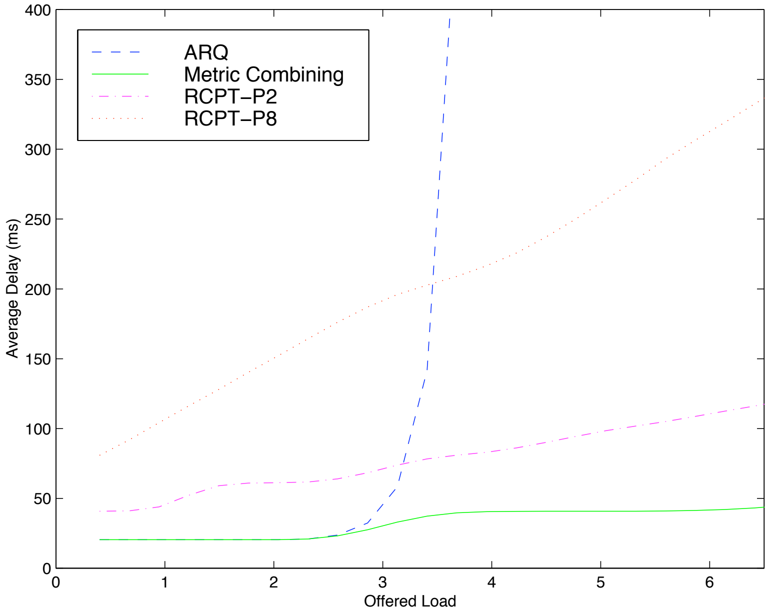


Figure 4.4: Average delay of turbo coded ARQ schemes for an asynchronous DS-CDMA network over fading channels

Figures 4.5-4.7 show the simulation results over shadowing channels. It is demonstrated that all performance curves are smoother than their counterparts over fading only channels due to the SINR averaging effect of the shadowing process. We also notice that the network throughput for all four ARQ schemes is smaller at light traffic loads and larger at heavy network loads than their counterparts over the fading channels. The reason behind this phenomenon is that at light traffic loads the system performance is dominated by the error events caused by deep fades. On the other hand, at very heavy load, most throughput comes from the positive fading events that cause significantly increased SINR. Because the shadowing process sometimes generates deep fades and sometimes boosts the signal level, on average, it degrades the system performance at light loads (negative fades hurt performance and positive fades have little effect on performance), and improves the performance at heavy loads (negative fades have little effect on performance and positive fades are beneficial).

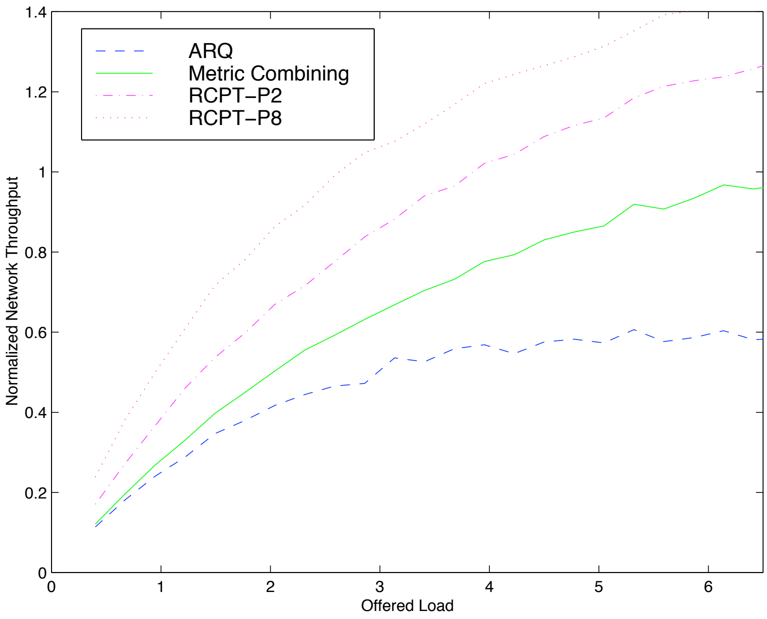


Figure 4.5: Total network throughput of turbo coded ARQ schemes for an asynchronous DS-CDMA network over shadowing channels

In Figures 4.8 and 4.9, the maximum normalized network throughput is plotted

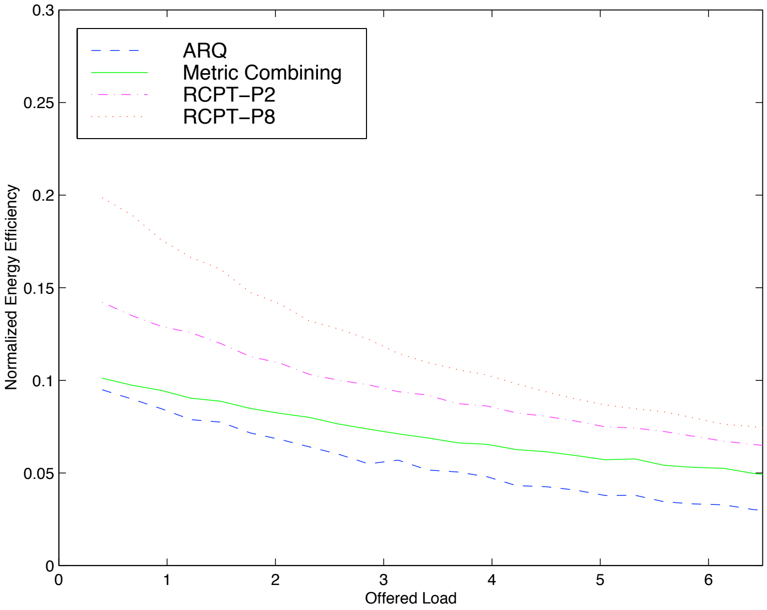


Figure 4.6: Energy efficiency of turbo coded ARQ schemes for an asynchronous DS-CDMA network over shadowing channels

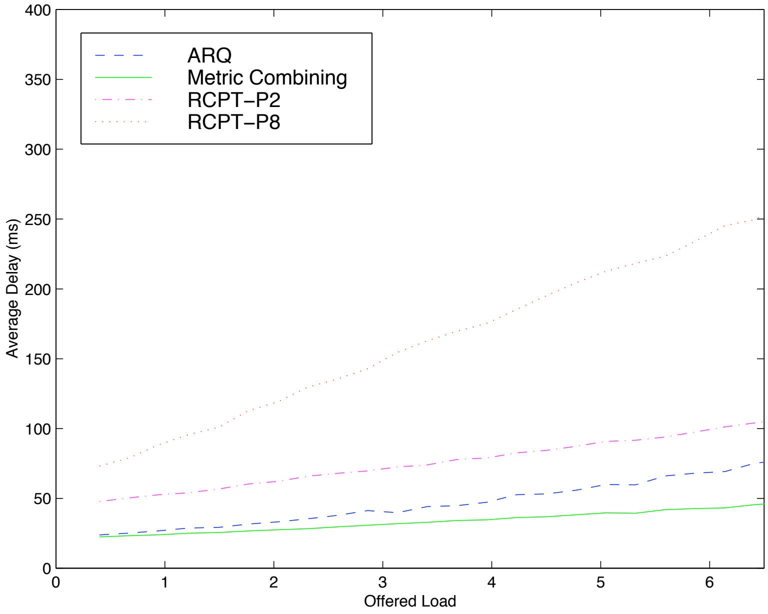


Figure 4.7: Average delay of turbo coded ARQ schemes for an asynchronous DS-CDMA network over shadowing channels

against the energy efficiency. Given a minimum energy efficiency requirement, the maximum normalized network throughput is obtained by optimizing over the signal-to-background noise ratio and the offered network load, which translate into the transmitter power and the number of users admitted into the system. It is observed that the maximum achievable network throughput decreases as the energy efficiency requirement increases. In both plots, RCPT type-II hybrid ARQ schemes are shown to achieve larger throughput than metric combining type-I hybrid ARQ and type-I hybrid ARQ with any energy constraint. We also notice that the optimal throughput is mostly achieved by the lowest signal-to-background noise ratio $\overline{\gamma}$ used in the simulations. Figure 4.8 corresponds to an average power controlled data network where shadowing is compensated through power control. Figure 4.9 corresponds to the case where no power control is available and correlated shadowing is considered. The gaps between type-I hybrid ARQ system and RCPT type-II hybrid ARQ systems can be interpreted as the difference between the minimum $\overline{E}_b/N_0$ that different systems can operate on. All systems, however, share the same slope, because of the same underlying interference model.

4.6 Conclusions

Rate $\frac{1}{3}$ turbo-coded type-I hybrid ARQ, metric combining type-I hybrid ARQ, and RCPT type-II hybrid ARQ schemes were investigated for the uplink traffic of asynchronous DS-CDMA data networks on multipath fading and shadowing channels with perfect side information. Compared with rate $\frac{1}{3}$ turbo-coded type-I hybrid ARQ and metric combining ARQ schemes, the RCPT type-II hybrid ARQ schemes provide significant increase in normalized throughput at the cost of additional storage requirements. Shorter puncturing period RCPT type-II hybrid ARQ schemes have

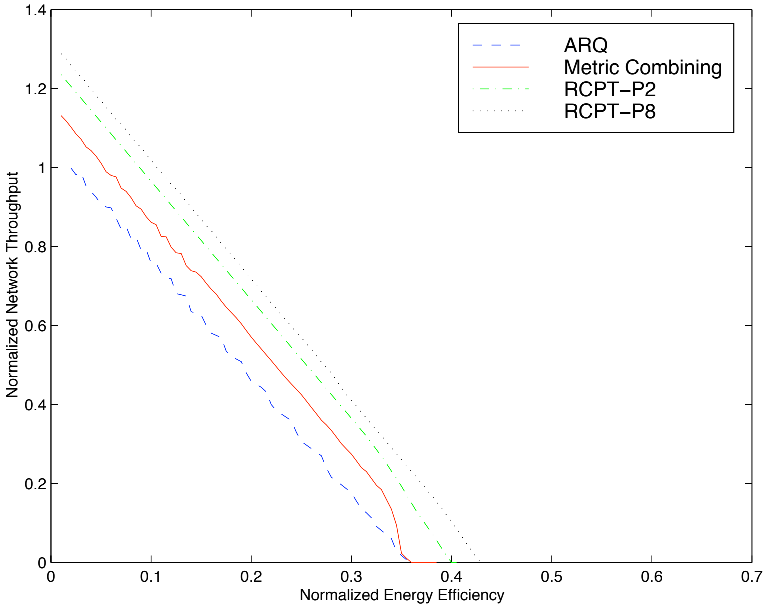


Figure 4.8: Maximum network throughput of turbo coded ARQ schemes under energy efficiency constraints for an asynchronous DS-CDMA network over fading channels

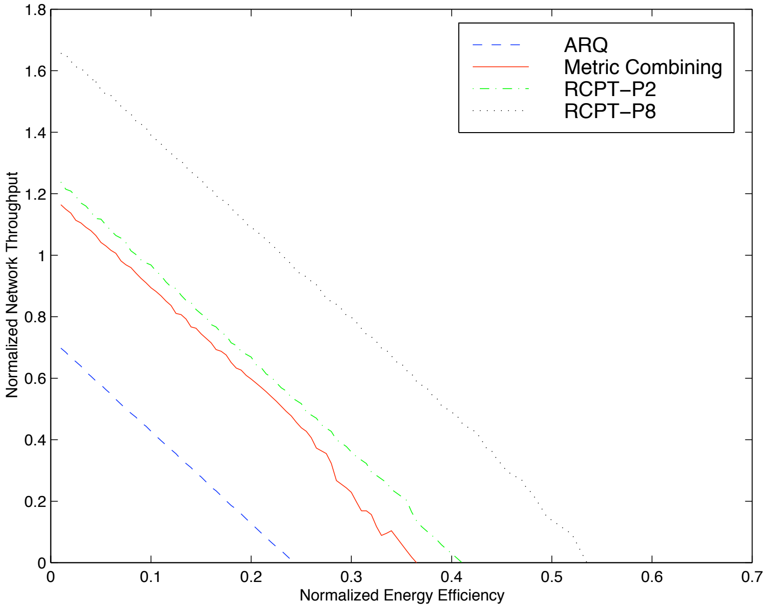


Figure 4.9: Maximum network throughput of turbo coded ARQ schemes for an asynchronous DS-CDMA network under energy efficiency constraints over shadowing channels

similar throughput and energy efficiency performance as longer period RCPT type-II hybrid ARQ protocols but with much smaller average delay. RCPT type-II hybrid ARQ schemes are also shown to be more energy efficient than the type-I hybrid ARQ and metric combining type-I hybrid ARQ schemes. It is observed that the energy efficiency requirement has significant impact on achievable network throughput.

Based on the results shown in this chapter, a system designer can make a series of decisions on the call admission policy. Given the delay and energy efficiency constraints, the maximum offered load can be determined from Figures 4.3 and 4.4 or Figures 4.6 and 4.7 for a certain signal power. Then the maximum throughput within the range of offered load can be determined from Figures 4.2 and 4.5.

CHAPTER V

Conclusions and Future Work

This thesis has concentrated on rate adaptive FEC codes, modified ARQ protocols, and their applications in wireless data networks. In Chapter II, error recursion algorithms are developed for the performance analysis of Reed-Solomon code based fixed-rate type-I hybrid ARQ schemes, a genie-aided rate-adaptive type-I hybrid ARQ scheme, and IRR type-II hybrid ARQ schemes over correlated fading channels. In type-II hybrid ARQ schemes, rate adaptation is triggered by the receipt of a negative acknowledgment at the transmitter, which in response sends incremental redundancy to form a stronger code at the receiver. In the genie-aided rate-adaptive type-I hybrid ARQ scheme, the transmitter determines the code rate for each transmission based on the perfect channel side information provided by a genie. The performance of this genie-aided scheme provides an upper bound on the performance of practical type-I hybrid ARQ schemes with imperfect channel estimations.

Our numerical analysis shows that IRR type-II hybrid ARQ schemes achieve notably higher throughput and energy efficiency than the optimal rate-adaptive type-I hybrid ARQ scheme, while the optimal rate-adaptive type-I hybrid ARQ scheme has smaller delay performance and requires less buffering in the system. High channel correlation is shown to degrade the system performance at moderate SNR for both type-I and type-II ARQ schemes. Soft-decision decoding of Reed-Solomon codes offers one to two dB gain for the the ARQ systems at moderate SNR, but degrades system performance at high SNR due to the additional parity bit used to generate the erasure information for each Reed-Solomon code symbol. The recursive methodology proposed in Chapter II can also be utilized to analyze other rate-adaptive block codes over general finite-state channels.

There are several interesting future research topics related to the work in this chapter. The delay performance is measured by the average number of retransmissions of each frame in our study. This metric is optimistic in the sense that we ignored the possible delay introduced on all subsequent frames by the frame in error. We also assumed an error-free feedback channel and the perfect decoding of header information. In type-I ARQ protocols, when header information cannot be correctly decoded or an acknowledgment frame is lost, the result is simply a throughput reduction by the percentage of lost frames. For type-II ARQ schemes, additional subsequence numbers are required for each redundancy block to ensure robustness in the presence of lost data frames and lost acknowledgment frames. The performance sensitivity to such errors is not well understood because of the varying frame sizes and the effect of lost frames on subsequent decoding procedures.

In practice, applications have maximum delay requirements and systems have limited buffer sizes, which require that the ARQ protocols declare network outages after a number of unsuccessful transmissions. The analysis in this thesis can be directly applied to these cases by modifying the maximum number of retransmissions evaluated, and adding the outage probability to the performance metrics. To enhance the performance of the ARQ protocols, packet combining can also be employed to aid decoding after each protocol restart.

In Chapter III, we proposed an effective method of constructing multiple product codes of various rates. The decoding complexity increases linearly with the code length and dimensions. Weight enumerators and union bounds are derived. Union bounds are shown to be tight at high SNR for both AWGN and Rayleigh fading channels. Simulation results show that the new product codes outperform S-random interleaved turbo codes of the same rate and comparable complexity for a block length of 676 bits when the target BER is greater than 10^{-4}. The well known claim that for a given BER, there is a threshold rate, *above* which product codes are better and *below* which turbo codes are better [40], is found to be not always true.

A family of rate compatible product codes are used to implement an IRR type-II hybrid ARQ scheme. This ARQ scheme achieves throughput above cut-off-rate bounds and within 2 dB of the channel capacity bound for SNR less than 5.5 dB over AWGN channels. Puncturing is used to obtain higher rate product codes but results in poorer performance compared with turbo codes of the same rate.

The results indicate that future investigations of punctured product codes are of great interest. In Chapter III, we made an argument that optimization of puncturing pattern is unnecessary. The observation is that the IOWE of punctured Hamming codes of the same rate are identical due to the symmetry of bit positions in Hamming codes. When a *uniform* random interleaver is used, the argument is valid because the IOWE punctured product code is just a function of the component IOWE. When a deterministic interleaver is used, however, the puncturing patterns of component codes do have effects on the IOWE of the punctured product code. Hence, the optimization of the puncturing pattern for a specific interleaver might yield better performance. On the other hand, as shown by the example in Chapter III, punctured Hamming codes have poor weight distribution regardless of puncturing patterns. The design of component codes with good high rate subcodes will be another key to the construction of product codes that are robust under puncturing.

Since the multiple product codes have superior performance than S-random interleaved punctured turbo codes, it would be interesting to investigate the decoding dynamics of product codes. In [70], a geometric interpretation of the turbo-decoding algorithm was described. A density evolution approach was proposed for the investigation of algorithm convergence and error performance. A similar approach might be applicable to the study of product codes.

In Chapter IV, we considered a few rate adaptive ARQ protocols in a more realistic wireless network setting and examined the performance with multiple users. The ARQ protocols that were investigated include $\frac{1}{3}$ turbo-coded type-I hybrid ARQ, metric combining type-I hybrid ARQ, and RCPT type-II hybrid ARQ schemes. These protocols are applied to the uplink traffic of an asynchronous DS-CDMA data network over multipath fading and shadowing channels. Compared with rate $\frac{1}{3}$ turbo-coded type-I hybrid ARQ and metric combining ARQ schemes, the RCPT type-II hybrid ARQ schemes provide a significant increase in normalized network throughput at the cost of additional storage requirements. Shorter puncturing period RCPT type-II hybrid ARQ schemes have similar throughput and energy efficiency performance as longer period RCPT type-II hybrid ARQ protocols but with much smaller average delay. RCPT type-II hybrid ARQ schemes are also shown to be more energy efficient than both type-I hybrid ARQ and metric combining type-I hybrid ARQ schemes. It is observed that the energy efficiency requirement has significant impact on achievable network throughput. Based on the results shown in this chapter, a system designer can make a series of decisions on the call admission policy.

The superior performance of type-II hybrid ARQ protocols for the uplink channels indicates their promising application to the downlink traffic also. With more web browsing applications on mobile devices, the downlink is becoming the bottleneck of wireless networks. In new wireless standards like CDMA-HDR and 3GPP, the uplink capacity is sacrificed to improve the downlink capacity. In this thesis, the multi-user interference model of the uplink channel is derived to facilitate the simulation of hybrid ARQ protocols. Similar interference models can also be derived for the downlink channel for performance evaluation.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] O. F. Acikel and W. E. Ryan, “Punctured turbo-codes for BPSK/QPSK channels,” *IEEE Transactions on Communications*, vol. 47, no. 9, pp. 1315–1323, September 1999.

[2] S. Alamouti and S. Kallel, “Adaptive trellis-coded multiple-phase-shift-keying for Rayleigh fading channels,” *IEEE Transactions on Communications*, vol. 42, no. 6, pp. 2305–2314, June 1994.

[3] M.-S. Alouini and A. Goldsmith, “Adaptive M-QAM modulation over Nakagami fading channels,” in *Proceedings of the IEEE Global Telecommunications Mini-Conference*, volume 4, pp. 218–233, 1997.

[4] M.-S. Alouini, X. Tang, and A. J. Goldsmith, “Adaptive modulation scheme for simultaneous voice and data transmission over fading channel,” *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 5, pp. 837–850, May 1999.

[5] A. Annamalai and V. K. Bhargava, “Throughput performance of slotted DS/CDMA ALOHA with packet combining over generalized fading channels,” *Electronics Letters*, vol. 33, no. 14, pp. 1195–1197, July 1997.

[6] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Transactions on Information Theory*, vol. IT-30, pp. 284–287, March 1974.

[7] A. Barbulescu and S. Pietrobon, “Rate compatible turbo codes,” *Electronics Letters*, vol. 31, no. 7, pp. 535–536, March 1995.

[8] C. W. Baum and M. B. Pursley, “Bayesian methods for erasure insertion in frequency-hop communications systems with partial-band interference,” *IEEE Transactions on Communications*, vol. 40, no. 7, pp. 1231–1238, July 1992.

[9] G. Benedetto, Sergio Montorsi, “Unveiling turbo codes: some results on parallel concatenated coding schemes,” *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 409–428, March 1996.

[10] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, “Soft-output decoding algorithms in iterative decoding of turbo codes,” Technical Report 42-124, JPL, TDA Progress Report, Feb 1996.

[11] S. Benedetto, D. Divsalar, G. Montorsi, and F. Pollara, "Serial concatenation of interleaved codes: performance analysis, design, and iterative decoding," *IEEE Transactions on Information Theory*, vol. 44, no. 3, pp. 909–926, may 1998.

[12] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proceedings of the 1993 IEEE International Conference on Communications*, pp. 1064–1070, May 1993.

[13] D. Bertsekas and R. G. Gallager, *Data Networks*, Prentice Hall, Upper Saddle River, NJ 07458, 2 edition, 1992.

[14] A. M. Y. Bigloo and T. A. Gulliver, "A slotted frequency-hopped multiple-access network with packet combining," *IEEE Transactions on Communications*, vol. 14, no. 9, pp. 1859–1865, December 1991.

[15] A. M. Y. Bigloo and T. A. Gulliver, "Slotted DS/SSMA ALOHA with packet combining in a Rayleigh fading channel," in *Proceedings of the IEEE Vehicular Technology Conference*, volume 3, pp. 1710–1714, 1996.

[16] A. M. Y. Bigloo, T. A. Gulliver, and V. K. Bhargava, "Maximum-likelihood decoding and code combining for DS/SSMA slotted ALOHA," *IEEE Transactions on Communications*, vol. 45, no. 12, pp. 1602–1612, December 1997.

[17] A. M. Bigloo, T. A. Gulliver, Q. Wang, and V. K. Bhargava, "Robust rate-adaptive hybrid ARQ scheme for frequency-hopped spread-spectrum multiple-access communication systems," *IEEE Transactions on Communications*, vol. 12, no. 5, pp. 917–924, June 1994.

[18] H. Bischl and E. Lutz, "Packet error rate in the non-interleaved Rayleigh channel," *IEEE Transactions on Communications*, vol. 43, no. 2/3/4, pp. 1375–1382, February/March/April 1995.

[19] R. E. Blahut, *Digital transmission of information*, Addison Wesley, 1990.

[20] I. E. Bocharova and B. D. Kudryashov, "Rational rate punctured convolutional codes for soft-decision Viterbi decoding," *IEEE Transactions on Information Theory*, vol. 43, no. 4, pp. 1305–1313, July 1997.

[21] J. B. Cain, G. C. Clark, Jr., and J. M. Geist, "Puncture convolutional codes of rate (n-1)/n and simplified maximum likelihood decoding," *IEEE Transactions on Information Theory*, vol. IT-25, no. 1, pp. 97–100, January 1979.

[22] D. Chase, "A class of algorithms of decoding block codes with channel measurement information," *IEEE Transactions on Information Theory*, vol. IT-18, pp. 170–182, January 1972.

[23] D. Chase, "Code combining – a maximum-likelihood decoding approach for combining an arbitrary number of noisy packets," *IEEE Transactions on Communications*, vol. COM-33, no. 5, pp. 385–393, May 1985.

[24] M. J. Chu and W. E. Stark, "Effect of mobile velocity on communications in fading channels," *IEEE Transactions on Communications*, vol. 47, , 1999. to appear.

[25] A.-G. A. Daraiseh and C. W. Baum, "Packet combining in frequency-hop spread-spectrum communication systems," *IEEE Transactions on Communications*, vol. 46, no. 1, pp. 23–33, January 1998.

[26] S. Dave, J. Kim, and S. Kwatra, "An efficient decoding algorithm for block turbo codes," *IEEE Transactions on Communications*, vol. 49, no. 1, pp. 41–46, January 2001.

[27] P. Decker, "Adaptive type-II hybrid ARQ/FEC protocol suitable for GSM," in *Proceedings of the IEEE Vehicular Technology Conference*, volume 1, pp. 330–333, 1994.

[28] D. Divsalar and R. McEliece, "Effective free distance of turbo codes," *Electronics Letters*, vol. 32, no. 5, pp. 445–446, February 1996.

[29] D. Divsalar and F. Pollara, "Turbo codes for PCS applications," in *Proceedings of the 1995 IEEE International Conference on Communications*, volume 1, pp. 54–59, Seattle, WA, June 1995.

[30] T. M. Duman and M. Salehi, "New performance bounds for turbo codes," *IEEE Transactions on Communications*, vol. 46, pp. 717–723, June 1998.

[31] M. Eroz and T. E. Fuja, "Hybrid-ARQ system using rate-compatible trellis codes designed for Rayleigh fading," *European Transactions on Telecommunications*, vol. 9, no. 6, pp. 483–495, 1998.

[32] S. Falahati, T. Ottosson, A. Svensson, and L. Zihuai, "Convolutional coding and decoding in hybrid type-II ARQ schemes on wireless channels," in *Proceedings of the IEEE Vehicular Technology Conference*, volume 3, pp. 2219–2224, 1999.

[33] R. G. Gallager, *Information Theory and Reliable Communications*, Wiley, New York, 1968.

[34] R. G. Gallager, *Low-Density Parity-Check Codes*, M.I.T. press, 1963.

[35] D. L. Goeckel, "Adaptive coding for time-varying channels using outdated fading estimates," *IEEE Transactions on Communications*, vol. 47, no. 6, pp. 844–855, June 1999.

[36] A. Goldsmith, "Capacity and dynamic resource allocation in broadcast fading channels," in *Proceedings of Allerton Conference on Communications, Control and Computing*, pp. 915–924, 1995.

[37] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronics Letters*, vol. 27, no. 23, pp. 2145–2146, November 1991.

[38] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," in *Proceedings of GLOBECOM'89*, pp. 47.1.1–47.1.7, Dallas, Texas, 89.

[39] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Transactions on Communications*, vol. 36, no. 4, pp. 389–400, April 1988.

[40] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 429–445, March 1996.

[41] B. A. Harvey and S. B. Wicker, "Packet combining systems based on the Viterbi decoder," *IEEE Transactions on Communications*, vol. 42, no. 2-4, pp. 1544–1557, Feb-Apr 1994.

[42] J. Hayes, "Adaptive feedback communications," *IEEE Transactions on Communications Technology*, vol. COM-16, pp. 1331–1337, sep 1968.

[43] J. Hokfelt, *On the Design of Turbo Codes*, PhD thesis, Lund University, Sweden, August 2000.

[44] T. Ji and W. E. Stark, "Concatenated punctured turbo Reed-Solomon codes in a hybrid FEC/ARQ DS/SSMA data network," in *Proceedings of the 1999 IEEE Vehicular Technology Conference*, pp. 1678–1682, Houston, Texas, May 1999.

[45] P. Jung and J. Plechinger, "Performance of rate compatible punctured turbo-codes for mobile radio applications," *Electronics Letters*, vol. 33, no. 25, pp. 2102–2103, December 1997.

[46] D. E. Kammer, "SINGARS — the new generation combat net radio system," in *Proceedings of Tactical Communications Conference*, pp. 64–72, April 1986.

[47] T. Kaneko, T. Nishijima, and S. Hirasawa, "Improvement of soft-decision maximum-likelihood decoding algorithm using hard-decision bounded-distance decoding," *IEEE Transactions on Information Theory*, vol. 43, no. 4, pp. 1314–1319, July 97.

[48] T. Kaneko, T. Nishijima, H. Inazumi, and S. Hirasawa, "Efficient maximum-likelihood-decoding algorithm for linear block codes with algebraic decoder," *IEEE Transactions on Information Theory*, vol. 40, no. 2, pp. 321–327, March 1994.

[49] A. B. Kiely, S. J. Dolinar, R. J. McEliece, L. L. Ekroot, and W. Lin, "Trellis decoding complexity of linear block codes," *IEEE Transactions on Information Theory*, vol. 42, no. 6 pt 1, pp. 1687–1697, November 1996.

[50] J. Lai and N. B. Mandayam, "Performance of reed-solomon codes for hybrid-ARQ over Rayleigh fading channels under imperfect interleaving," *IEEE Transactions on Communications*, vol. 48, no. 10, pp. 1650–1659, October 2000.

[51] L. Lee, "New rate-compatible punctured convolutional codes for Viterbi decoding," *IEEE Transactions on Communications*, vol. 42, no. 12, pp. 3073–3079, December 1994.

[52] S. Lin and P. S. Yu, "Hybrid ARQ scheme with parity retransmission for error control of satellite channels," *IEEE Transactions on Communications*, vol. COM-30, no. 7, pp. 1701–1719, July 1982.

[53] H. Lou and A. Cheung, "Performance of punctured channel codes with ARQ for multimedia transmission in Rayleigh fading channels," in *Proceedings of the IEEE Vehicular Technology Conference*, volume 1, pp. 282–286, 1996.

[54] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 32, pp. 1645–1646, August 1996.

[55] D. M. Mandelbaum, "An adaptive-feedback coding scheme using incremental redundancy," *IEEE Transactions on Informaion Theory*, vol. 20, pp. 388–389, May 1974.

[56] S. Nanda, K. Balachandran, and S. Kumar, "Adaptation techniques in wireless packet data services," *IEEE Communications Magzine*, vol. 38, no. 1, pp. 54–64, January 2000.

[57] H. Nickl, J. Hagenauer, and F. Burkert, "Approaching Shannon's capacity limit by 0.27 db using simple Hamming codes," *Communications Letters*, vol. 9, no. 5, pp. 130–132, September 1997.

[58] T. Niinomi, I. Sasase, and S. Mori, "Selective repeat type-II hybrid ARQ/FEC scheme using rate-compatible punctured convolutional code," in *IEEE International Conference on Communications*, volume 3, pp. 1251–1255, 90.

[59] J. P. Odenwalder, *Optimum Decoding of convolutional Coldes*, PhD thesis, University of California, Los Angeles, 1970.

[60] J. P. Odenwalder, "Hybrid coding systems study final report," NASA Document NASA-CR-114486, NASA, San Diego, CA, USA, 1972.

[61] C. Pimentel and I. F. Blake, "Non-interleaved Reed-Solomon coding performance on finite state channels," in *IEEE International Conference on Communication*, volume 3, pp. 1493–1497, 1997.

[62] L. Ping, S. Chan, and K. L. Yeung, "Efficient soft-in-soft-out sub-optimal decoding rule for single parity check codes," *Electronics Letters*, vol. 33, no. 19, pp. 1614–1616, September 1997.

[63] J. G. Proakis, *Digital communications*, McGraw-Hill, 1995.

[64] M. B. Pursley and S. D. Sandberg, "Incremental-redundancy transmission for meteor-burst communications.," *IEEE Transactions on Communications*, vol. 39, no. 5, pp. 689–702, May 1991.

[65] M. B. Pursley and S. D. Sandberg, "Variable-rate hybrid ARQ for meteor-burst communications," *IEEE Transactions on Communications*, vol. 40, no. 1, pp. 60–73, January 1992.

[66] M. B. Pursley and C. S. Wilkins, "Adaptive-rate coding for frequency-hop communications over Rayleigh fading channels," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 7, pp. 1224–1232, 1999.

[67] R. M. Pyndiah, "Near-optimum decoding of product codes: block turbo codes," *IEEE Transactions on Communications*, vol. 46, no. 8, pp. 1003–1010, August 1998.

[68] D. Rankin and T. A. Gulliver, "Randomly interleaved single parity check product codes," in *Proceedings of the 1999 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, volume 1, pp. 420–423, Victoria, BC, 1999.

[69] M. Rice and S. B. Wicker, "Adaptive error control for slowly varying channels," *IEEE Transactions on Communications*, vol. 42, no. 4, pp. 917–926, April 1994.

[70] T. Richardson, "The geometry of turbo-decoding dynamics," *IEEE Transactions on Information Theory*, vol. 46, no. 1, pp. 9–23, January 2000.

[71] D. N. Rowitch and L. B. Milstein, "Rate compatible punctured turbo (RCPT) codes in a hybrid FEC/ARQ system," in *Proceedings of the 1997 IEEE Global Telecommunications Mini- Conference*, volume 4, pp. 55–59, Phoenix, AZ, November 1997, IEEE.

[72] D. N. Rowitch and L. B. Milstein, "On the performance of hybrid FEC/ARQ systems using rate compatible punctured turbo (RCPT) codes," *IEEE Transactions on Communications*, vol. 48, no. 6, pp. 948–959, June 2000.

[73] D. N. Rowitch, *Convolutional and turbo coded multicarrier direct sequence CDMA, and applications of turbo codes to hybrid ARQ communication systems*, PhD thesis, University of California, San Diego, 1998.

[74] I. Sason and S. Shamai(Shitz), "Improved upper bounds on the ML decoding error probability of parallel and serial concatenated turbo codes via their ensemble distance spectrum," *IEEE Transactions on Information Theory*, vol. 46, no. 1, pp. 24–47, January 2000.

[75] K. Sayood, *Introduction to data compression*, Morgan Kaufmann Publishers, San Francisco, CA, USA, 1996.

[76] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, pp. 379–423 (Part I) and 623–656 (Part II), 1948.

[77] C. E. Shannon, “Probability of error for optimal codes in a Gaussian channel,” *Bell System Technical Journal*, vol. 38, no. 3, pp. 611–656, May 1959.

[78] W. Stark, “Coursepack for digital communications theory,” University of Michigan, Ann Arbor.

[79] A. S. Tanenbaum, *Computer Networks*, Prentice Hall PTR, Upper Saddle River, N.J., USA, 3rd edition, 1996.

[80] S. Verdú and S. Shamai (Shitz), “Spectral efficiency of cdma with random spreading,” *IEEE Transactions on Information Theory*, vol. 45, no. 2, pp. 622–640, March 1999.

[81] A. J. Viterbi, “A robust ratio-threshold technique to mitigate tone and partial band jamming in coded MFSK systems,” in *IEEE Military Communications Conference Record*, pp. 22.4.1–22.4.5, October 1982.

[82] A. J. Viterbi, *CDMA Principles of Spread Spectrum Communication*, Addison-Wesley, 1995.

[83] *VSC9210: FEC encoder and decoder chipset*, Vitesse Semiconductor Corporation.

[84] N. Wiberg, *Codes and decoding on general graphs*, PhD thesis, Dept. Elect. Eng. Linkoping University Sweden, 1996.

[85] S. B. Wicker, *Reed Solomon Codes and Their Applications*, New York: IEEE Press, 1994.

[86] S. B. Wicker and M. J. Bartz, “Type-II hybrid ARQ protocols using punctured MDS codes,” *IEEE Transactions on Communications*, vol. 42, no. 2/3/4, pp. 1431–1440, February/March/April 1994.

[87] K. A. Witzke and C. Leung, “Comparison of some error detecting CRC code standards,” *IEEE Transactions on Communications Technology*, vol. COM-33, no. 9, pp. 996–998, September 1985.

[88] J. K. Wolf, “Efficient maximum likelihood decoding of linear block codes using a trellis,” *IEEE Transactions on Information Theory*, vol. IT-24, no. 1, pp. 76–80, January 1978.

[89] J. K. Wolf and R. D. Blakeney, “Exact evaluation of the probability of undetected error for certain shortened binary CRC codes,” in *Proceedings - IEEE Military Communications Conference 1988*, pp. 287–292, 1988.

[90] K. Wu, S. Lin, and M. Miller, "Hybrid ARQ scheme using multiple shortened cyclic codes," in *Proceedings of GLOBECOM '82*, p. 7120716, Miami, FL, USA, 1982.