WIRELESS IMAGE TRANSMISSION OVER FADING CHANNELS USING MULTIPLE-DESCRIPTION CONCATENATED CODES

BY

DANIEL GROBE SACHS

B.S., University of Illinois at Urbana-Champaign, 1998

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2000

Urbana, Illinois

ABSTRACT

Recent advances in scalable image compression have made it possible to decode images in a progressive fashion. However, scalable coders like the emerging JPEG-2000 standard are more susceptible to the uncorrectable bit errors often caused by channel fading. Many solutions to this problem have been proposed, but most are sensitive to the location of the errors or require large interleavers to spread errors out. In this work, we combine an efficient multiple-description structure based on forward error correction with an outer channel encoding to create a framework to efficiently send image descriptions over a wireless channel. The framework is a concatenated channel code including a row (outer) code based on convolutional error-correcting codes and CRC (cyclic redundancy check) codes for error detection, and a source-channel column (inner) encoding consisting of the bit-scalable image and an optimized array of unequal protection Reed-Solomon erasure-correction codes. By systematically matching the unequal protection codes to the embedded source bitstream, we can enable graceful degradation and minimize expected distortion at the receiver.

This approach to image transmission over fading channels offers significant improvements in both peak and expected image quality when compared to current state-of-the-art techniques. Our packetization scheme is also naturally suited for hybrid packet-network/wireless channels, such as those used for wireless Internet access.

For Sherlock

ACKNOWLEDGMENTS

I would like to acknowledge my advisors, Kannan Ramchandran and Douglas L. Jones, for their insight and guidance directing me in my graduate research. I would also like to thank Rohit Puri for allowing me the use of the multiple description optimizer described Section 3.2, and Anand Raghavan for his contributions to the simulations and for his input on the conference publication which has been incorporated into this thesis.

I would also like to thank my parents, Kathleen L. Bode, Brendan Dunn, and Ravikumar Venkateswar for helping me maintain my sanity for the past two years.

My graduate research was sponsored in part by NASA under grant NASA NAG 3-2263.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Motivation for Joint Source-Channel Coding

In his seminal work on information theory [1], Claude Shannon proved the separation theorem: that in the limiting case of infinite time and complexity, optimal performance can be achieved using separate source and channel codes. While this statement might lead one to conclude that good performance can be achieved using separated source and channel coding in more realistic regimes, the seperation theorem does not actually apply when delay or complexity is limited. This means that in these limited regimes, it is not necessarily true that a separated structure can achieve optimal performance. In this thesis, I study an application of joint source-channel coding techniques to improve the expected distortion of images transmitted over wireless channels. Past work [2], [3] has shown that performance can be improved significantly through the application of these techniques. However, these papers do not consider the more hostile fading channel model; another paper which does consider channel fading [4] does not consider explicit joint source-channel optimization.

The main contribution of this work is a source-channel coding framework which extends existing techniques for transmitting images over wireless fading channels with an explicit joint optimization of source and channel codes. This framework uses an outer code to transform the fading channel into a packet-loss channel, and an inner multiple-description code to achieve

graceful degradation during fades. We will show that this framework can improve performance significantly over state-of-the-art techniques for image transmission over fading channels, improving by over 0.6 dB the expected mean-squared error at the receiver achieved by the methods described in [4].

This thesis is largely derived from a paper presented at the SPIE Electronic Imaging 2000 conference [5].

1.2 Thesis Outline

This thesis is divided into five chapters. Chapter 1 motivates the joint source-and-channel encoding approach to the transmitting images over hostile channels, outlines the overall structure of the thesis. This chapter also outlines various systems for transmitting images over hostile wireless channels, starting with techniques in the literature to send images over memoryless channels and proceeding to various channel-coding and source-channel coding methods for transmitting images over the more complex flat-fading channel with memory.

In Chapter 2, we carefully consider the difficult case of transmitting images over channels with fading coefficients that change slowly over time ("slow-fading channels"). We describe a framework to efficiently send images over these channels using a product code with joint source-channel code optimization. The code framework consists of an inner erasure-correction code with unequal protection, meaning that the protection level for the inner code is varied according to the importance (in the rate-distortion sense) of the bits being encoded. The outer code consists of an error-detection code wrapped with an error-correction code, arranged so that a failure of the error-correction code to recover the transmitted packets is detected by the error-detection code. In Chapter 3, the problem of bit allocation for this source-channel framework is addressed. In this chapter, we develop a fast optimization algorithm for the inner

unequal-protection code. We also examine the optimization of the outer error-correction code rate.

Chapter 4 reports the results of simulating the proposed scheme on test images, using several combinations of outer code rate, packet size, and total rate. We find that the performance of the proposed scheme is very good, improving significantly on state-of-the-art techniques for transmitting images over wireless channels described in existing literature.

Finally, Chapter 5 draws conclusions about the performance of the proposed error-control framework, and examines possible directions for future research.

1.3 Systems for Image Transmission over Wireless Channels

Recent advances in image compression have enabled the creation of efficient, embedded source coders. These source coders generate PSNR-scalable bitstreams, in which each additional bit is used to refine an image and improve reconstruction quality. However, these scalable image encodings also tend to be fragile — that is, the encoded bitstreams are highly susceptible to bit errors. Although these encoders achieve better fidelity with fewer bits than ever before, even single-bit errors can render the entire transmitted image useless. Since wireless channels suffer from significant bit-error rates (often higher than one error per hundred bits), some mechanism to protect the encoded image is required. Without such a mechanism, the channel bit errors will prevent accurate decoding of the image.

1.3.1 Wireless channel models

Several wireless channel models have been widely used in the literature. The most simplistic of these is the binary symmetric channel (BSC) model, which is defined by a single parameter ϵ representing the probability that a given bit is flipped from a one to a zero or vice versa.

Binary symmetric channels are memoryless, and each bit is flipped independently and with equal probability. The simplicity of this channel model makes it easy to use; conventional channel codes, such as convolutional codes using the Viterbi algorithm for decoding, perform well on this type of channel. Typical bit-error rates for a BSC found in the literature are 10^{-1} to 10^{-3}, meaning that from 1 bit in 10 to 1 bit in 1 000 are flipped.

A more realistic channel model often discussed in the literature is the additive white Gaussian noise (AWGN) channel. This type of channel model is similar to the BSC model in that each bit is flipped with equal probability; the difference is that the AWGN model admits soft decoding — that is, in addition to information on *what* the bits decoded to, we get information about *how likely* the bit is what we say it is. We assume a BPSK (binary phase shift keying) transmission system, which transmits 1 bit per symbol transmitted. In a BPSK transmission system on an AWGN channel, each bit starts out as either $+A$ or $-A$ (modulated onto a sinusoidal carrier), where $A = \sqrt{E}$ is the amplitude of the transmitted signal, and E is the signal's energy. Typically, $+A$ represents a one, and $-A$ represents a zero. To this signal is added normally distributed noise with a variance that characterizes the signal-to-noise ratio, or SNR, of the channel. Note that because of the shape of a Gaussian distribution, if we make a decoding mistake, the received value after matched filtering is likely to be close to zero. This information allows more accurate decoding of channel codes because we can take the probability that a bit is correct into account when we attempt to determine if it is in error.

While the AWGN channel is a good model for a stationary transmitter and receiver where the channel noise is dominated by thermal noise at the receiver, this model is inadequate for mobile wireless networks. In addition to the Gaussian noise modeled by the AWGN channel, the signal also encounters *fading* on the path between the transmitter and the receiver. This

fade varies the amount of signal energy received due to multipath interference (where different reflections of the signal interfere with each other) and shadowing. Both of these factors can vary wildly as the position of the antennas vary.

There are many possible models for this fading. One popular model is a Markov chain, where the wireless link is modeled as one of a family of AWGN channels with different signal-to-noise ratios. The different channels in the family are each associated with a state in a Markov chain; transition probabilities are defined between each pair of states. Another popular model is Rayleigh flat fading. In this model, the fade is a continuous function that varies constantly as the transmitter and receiver move relative to each other. Rayleigh fading is so named because the distribution of the received signal level has a Rayleigh distribution.

More advanced fading models include frequency-selective fading, which incorporates the frequency-dependent nature of fading. (The frequency affects the phase offsets of the multipath components of the signal, which in turn affects the interference patterns that cause fading.) Although these techniques are necessary for analysis of wideband technologies like direct-sequence code-division multiple access (DS-CDMA) and orthogonal frequency division multiplexing (OFDM), they are not necessary for the narrowband BPSK implementations that are studied herein.

Therefore, this thesis concentrates on narrowband Rayleigh fading based on the Clarke flat fading model. We simulate this model using the Jakes fading simulator described in [6]. We consider the difficult case of slow fading, for which conventional convolutional channel codes perform poorly.

1.3.2 Existing techniques for wireless image transmission

Many error-control solutions have been proposed to address the problem of transmitting images over wireless channels. These techniques include approaches that add redundancy at the source level, such as error-resilient source encodings and source-based multiple description, as well as techniques that use fragile source encodings but attempt to prevent bit errors from being seen by the decoder. The technique for sending images over wireless channels discussed in this paper is of the second type, but we view the inner channel codes as part of a multiple-description image encoding rather than as part of the code that protects the image against channel-induced bit errors.

Many techniques have been proposed for the transmission of images over wireless networks. One obvious technique is to take an existing image encoder, and then protect the bitstream using conventional codes — a separated source-channel encoding scheme. Many schemes in the literature that use an existing image coder are based on the popular SPIHT (Set Partitioning Into Hierarchical Trees) image coder [7]. Although the SPIHT coder has very poor error resilience (single-bit errors often cause a loss of synchronization between the encoder and decoder, and can result in a badly distorted image), it is a *progressive* encoding in which the decoding can be terminated at any bit in the stream. Therefore, if channel-induced bit errors can be detected, decoding can be terminated before the bit containing the error is decoded. If this is not done, the error is likely to cause later bits to be decoded incorrectly as well, which significantly increases decoding error.

One application of separate source encoding and channel protection was proposed by P. Sherwood and K. Zeger [8]. In this system, the SPIHT-encoded image is divided into a series of packets. Each packet is then protected with an error-correcting code and an error-detection

code which allows failures of the error-correcting code to be detected. In this system, "equal protection" rate-compatible punctured-convolutional (RCPC) codes are used for error correction, and cyclic redundancy check (CRC) codes are used for error detection. (In this context, "equal protection" refers to the fact that each packet is protected equally against channel errors.) The progressive nature of the SPIHT image encoding allows the data bits to be decoded up to the first packet in which an error was detected, allowing an acceptable image to be decoded unless an early packet in the sequence is lost.

The problem with this scheme is that the loss of the first image packet is catastrophic because it prevents the decoding of any subsequent image packets even if they are received intact. To address this problem, joint source-channel coding schemes have been proposed which account for the varying importance of the bits in the progressive image description. One example of such a scheme is proposed by Appadwedula et al. in [9]. In this scheme, symbol energy and Reed-Solomon code rates are allocated using a gradient-descent search to minimize expected distortion.

A similar scheme, proposed by V. Chande and N. Farvadin in [3], combines the Sherwood and Zeger packetized encoding (with RCPC and CRC codes) and joint source-channel code optimizations like those in [9]. In Chande's scheme, the code schedule (which defines the RCPC code rate for each packet transmitted) is chosen for each packet from a family of channel codes by a dynamic-programming optimizer that maximizes an objective measure of performance. In [3] the objective measures considered were the expected mean-squared error at the receiver and the expected number of bits between the beginning of the transmitted image and the first detected error. This optimization lowers the probability that the early bits in the encoded data stream will be lost, at the expense of raising the probability that later source bits will be

lost. Because the early, important bits are given more protection than the later, less important bits of the image description, the optimization of unequal-protection codes using these criteria significantly decreased the expected distortion at the receiver.

This type of scheme works well for binary symmetric or Gaussian noise channels because the RCPC code rates can be easily chosen so that the probability of packet loss is small; optimization can be achieved by balancing the received image quality against the probability that part or all of the image is lost. However, such a scheme is less than ideal for a fading channel, because on such a channel some of the packets will inevitably be lost due to deep fades. Although the channel codes can be chosen so that this is unlikely, this is wasteful of bits when the channel signal-to-noise ratio is high. Unfortunately, it is a priori unknown whether a given packet will encounter severe fading. Even Chande and Farvadin's scheme assumes that the channel is memoryless or nearly so — although low-rate RCPC codes are used to protect the most important data, the extremely low coding rate required to ensure these first packets are received intact even in deep fades is wasteful of bits and hurts performance significantly.

There are several ways to address this weakness. First, an interleaver can be used to decorrelate errors in nearby bits by spreading adjacent bits out over time; this is examined in [9]. Reasonably-sized interleavers are effective on fast-fading channels. They essentially eliminate correlation between the fading coefficient of nearby bits, reducing the channel to a memoryless Rayleigh channel. Because the bit errors caused by a memoryless Rayleigh channel are scattered randomly, they can be corrected using conventional channel codes, including Reed-Solomon and RCPC codes. However, the delay through a stream interleaver increases as the square of the separation it introduces between any two adjacent bits; this increased delay also

represents increased complexity. Therefore, for slow-fading channels, the delay and complexity of an interleaver sufficient to significantly decorrelate the channel are prohibitive [4].

Instead of using an interleaver, we can accept the fact that packets subject to heavy fade are likely to be lost, and design the source code to work around this. In [4], Sherwood and Zeger introduce a scheme with concatenated RCPC+CRC and Reed-Solomon erasure correction codes; the Reed-Solomon codes allow the transmitted data to be reconstructed even if several of the packets are lost. This type of scheme allows the strength of the RCPC outer code to be set such that some packets are always lost (increasing the rate of the code and therefore the number of source bits sent); the inner Reed-Solomon code can reconstruct the lost packets using fewer extra parity bits than the RCPC+CRC codes alone would require. Sherwood and Zeger found this scheme to be quite effective, decreasing expected mean-squared distortion at the receiver significantly compared to the use of RCPC+CRC codes only [4].

Performance can be increased even more by putting extra protection on the first few packets. Sherwood and Zeger offer two different "unequal protection" schemes to accomplish this: "UEP1" and the lower-rate "UEP2," which offers better expected performance at the cost of larger reconstruction error if few packets are lost. In these schemes, extra coding is used to further reduce the probability of a decoding failure occurring in the first 400 and 2 000 transmitted bits. By coarsely matching channel codes to the progressive source image, performance is improved — justifying the effort of doing more complete joint source-channel coding.

CHAPTER 2

A PRODUCT-CODE FRAMEWORK FOR TRANSMITTING IMAGES

In [4], Sherwood and Zeger used a combination of error-correction coding and packet-erasure correction to create an error-control scheme for transmitting images over fading channels that improved performance significantly over the previous state of the art. More importantly, however, they also introduced the view that channel error bursts caused by fades are similar in nature to packet losses, and that a fading channel can be transformed into a packet loss channel through the use of channel codes. However, their work does not take full advantage of this observation. The Reed-Solomon code that they use across packets is ad hoc and certainly not the best approach for all circumstances; cases where the total bit rate available is not approximately 0.25 bpp are not considered, nor are cases where the extra protection on the first bits does not actually improve performance. For these reasons, a more systematic approach for allocating protection across the packets is desirable.

To address this weakness of the Sherwood and Zeger design, we consider the use of a multiple-description image coder instead of the ad-hoc Reed-Solomon codes. These coders are designed to create several independent descriptions of the encoded image. Given only one of these image descriptions, the image can be reconstructed with some level of quality;[1] as more of

[1] Note that if many descriptions are generated, more than one description may be required to decode a useful representation of the image.

the descriptions are made available to the receiver, the received image quality improves. Ideally, the image quality at the receiver depends only on *how many* image descriptions are received, and not on *which* descriptions are received.

2.1 Multiple Description Using Forward Error Correction

Several techniques exist in the literature for generating these multiple-description (MD) codes, including both source-domain approaches and approaches that combine a progressive image coder like SPIHT with forward error-correcting codes (FEC). Source-domain MD codes [10], [11] are usually less flexible than MD codes designed using forward error correction and often incur more distortion for a given bit rate and packet-loss level; some, such as the multiple-description version of SPIHT described in [11], are more difficult to decode as well. For this reason, I concentrate on multiple description using forward error correction (MD-FEC) in this thesis.

As in the Sherwood and Zeger encoding, the progressive bitstream used by the MD-FEC encoding is generated by the popular SPIHT image coder [7]. This encoder is a state-of-the-art image coder with the valuable *embedding* property, in that the encoding for any given rate is the prefix of the encoding of the same image for any higher rate. This property implies the *progressive* property, meaning that additional bits provide for an improved reconstruction at the receiver. The embedding property simplifies the creation of the source code, since the image can be encoded (and the rate-distortion curve generated) before the final rate that will be used is known.

An approach for using these Reed-Solomon channel codes to create a multiple-description bitstream was introduced in 1999 in independent work by A. Mohr et al. [13], and R. Puri and K. Ramchandran [14]. If Reed-Solomon codes are allocated across blocks so that each

Reed-Solomon block contains exactly one symbol in each packet, which packets arrive at the receiver is irrelevant.

The MD-FEC algorithm "equalizes" the bitstream by spreading contiguous data across packets as shown in Figure 2.1. The i^{th} Reed-Solomon symbol in each packet forms the i^{th} column. Each column c_i forms an (n, k_i) Reed-Solomon code; k_i source symbols are placed into the column, and the remaining space is taken by the Reed-Solomon parity ("FEC") symbols. The number of data symbols k_i can be chosen for each column. Since we have a progressive source, each source symbol is more important than the one following it. So, for columns c_i, c_j, $i < j$, we would want $k_i \leq k_j$. A column c_i can be decoded if and only if k_i or more packets are received. Since k_i is nondecreasing, if m packets are received we receive all source bits up to the first column encoded with an $(n, m+1)$ code. Therefore, this scheme, combined with a fully embedded image coder like SPIHT, satisfies the intention of offering multiple descriptions of the source image. All of the descriptions — the packets — are equally important, and any arbitrary subset of them can be decoded at the receiver, yielding the same image as any other subset of the same size.

This forward-error-correction approach to multiple description is extremely flexible. The reconstruction quality for any given number of received packets can be defined by simply choosing R_i to be the number of symbols required to achieve the desired image quality at the receiver. In this way, we can create an arbitrary profile that, for the same number of packets decoded at the receiver, achieves the same mean-squared distortion as any other multiple-description algorithm. Furthermore, simulations have shown that MD-FEC encoded SPIHT requires fewer bits per packet to achieve a given quality profile than competing source-domain techniques [14].

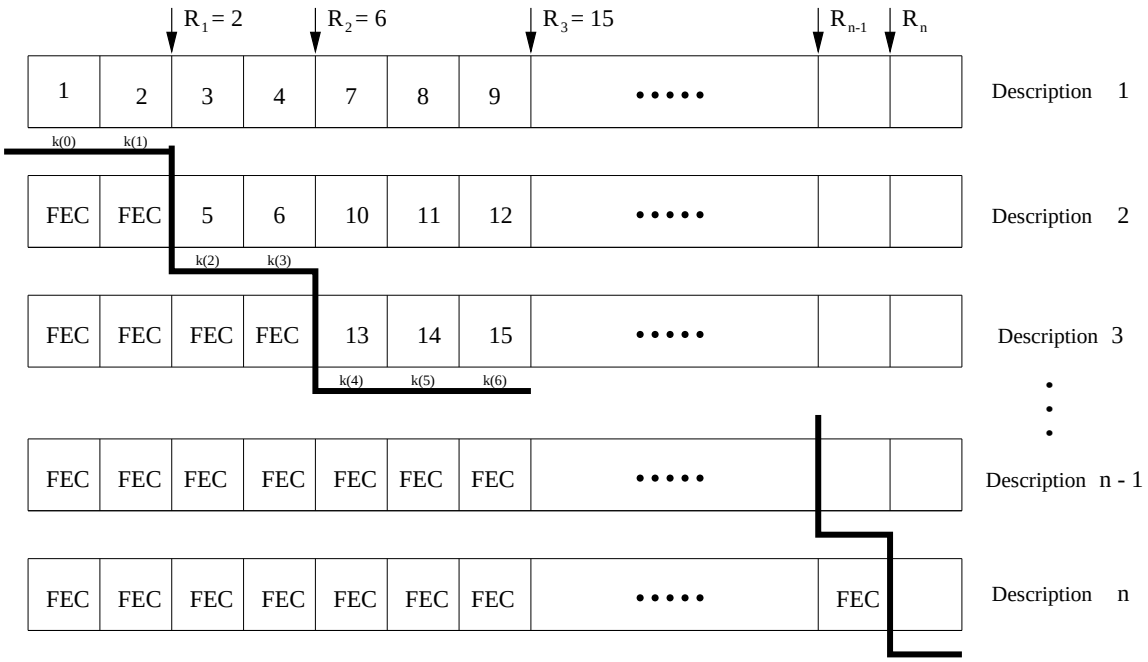

Figure 2.1 Multiple description using forward error correction

2.2 Multiple Description for Wireless Image Transmission

The main innovation of this work is the application of multiple-description techniques to the problem of transmitting images over wireless fading channels. To apply multiple-description techniques, we rely on a product code framework that uses a two-layer concatenated code. The outer code is an RCPC and CRC error-correction and -detection code; the inner code is the multiple-description SPIHT-encoded image described in the previous section. The inner MD-FEC encoding allows the quality of the image decoded at the receiver to degrade gracefully as the number of packets corrupted beyond the ability of the outer RCPC codes to correct increases. However, this encoding is also critically dependent on appropriate choices for the inner and outer code rates to correctly match the image being encoded to the characteristics of the fading channel. By combining a fast, nearly-optimal Lagrange optimizer for the Reed-

Solomon row codes with this concatenated-code framework, we create an efficient, practical image transfer system which gives results that are better than the current state of the art.

The MD-FEC optimizer allocates the unequal protection Reed-Solomon codes based on the rate-distortion curve and the channel state in terms of a loss profile (called the "channel PMF" in [13]). Once the code structure is determined, the data is split into packets and the Reed-Solomon forward error correction is added. Finally, the outer packet error detection and correction codes are added, and the resulting packets are transmitted to the receiver.

The exact nature of the error-correction and -detection code is not important; a variety of different types of codes, including both conventional convolutional codes and more advanced codes, such as turbo codes [15] or low-density parity check (LDPC) codes [16], can be used as long as the codes can protect the packets against bit errors and detect decoding failure with a high probability. Because of their relative simplicity and the ease with which a variety of code rates can be generated, the rate-compatible punctured convolutional (RCPC) codes described in [17] are used. The codes used are based on a memory 4, rate 1/3 mother code; a variety of punctured rates, ranging from 8/9 to 1/3, are available. Included in the RCPC-encoded data is a 16-bit cyclic redundancy check (CRC), an effective error-detection code with a low probability of false positives.[2] A block diagram of the system is included as Figure 2.2. The complete code structure, including the inner MD-FEC and outer Reed-Solomon codes, is shown in Figure 2.3.

Since the MD-FEC code design depends on many parameters, including the rate-distortion curve of the image being sent (which is not available to the receiver), overhead information must be added to each packet to describe the structure of the inner Reed-Solomon codes to the receiver. The required overhead includes the following:

[2]Note that the inner Reed-Solomon codes provide some protection against undetected erroneous packets as well, so the improbable event of a false positive does not significantly hurt the system's performance.

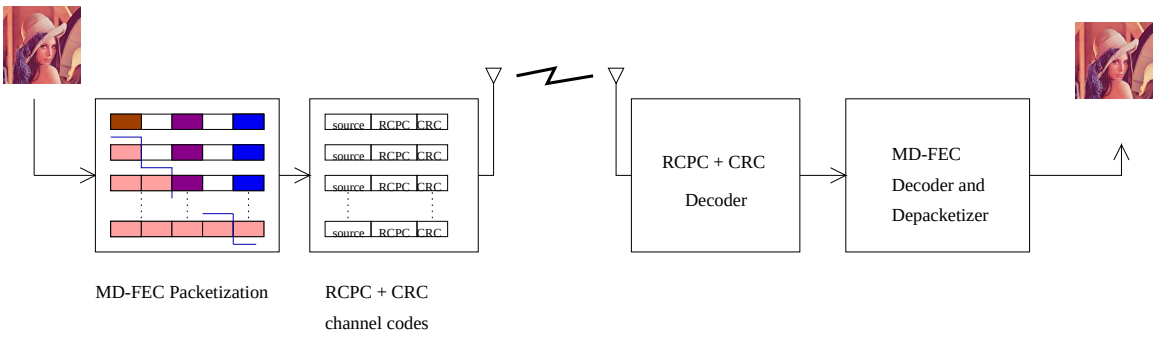

Figure 2.2 Block diagram for the proposed system

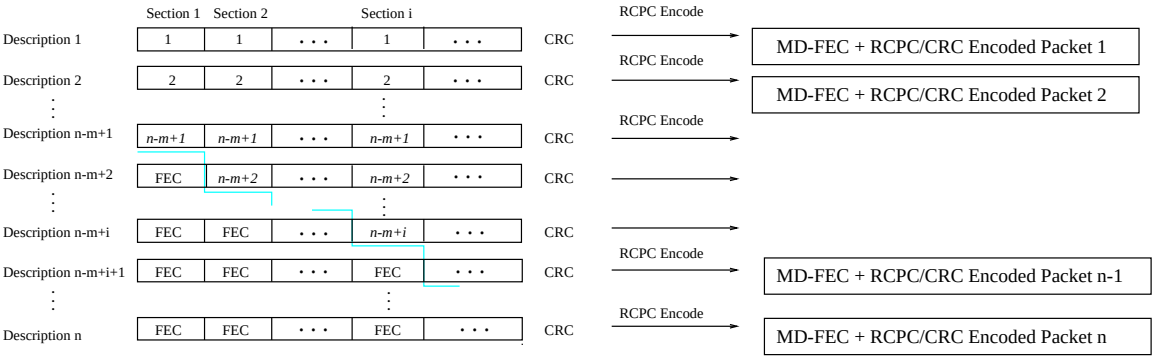

Figure 2.3 Multiple description for wireless image transmission

- One Reed-Solomon symbol per packet, which encodes the number of data symbols in the first column.

- Two Reed-Solomon symbols as the first data symbols in each protection level. These symbols encode the length of the current protection level (in columns)[3] and the number of data symbols per column in the next protection level.

[3]This is limited to describing a number less than or equal to the number of elements of the Galois field from which the Reed-Solomon codes are derived. Typically, a Galois field of size $2^8 = 256$ is used. If more than 256 Reed-Solomon columns are generated, and a protection level is allocated more than 256 columns, one of two approaches can be taken. Either more symbols can be used to describe the length of each protection level (using two symbols, a total of three symbols' overhead per level, allows 65 536 columns), or the protection level/length pair can be repeated every 256 columns. None of the cases examined in this thesis used more than 256 columns.

Although the distinction is artificial, for our packetization scheme it is convenient to think of the RCPC+CRC codes as being "channel" codes and the Reed-Solomon codes as being part of the "source" code. The RCPC codes allow reasonable confidence that the transmitted descriptions will be received correctly except during deep fades; the added CRC codes allow the channel to be treated as an erasure channel by preventing erroneous data from being decoded and processed.

Distinguishing between the RCPC+CRC "channel" codes and the Reed-Solomon "source" codes allows us to analyze the transmission of the embedded image over a wireless channel without considering individual bit errors. Since the CRC codes essentially prevent any bit errors introduced by the channel from being seen by the "source" codes, we can assume that there are no bit errors in the channel. We therefore view the wireless channel and the RCPC+CRC codes as a system for transmitting multiple descriptions of an image with some probability of successfully delivering different subsets of those descriptions.

This encoding improves on the unequal protection encoding proposed by Sherwood and Zeger in [4] in that in ours, all packets have essentially identical importance,[4] whereas in their scheme packets are not all of equal importance. Their scheme cannot recover the image if certain packets are lost. In fact, even with their most powerful unequal protection scheme, the loss of only 12 out of 140 packets can result in the loss of the first 200 bits of the image description, rendering the entire image irrecoverable. In our scheme, no such pathological case exists. On the other hand, in most cases their scheme incurs less of a delay before a useful subset of the image data can be decoded, since the contiguous symbols in the packets correspond to consecutive

[4]All packets are identically important if the possibility of recovering part of a column, when fewer packets than the number of data symbols in that column are received, is ignored.

source symbols. Our proposed approach distributes the data across packets, preventing the decoding of image data until a significant subset of packets has arrived.

CHAPTER 3

CODE ALLOCATION ALGORITHMS FOR WIRELESS CHANNELS

In this chapter, we describe several algorithms which allocate the protection levels for the inner Reed-Solomon codes so as to minimize an objective measure of image distortion. Efficient techniques for this are key to the wireless image-transmission system described in the previous chapter, because the number of possible code allocations is large and an exhaustive search is impractical. In this chapter, we develop a computationally tractable method for deriving the optimal protection levels, and the corresponding expected performance, given a packet-loss profile. We then develop an efficient algorithm (which takes time that is linear in the number of packets sent) which closely approximates this optimal solution.

Unlike the Reed-Solomon codes, the outer RCPC codes offer only a small number of possible rates; in our experiments, nine distinct rates were available. Since each packet must be sent with the same RCPC code rate (to do otherwise would violate the assumption that, a priori, each packet is equally likely to be lost), optimizing between the outer RCPC codes and the inner Reed-Solomon codes can be done using an exhaustive search if full channel-state information is available. However, since perfect channel-state information is often unavailable, optimization of the RCPC code rate is examined briefly as well.

3.1 Optimization Problem Statement and Solution

A complete code allocation strategy π consists of a protection level π_i for each column i (of ℓ) in the multiple-description code. The protection level chooses the number of source data symbols in that column; the remaining symbols are Reed-Solomon check symbols used as an erasure code. (Each parity symbol can correct one row erasure in that column.) Since embedded coders such as SPIHT lose synchronization when a bitstream with even one bit error is presented, each column is required to have at least as much protection as the previous column. When this is done, the decoded SPIHT bitstream will always be a contiguous block starting with the first symbol. For simplicity, we assume that any column i that does not contain at least π_i symbols is completely unrecoverable.

We can solve this allocation problem by noting that the optimal protection level for the current column must be the level that provides the largest combined MSE reduction for the current and all remaining columns. Because choosing the protection level for the current column affects the protection levels that can be chosen for the remaining columns, a greedy algorithm — choosing the protection level that gives the largest reduction in MSE for this column — is not globally optimal. Note, however, that a naive implementation of this algorithm results in a computational complexity that is exponential in the number of columns, and cannot be implemented for reasonable problem sizes.

Dynamic programming can be used to significantly decrease the cost of finding the optimal solution exactly. The dynamic programming solution has a cost that is a polynomial of the input size; the total number of subproblems is reduced to $O(R^2)$, and *memoization* (the recording of computed subproblem solutions in a table for future use) is used to return the solution for an already-solved subproblem in constant time. The reduction in the number of subproblems

comes from the observation that we need only three pieces of information to uniquely identify a subproblem: the number of data symbols sent in all of the columns so far, which can take R possible values; the current column number, which can take ℓ values; and the number of data symbols in the previous column, which can take p possible values. This means that the number of subproblems to be solved, and therefore the size of the table, is $O(Rp\ell) = O(R^2)$. Solving each subproblem requires $O(p)$ time, plus the time required to execute any recursive calls to calculate table values which are not yet filled in. Since only $O(R^2)$ table entries exist, the entire problem can take no more than $O(p) \cdot O(R^2) = O(pR^2)$ time.

While the dynamic programming solution does reduce the running time of the problem to a polynomial of the input size, it requires a very large table to accomplish this,[1] and still takes enough time that it cannot be implemented practically on embedded hardware. Since the optimal solution is too complex to be practically implemented, it is worth exploring other solutions to this allocation problem.

One way of reducing both the table size and the running time of this dynamic programming optimizer is to approximate the rate-distortion curve of the progressive image coder with a linear function [3]. Making this approximation reduces the table size to $O(R)$ and the running time to $O(pR)$. This is because each additional data symbol reduces the cost of the solution by exactly one unit, so it is not necessary to examine the number of previously transmitted data elements when doing a table lookup. However, since the SPIHT rate-distortion curve is not even approximately linear (it is approximately exponential), this approximation can result in a significant reduction in expected performance at the receiver.

[1] The table generated in solving the 24-packet, 0.237-bpp, rate 4/7 case was approximately 75 Mbytes.

3.2 An Optimizing Heuristic Based on Lagrange Multipliers

There have been several optimizing techniques proposed recently to cut down on the cost of the optimal allocation described above. These include approximating the rate-distortion curve of the image being transmitted as a linear function [3], greedy algorithms [12], and gradient-descent algorithms [13]. Except for the gradient-descent algorithm (for which it is difficult to define an exact running time), all of these algorithms improve running time to $O(R)$ at the cost of a small increase in expected mean-squared error at the receiver. However, the work of R. Puri and K. Ramchandran in [14] offers an optimization algorithm that uses Lagrangian optimization to solve a related continuous optimization problem. The advantage of this technique over competing techniques is that given a convex rate-distortion curve, this technique can find a bit allocation in $O(p)$ time. The following description of this technique is adapted with permission from [14].

Unlike the dynamic-programming technique described above, which evaluates the rate partitioning on a column-by-column basis, the Lagrangian optimizer achieves its speedup by placing the boundaries between the different protection levels. It optimizes a rate partitioning $\mathbf{R} = R_0, \ldots, R_{N-1}$, where each of the rates R_i represent the number of source symbols that can be successfully decoded if $i+1$ descriptions arrive intact at the receiver. (Equivalently, R_i is the number of source symbols in column codes that have no more than $i+1$ data symbols.) This algorithm solves a continuous version of the discrete problem above; it does not address the fact that each column must be an integral number of Reed-Solomon symbols wide.

Note that $R_i \leq R_j$ must hold for $i < j$ from the requirement that the number of data symbols in a column is a nondecreasing function of the column number.

We can define the expected distortion $E[D]$ based on the rates R_i as follows:

$$E[D] = q_{-1} \cdot E + \sum_{j=0}^{N-1} q_j \cdot D(R_j), \tag{3.1}$$

where E is the source variance, or the mean squared error incurred when the source is represented by zero bits, and q_i is the probability that $i+1$ packets were successfully received at the receiver. The total rate used is

$$R_t = \sum_{j=0}^{N-1} \alpha_j \cdot (R_j) \tag{3.2}$$

where

$$\alpha_j = \frac{N}{(j+1)(j+2)}, \quad j = 0, \ldots, N-2$$

and $\alpha_{N-1} = 1$.

Problem Statement: Given the number of packets p, each packet of size ℓ (i.e., the total rate budget is $R^* = p \cdot \ell$), and a progressive bitstream with rate-distortion curve $D(r)$, minimize $E[D]$ subject to the following two constraints:

$$R_t \leq R^* \tag{3.3}$$

$$R_0 \leq R_1 \ldots \leq R_{p-2} \leq R_{p-1} \tag{3.4}$$

There is also a third constraint, which we ignore for now:

$$R_i - R_{i-1} = k_i(i+1) \tag{3.5}$$

This constraint (3.5) exists due to the fact that each protection level must span an integral number of columns.

Subject to constraint (3.3) only, the solution can be easily found using the theory of Lagrange multipliers [14], [18]. Also, if $\frac{\alpha_n}{q_n} > \frac{\alpha_{n+1}}{q_{n+1}}$, then constraint (3.4) is automatically satisfied, since

the product of the rate-distortion curve slope at point R_i and $\frac{\alpha_n}{q_n}$ is a constant, and, since the rate-distortion curve is convex, its slope is monotonically increasing.

In general, however, $\frac{\alpha_n}{q_n}$ may not be monotonically decreasing. Therefore, satisfying constraint (3.4) in the general case requires a result from the analysis of this algorithm [14]:

Result *If* $\frac{\alpha_n}{q_n} < \frac{\alpha_{n+1}}{q_{n+1}}$, *then* $R_n = R_{n+1}$ *in the optimal solution.*

Proof Assume that there exists an optimal solution such that $R_n < R_{n+1}$. Then take away ΔR bits from R_{n+1} and give $\Delta R \cdot \frac{\alpha n+1}{\alpha n}$ bits to R_n (so that the rate constraints (3.3) and (3.4) are satisfied). Now, the net decrease in our cost is

$$Decrease = \Delta R \cdot \alpha_{n+1} \left[\frac{q_n}{\alpha_n} \lambda(R_n) - \frac{q_{n+1}}{\alpha_{n+1}} \lambda(R_{n+1}) \right] \tag{3.6}$$

Since $\frac{\alpha_n}{q_n} < \frac{\alpha_{n+1}}{q_{n+1}}$ and $\lambda(R_n) > \lambda(R_{n+1})$, the decrease is positive, contradicting our assumption that the original solution is optimal. Therefore, in the optimal solution $R_n = R_{n+1}$. Q.E.D.

In other words, in the case where, for some n, $\frac{\alpha_n}{q_n} \geq \frac{\alpha_{n+1}}{q_{n+1}}$, the optimal solution to the original problem is the same as that to a reduced problem where $R_n = R_{n+1}$ is replaced by $R_n^{'}$ so that $\alpha_n^{'} = \alpha_{n+1} + \alpha_{n+2}$ and $q_n^{'} = q_{n+1} + q_{n+2}$. We can therefore solve the reduced problem where $\frac{\alpha_n}{q_n} < \frac{\alpha_{n+1}}{q_{n+1}}$, automatically satisfying constraint (3.4).

However, this solution (which is optimal to within a convex-hull approximation of the rate-distortion curve of the coded image) does not take into account the Reed-Solomon coding constraint (3.5). We currently do not know if an elegant, optimal solution which accounts for this constraint exists, but we can use a simple heuristic to map the optimal solution into a nearby solution in the space of allowable configurations for the MD-FEC transcoder: decrease each R_i until $R_i - R_{i-1}$ becomes an integer multiple of $i+1$, then increase the last R_i for which

$R_i - R_{i-1}$ is nonzero until $R^* = R_t$. This heuristic biases the solution toward increased peak PSNR at the receiver. While other heuristics are possible, we have observed that in simulations the performance of the heuristic solution is always very close to that of the optimal solution. Also, this Lagrange-based optimizer is much faster than competing techniques in the literature, such as the greedy descent algorithm proposed by Mohr et al. [13]. The running time of the Lagrange optimizer is $O(p)$ per value of λ tested; the value for λ is found using a fast bisection search and in testing is usually found within 32 iterations.[2] Although the required R-D curve can be derived with little extra computation from the SPIHT encoder, the rate-distortion curve sent to the optimizer must be convex. Computing a convex hull takes $O(R \log R)$ time. However, since the SPIHT rate-distortion curve is approximately convex when sampled at widely spaced intervals, we can avoid this step at a small cost to expected image quality at the receiver.

As the MD-FEC optimizer uses the rate-distortion curve of the image being transmitted (which is not available to the receiver) in its calculations, the code allocation must be described by the packets sent to the receiver. Although these per level overhead costs are not considered in the optimization process, which assumes that there is no added cost for creating a new protection level, the added per-level cost of describing the code configuration to the receiver is included when the final performance figures are calculated. Typically, few protection levels are actually used, and the bits used to describe the code allocation to the receiver have only a modest impact on the expected performance at the receiver.

3.3 Optimizing RCPC Code Rates

The above discussion reveals several algorithms for allocating the protection levels for the inner Reed-Solomon codes in a source-aware manner. However, in addition to allocating the

[2]Note that λ can be found to within any arbitrary tolerance in constant time using a bisection search.

Reed-Solomon code rates, bits must be allocated to the outer RCPC+CRC codes. This is simplified by the fact that all packets are equally important, and therefore unequal protection on the packets is neither necessary nor desirable; thus, only a few possibilities exist for the outer code rate. (In our code, nine possible rates were available from the RCPC encoder and decoder.) Because the Reed-Solomon code rate optimizer tells us not only the optimal code allocation, but also the expected distortion (cost) of that allocation at the receiver, we can simply run the optimization on all possible outer code rates, then use the outer code rate that results in an inner code that has the lowest expected distortion at the receiver.

However, this is a somewhat naive view of the rate-allocation problem. While it is true that the optimization problem can be reasonably solved by a brute-force search, this does not address the fact that while the packet-loss profile of the current RCPC code rate may be available (if, for example, the receiver relays the number of packets received from each image to the transmitter using a feedback channel), the packet-loss profile from other RCPC code rates is hard to derive from this information.[3] It is therefore reasonable to make the assumption that the only packet-loss profile available at any given time is that of the current RCPC code rate. We will further assume that the expected MSE at the receiver is a unimodal function of the RCPC code rate. If this is the case, we can do a reasonable job of estimating the correct RCPC code rate to use by simply doing a gradient-descent search. The code rate is changed after a sufficient number of images have been sent to develop an accurate channel profile.

The code rate is stepped in the same direction as the last step if overall performance improved, or in the opposite direction if overall performance suffered. Imperfect performance

[3]Actually, the rate-compatible property of the RCPC codes can be used to estimate the packet-loss profile if less protection — a higher rate code — is sent, but it is hard to estimate whether additional parity would have allowed us to decode the packet.

estimates provide some protection against local maxima; if necessary, a stochastic algorithm (such as simulated annealing) could be used to find the global maximum.

Unfortunately, the gradient-descent optimization method described above has three important limitations. The first limitation is that it is slow; sufficient data to generate a reasonably accurate packet-loss profile — probably a hundred or more images — must be generated before the system can switch from one RCPC code rate to another. The second is that the optimizer can become stuck in local maxima; the third, and most important, is that by its nature the system cannot always remain in the state that achieves the best possible performance.

CHAPTER 4

SIMULATIONS AND PERFORMANCE ANALYSIS

We next analyze the performance of the multiple description product code described in Chapter 2. We evaluate its performance by simulating sending the 512 $\times$ 512 Lena image used by Sherwood and Zeger in [4] over a wireless channel. We also simulate sending the 512 $\times$ 512 "Fingerprint" image from the JPEG-2000 test image suite.

4.1 Channel Model and RCPC Codes

Because the algorithms above require a profile of the packet arrival probability to optimize the MD-FEC stream, we use a channel simulator to model the fading channel along with the RCPC and CRC channel codes. Since the multiple-description codes described above and the RCPC channel codes are two logically separate components of the system, we can generate the packet loss probabilities with the simulator, then analyze the performance of the overall system using those probabilities.

Our evaluation of the concatenated RCPC+CRC and Reed-Solomon code model for sending images over a wireless channel uses the same channel model proposed by Sherwood and Zeger [4]: a Clarke flat-fading channel with an average signal-to-noise ratio of 10 dB and a normalized Doppler frequency of 10^{-5} Hz. This channel remains in a severely impaired state (with a bit

error rate of 0.1 or more) for approximately 10 000 bits [4]. BPSK (antipodal) signalling was simulated.

The RCPC encoder and decoder are based on code written by P. Karn [19] and were explicitly simulated. The MD-FEC (Reed Solomon codes) were simulated implicitly, and the CRC codes were assumed to detect any errors perfectly — any packet mismatch was assumed to cause a CRC error, and the CRCs were never explicitly calculated. Error detection was implemented by simply comparing the transmitted and decoded data bit-for-bit.

List-Viterbi decoding was used; if none of the ten lowest-metric paths matched the packet being sent, the packet was declared to have been lost. In these simulations, the decoder used does soft decoding assuming a Gaussian channel with a 7.5-dB SNR, a bias which was found to improve performance. Performance could potentially be improved by providing a more accurate channel model to the RCPC decoder.

4.2 Product-Code Performance Evaluations

We evaluated the performance of the MD-FEC system by using the channel simulator and RCPC encoder described above. The RCPC code rate was optimized by simply choosing the code rate that resulted in the highest expected performance at the receiver. The packet-loss profile was fed into the MD-FEC optimizer, and the bit allocation was used along with the packet-loss profile to compute the expected performance at the receiver.

Peak and mean PSNR figures given here are based on simulation of the transmission of 5,000 images unless otherwise noted; all simulations tested at least 1,000 transmitted images. The initial state of the fading simulator was chosen independently and randomly for each image transmitted. Except where noted, all mean and peak PSNRs given for MD-FEC codes account for the overhead required to describe the code configuration to the receiver. Note that unlike

Mohr et al., Sherwood and Zeger define expected PSNR at the receiver as the expected mean-squared error, converted to PSNR in the usual way. The same definition is used here for comparisons against their work.

The first test we performed was comparing the performance of the proposed system against the system presented by Sherwood and Zeger in their ICIP '98 paper [4]. To make this comparison, we started with simulations using the same packet length and number of packets: 200 bits of information per packet were encoded into 444 channel bits, protected by a rate 1/2 RCPC+CRC code. The information bits stored in the packet consist of source data encoded with either the Sherwood and Zeger "UEP2" scheme or our MD-FEC row codes. Two images were tested: the JPEG-2000 "fingerprint" test image, and the 512×512 Lena image used by Sherwood and Zeger in their testing.

Our implementation of the Sherwood and Zeger "UEP2" code required that we choose the number of packets n such that $n \bmod 16 = 12$; to achieve this at a bit rate of approximately 0.25 bpp, 140 packets were used. This resulted in a total rate, including both source and channel codes, of 0.237 bits per pixel (bpp). This is equivalent to a total rate of 62,160 bits per image, including all source and channel codes. At this rate, our implementation of the Sherwood and Zeger "UEP2" code achieves a mean MSE at the receiver corresponding to a PSNR of 27.75 dB. Sherwood and Zeger's implementation achieves a 27.815 dB expected MSE at the receiver. (The small difference may be accounted for by the use of a bit rate of 0.237 bpp instead of 0.25 bpp.) The peak delivered PSNR for our implementation of the Sherwood and Zeger code is 28.26 dB.

On the same channel, using the same 444-bit packet size, RCPC code rate, and list of packets that were successfully decoded at the receiver, our RCPC/CRC+MDFEC scheme using

the Lagrange optimizer achieves an expected MSE corresponding to a 27.90 dB PSNR, and a peak PSNR (achieved approximately 90% of the time) of 28.84 dB. This means that by simply substituting our source-optimized multiple-description row codes for Sherwood and Zeger's "UEP2" row code, we improve the mean PSNR at the receiver by 0.15 dB, and improve the perceptually important peak performance metric as well. Figure 4.1 shows a comparison of the performance of the Sherwood and Zeger codes with our MD-FEC approach.

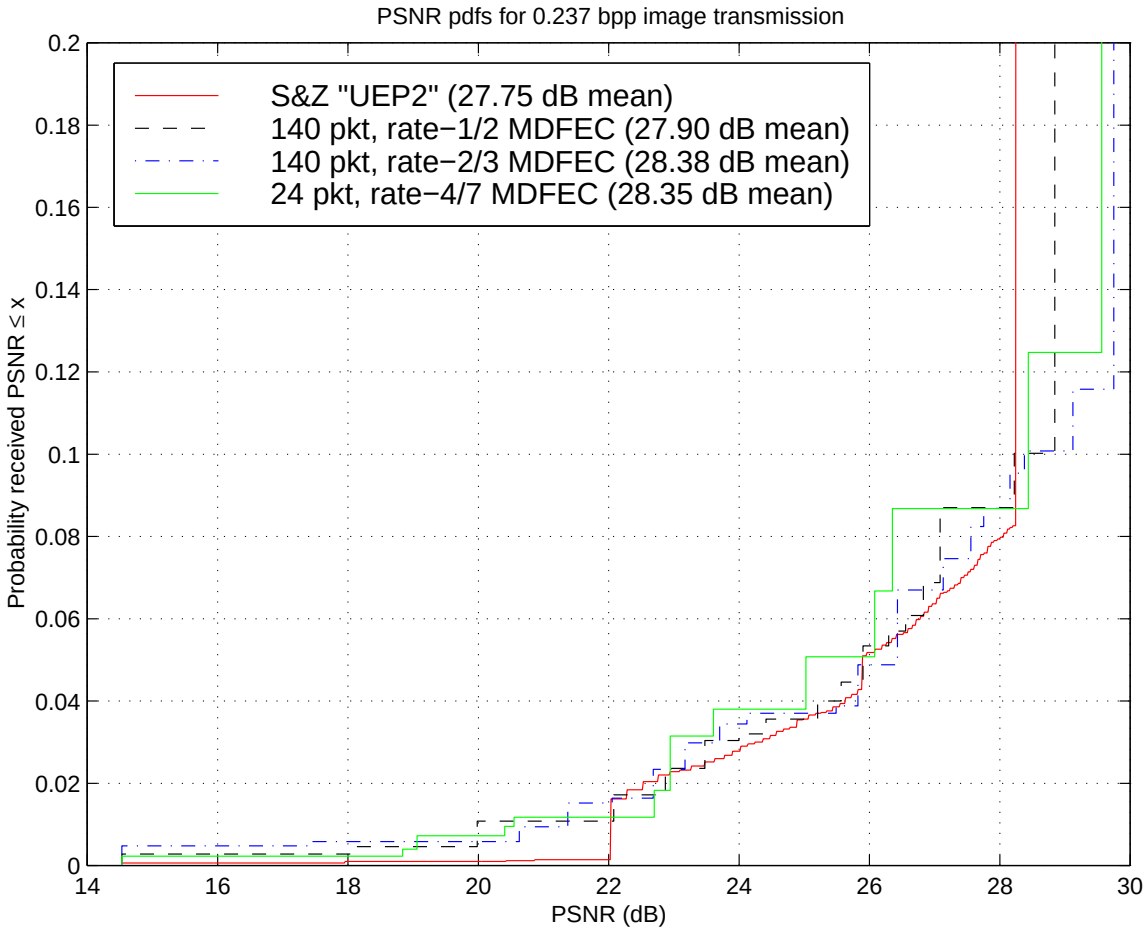

Figure 4.1 Comparison of RCPC/CRC+MDFEC and Sherwood and Zeger UEP2

This 0.15 dB gain over Sherwood and Zeger's "UEP2" scheme can be improved by better matching the packet length and RCPC code rate to the channel. The rate 1/2 RCPC code

Table 4.1 Performance comparison for 512 × 512 Lena image

Encoding	*Code Rate*	*Expected MSE*	*Peak PSNR*	*90% PSNR*
Sherwood and Zeger "UEP2"	0.28	27.75 dB	28.26 dB	28.26 dB
140-packet MDFEC rate-1/2 RCPC codes	0.30	27.90 dB +0.15 dB	28.84 dB +0.58 dB	28.22 dB -0.04 dB
140-packet MDFEC rate-2/3 RCPC codes	0.37	28.38 dB +0.63 dB	29.73 dB +1.49 dB	28.16 dB -0.10 dB
24-packet MDFEC rate-4/7 RCPC codes	0.35	28.35 dB +0.60 dB	29.56 dB +1.25 dB	29.44 dB +1.18 dB
1 bpp, 96-packet MDFEC rate-4/5 RCPC codes	0.46	36.34 dB	36.80 dB	36.34 dB

results in too much protection, and performance in the 140-packet case can be improved by simply increasing the RCPC code rate to 2/3. This change results in an expected PSNR at the receiver of 28.38 dB, an improvement of 0.63 dB over Sherwood and Zeger's "UEP2" code. The peak PSNR increases to 29.75 dB, achieved 88% of the time. Table 4.1 summarizes the above results for the Lena image.

We also examined the performance of both the Sherwood and Zeger "UEP2" and the wireless MD-FEC encodings on the more difficult JPEG-2000 "fingerprint" image. The wireless MD-FEC algorithm once again offers significantly improved performance, in terms of both peak PSNR and expected mean-squared error at the receiver. A summary of these performance results is included as Table 4.2; the probability distribution functions for the received PSNR are included as Figure 4.2.

4.3 Effects of Packet Length on Performance

It is also worth exploring the effect on performance of reducing the number of packets without changing the total number of channel bits transmitted. Although using fewer packets

Table 4.2 Performance comparison for 512 × 512 fingerprint image

Encoding	*Code Rate*	*Expected MSE*	*Peak PSNR*	*90% PSNR*
Sherwood and Zeger "UEP2"	0.28	26.21 dB	26.62 dB	26.62 dB
140-packet MDFEC rate-1/2 RCPC codes	0.34	26.47 dB +0.26 dB	26.91 dB +0.29 dB	26.39 dB -0.23 dB
140-packet MDFEC rate-2/3 RCPC codes	0.44	26.89 dB +0.63 dB	27.49 dB +0.87 dB	25.44 dB -1.18 dB
24-packet MDFEC rate-4/7 RCPC codes	0.41	26.86 dB +0.65 dB	27.35 dB +0.73 dB	27.35 dB +0.73 dB
1 bpp, 96-packet MDFEC rate-4/5 RCPC codes	0.49	32.94 dB	33.54 dB	32.65 dB

results in more data loss when an uncorrectable error occurs while decoding a packet, other factors can offset this limitation. This is because the short packets significantly increase the per-packet overhead (including Reed-Solomon code description, CRC, and trellis termination bits), and because the fade is so slow that even very large packets are shorter than the mean fade duration. More importantly, reducing the number of packets also reduces the number of erasures to be corrected in each column. This reduces the cost of the Reed-Solomon decoding at the receiver.

To determine the effects of increasing the packet length on performance, we tested several different packet lengths and RCPC code rates at the same total rate of 0.237 bpp. We found that good performance could be achieved by sending 24 packets of 2,590 bits each and using an RCPC code rate of 4/7. (Performance suffered significantly if fewer packets were used.) This structure resulted in a mean MSE at the receiver corresponding to a PSNR of 28.35 dB, an improvement of 0.60 dB over the performance of the Sherwood and Zeger "UEP2" scheme on the same channel and only 0.03 dB worse than the 140-packet case. The peak PSNR, achieved approximately 87% of the time, is 29.56 dB. For this 24-packet case, Figure 4.3 shows images

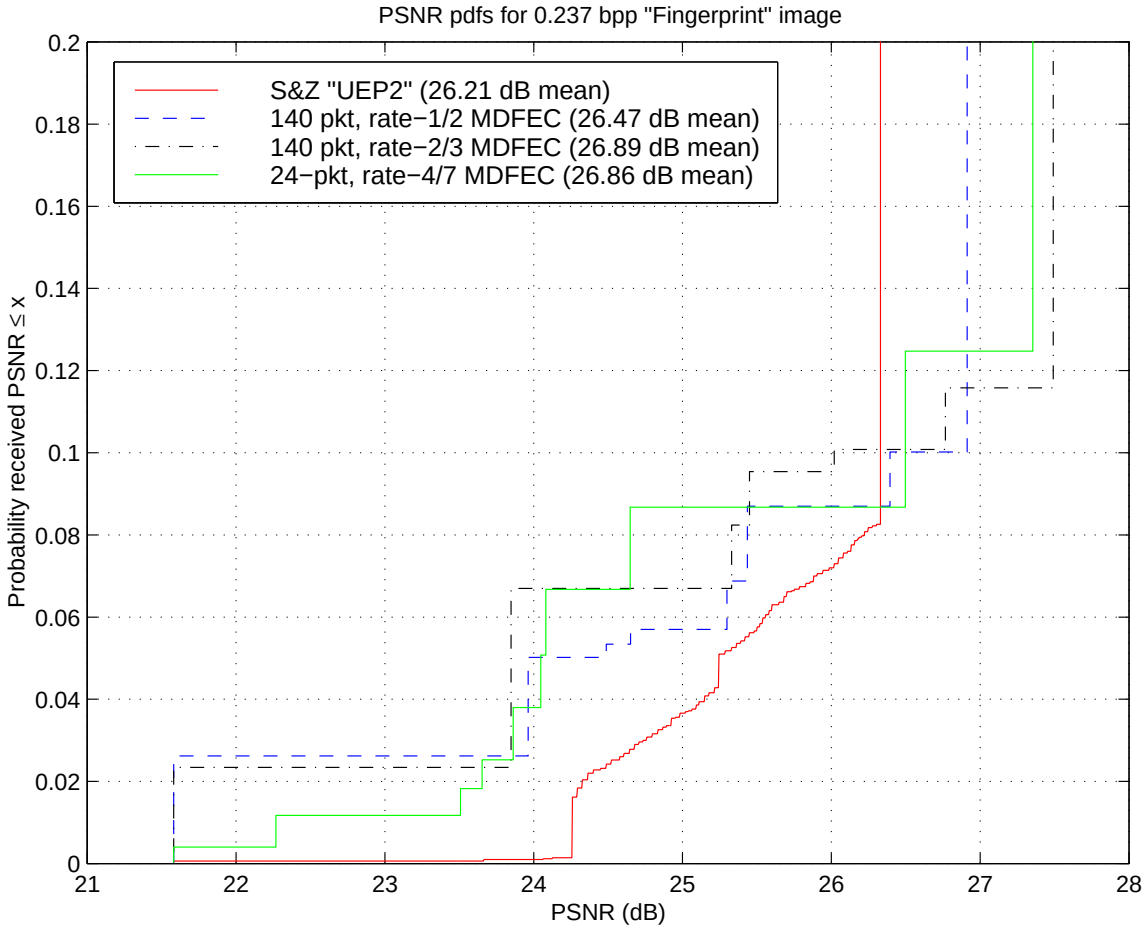

Figure 4.2 Comparison of RCPC/CRC+MDFEC and Sherwood and Zeger UEP2

with distortions greater than or equal to the image received 85%, 90%, 95%, and 99% of the time.

To verify that the Lagrange optimizer still performs well when fewer packets are used, we compared the expected performance for the 24-packet case against the true optimal solution, calculated using the $O(pR^2)$ dynamic program. Since none of the optimizers include the per-level cost as part of their optimization process, for this test we allowed the bits normally reserved to describe the Reed-Solomon codes to the receiver to be used to transmit source image data instead. If this is done, the heuristic optimizer results in an expected MSE of 28.41

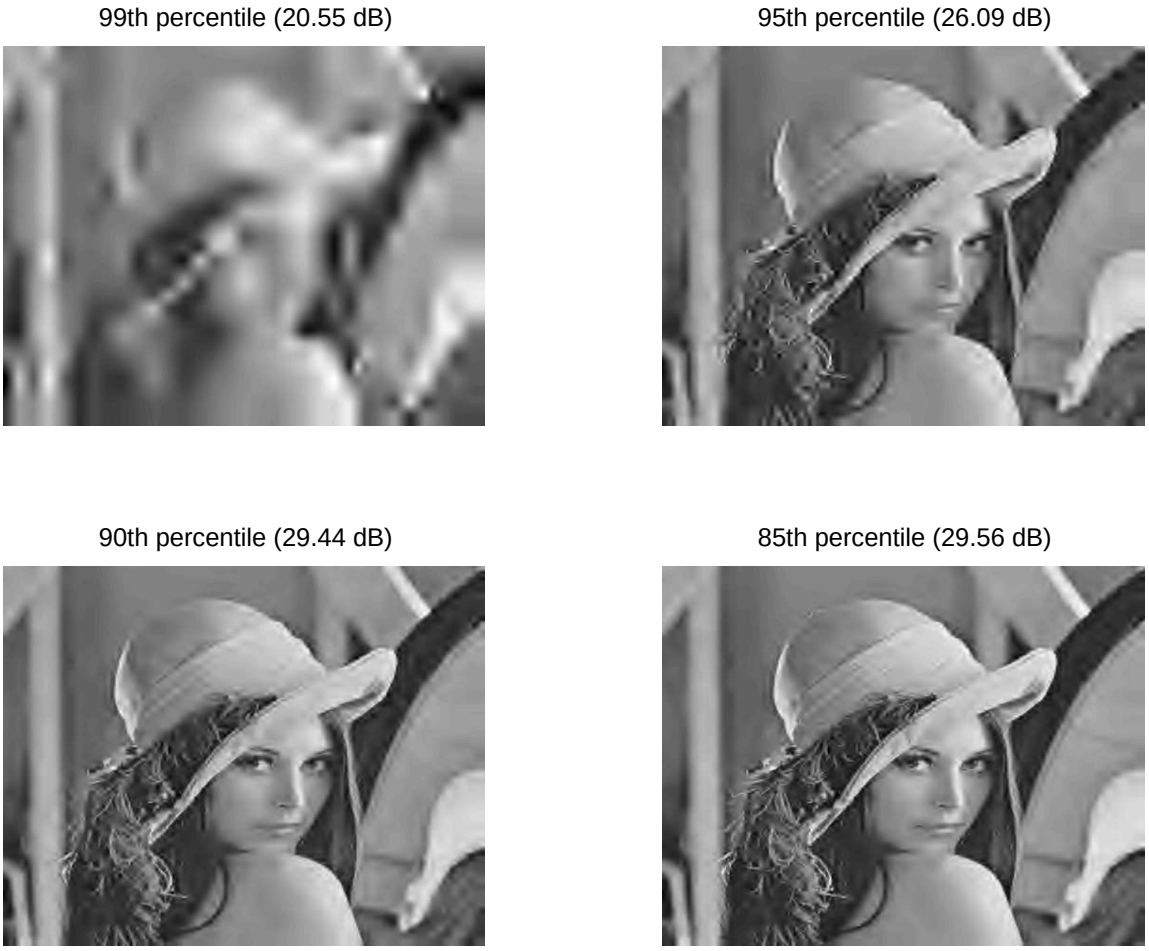

Figure 4.3 Received images at 0.237 bpp (24-packet, rate 4/7 RCPC codes)

dB (29.61 dB peak) at the receiver for the 24-packet, rate 4/7 case. The provably-optimal dynamic programming optimizer provides a solution with an expected PSNR of 28.48 dB, only 0.07 dB better.

Even when using a much smaller number of packets, our scheme provides a significant improvement (1.3 dB) over the Sherwood and Zeger "UEP2" scheme 87% of the time, while offering essentially equivalent performance when many packets are lost.[1] Using fewer packets

[1] The Sherwood and Zeger "UEP1" scheme improves peak performance by approximately 0.5 dB, but only achieves this peak approximately 75% of the time.

also simplifies decoding and reduces the cost of any additional per-packet overhead, such as sequence numbers, that may be required.

Due to the fact that 0.237 bpp is not sufficient to send the 512×512 Lena image with good perceived quality, the performance of the MD-FEC scheme at a rate of 1 bpp was also tested. The PSNR pdf for MD-FEC at 1 bpp is included below as Figure 4.4. The configuration for this test was 96 packets of 2,730 bits each; the RCPC code rate used was 4/5. Based on 1,000 simulated image transmissions, the expected PSNR at the receiver was 36.34 dB for this configuration, and the peak PSNR (achieved approximately 90% of the time) was 36.80 dB. Figure 4.5 shows images with distortions greater than or equal to the distortion of the image received 90%, 95%, 98%, and 99% of the time.

Tables 4.1 and 4.2 above show the results for 0.237 bpp using 24 packets and 1 bpp using 96 packets for both the Lena and JPEG-2000 "fingerprint" image.

4.4 Effect of RCPC Code Rate on Performance

To validate the assumption that the overall performance is a convex function of the RCPC code rate, Figure 4.6 compares the expected PSNR at the receiver to the RCPC code rate used for the three packet structures evaluated above. While these curves show that the performance is approximately a nearly-convex function of the RCPC code rate, small local maxima are visible.

If we optimize across RCPC code rates using the gradient-descent method outlined above, in steady state the RCPC code rate will oscillate around the peak if we assume perfect performance estimates and that the underlying RCPC code rate versus expected performance curve is unimodal. Under these conditions, the gradient-descent RCPC code optimizer will send approximately one-fourth of the images with an RCPC code rate one step below optimal, another

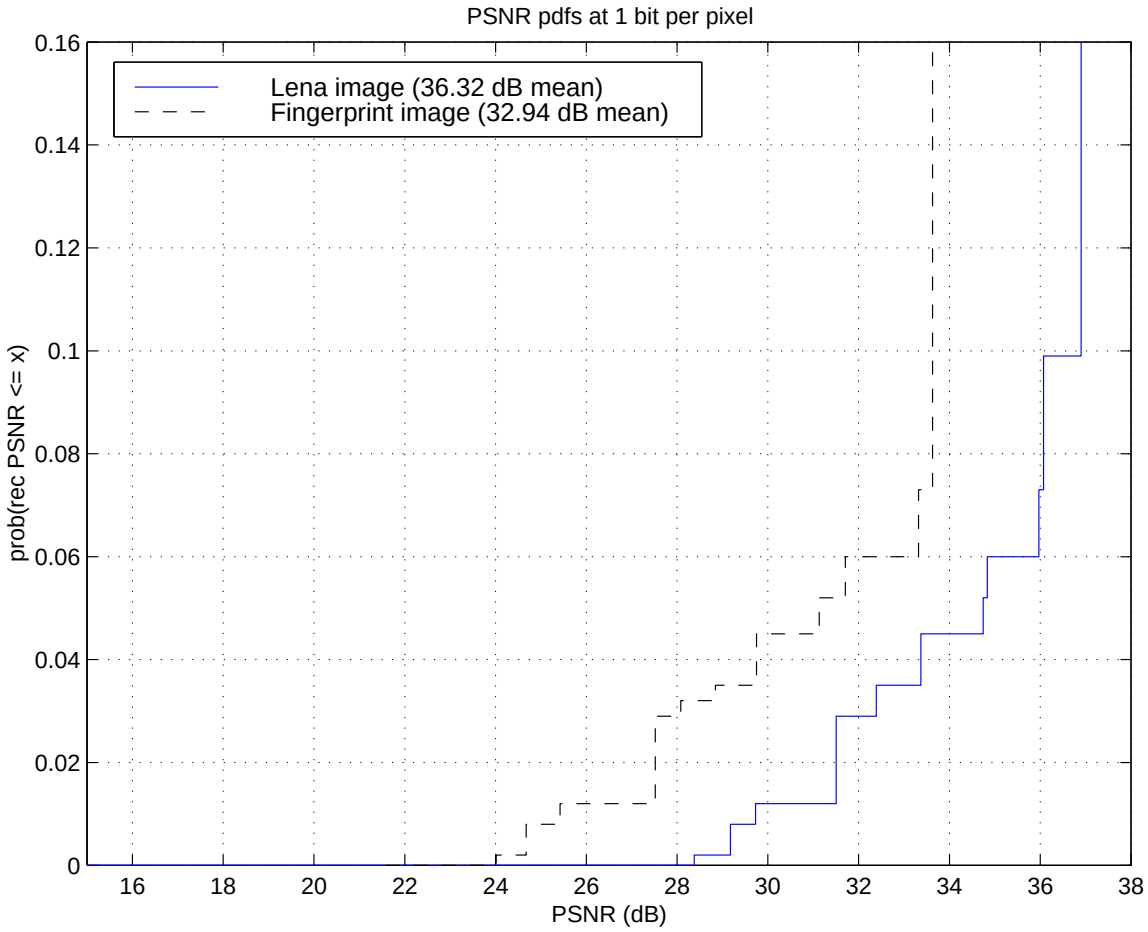

Figure 4.4 Performance of MD-FEC system at 1.0 bpp

one-fourth of the images with an RCPC code rate one step above optimal, and the remaining one-half of the images at the optimal RCPC code rate.

This means that the expected mean-squared error of the gradient-descent RCPC optimizer for the Lena image transmitted at 0.237 bpp is 28.23 dB for the 140-packet case, and 28.26 dB for the 24-packet case. Compared to always choosing the optimal RCPC code rate, we suffer an expected MSE penalty of approximately 0.1 dB for the 24-packet case, and slightly more than 0.1 dB for the 140-packet case. For the 1-bpp case, the mean MSE at the receiver would be 36.30 dB, a trivial (0.04 dB) penalty.

99th percentile (30.73 dB)

95th percentile (34.73 dB)

90th percentile (36.34 dB)

85th percentile (36.80 dB)

Figure 4.5 Received images at 1 bpp (96-packet, rate 4/5 RCPC codes)

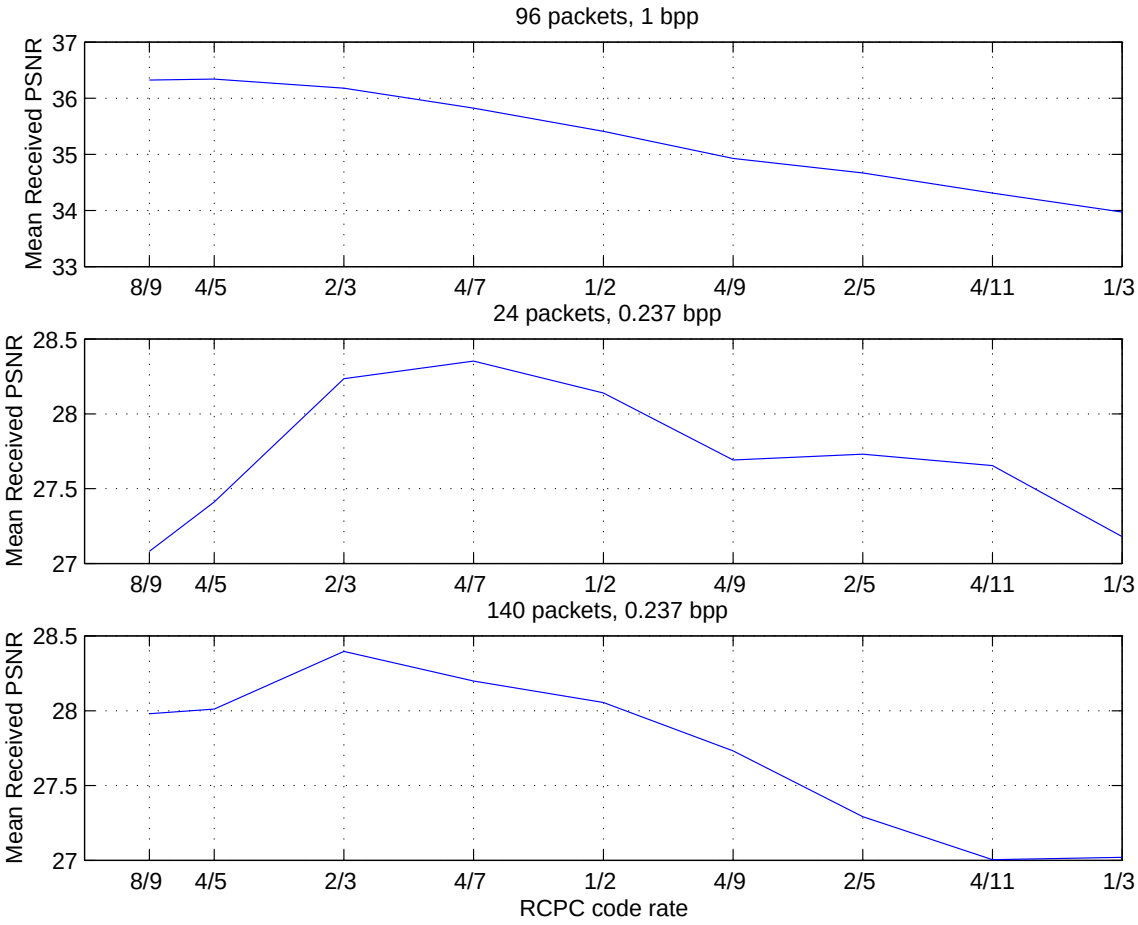

Figure 4.6 RCPC code rate versus expected received image quality

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

In this work, we have presented a source-channel framework for transmitting images over wireless channels which combines Sherwood and Zeger's channel-coding approach with an efficient multiple-description scheme based on forward error correction. By tackling the wireless image transmission problem using a multiple description approach, we achieve graceful degradation as channel fade worsens, along with good performance when the channel is clean. We improve on the "unequal protection" code described by Sherwood and Zeger by transcoding the SPIHT bitstream into descriptions that have truly equal importance. This system represents a general framework for image transmission, which allows the choice of arbitrary rates and can be used for any progressive image encoding. We have shown that this framework offers better expected and peak performance than Sherwood and Zeger's encoding, even if many fewer packets are used.

One important point requiring future work is the bit allocation between the outer RCPC codes and the inner multiple-description code. Presently, we do an exhaustive search over all the RCPC code rates to come up with the optimal allocation strategy. Because the number of RCPC code rates is small, this is a reasonable strategy if a packet-loss profile is available for all possible RCPC code rates. However, in practice, estimating the packet-loss profiles for all possible RCPC code rates is difficult. A gradient-descent scheme has been presented that simplifies the

coupled channel-estimation problem, although this scheme has several undesirable properties. Future work could include research on gradient-descent strategies and novel approaches to the channel-estimation problem.

Another important topic for future work is the extension of this scheme to the wireless video transmission problem. These techniques can easily be extended to apply to video transmission through the use of a progressive video encoder such as 3D-SPIHT. By combining a progressive image encoder with MD-FEC transcoding, we can prevent packet losses from causing missing frames or portions of images. Rather than dropping portions of frames, such a system could reconstruct complete, though less detailed, images during periods of fade.

One of the most promising aspects of this work is the fact that the multiple-description encoding that protects the transmitted images against fades also makes the scheme robust against packet losses outside the wireless link. Because the inner MD code optimization is dependent only on the end-to-end packet-loss distribution and all packets have equal importance, whether packets are lost to the wireless channel or to network congestion is irrelevant. Therefore, this scheme is ideally suited for the hybrid packet-loss and wireless networks toward which today's Internet is evolving.

REFERENCES

[1] W. Weaver and C. E. Shannon, *The Mathematical Theory of Communication*, Urbana, IL: University of Illinois Press, 1949, republished 1963.

[2] G. Davis and J. Danskin, "Joint source and channel coding for image transmission over lossy packet networks," in *SPIE Conference on Wavelet Applications of Digital Image Processing XIX*, 1996, pp. 376-387.

[3] V. Chande and N. Farvadin, "Joint source-channel coding for progressive transmission of embedded source coders," in *IEEE Data Compression Conference*, March 1999, pp. 52-61.

[4] P. G. Sherwood and K. Zeger, "Error protection for progressive image transmission over memoryless and fading channels," in *International Conference on Image Processing*, vol. 1, October 1998, pp. 324-328.

[5] D. Sachs, A. Raghavan, and K. Ramchandran, "Wireless image transmission using multiple-description based concatenated codes," to appear in *SPIE Conference on Image and Video Communications and Processing 2000*, January 2000.

[6] W. C. Jakes, Jr., Ed., *Microwave Mobile Communications*. New York: John Wiley and Sons, 1974.

[7] A. Said and W. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 6, pp. 243-250, June 1996.

[8] P. G. Sherwood and K. Zeger, "Progressive image coding on noisy channels," in *IEEE Data Compression Conference*, March 1997, pp. 72-81.

[9] S. Appadwedula, D. Jones, K. Ramchandran, and L. Qian, "Joint source channel matching for wireless image transmission," in *International Conference on Image Processing*, vol. 2, October 1998, pp. 137-141.

[10] V. A. Vaishampayan. "Design of multiple description scalar quantizers," *IEEE Transactions on Information Theory*, vol. 39, pp. 821-834, May 1993.

[11] N. Varnica, M. Fleming, and M. Effros, "Multi-resolution of the SPIHT algorithm for multiple description," in *IEEE Data Compression Conference*, March 2000, pp. 303-312.

[12] A. Mohr, E. Riskin, R. Ladner, "Approximately optimal assignment for unequal loss protection," December 1999, http://rcs.ee.washington.edu/dcl/amohr/papers/icip00 /amohr-icip00-draft.ps.

[13] A. Mohr, E. Riskin, and R. Ladner. "Graceful degradation over packet erasure channels through forward error correction," in *IEEE Data Compression Conference*, March 1999, pp. 92-101.

[14] R. Puri and K. Ramchandran, "Multiple description coding using forward error correction codes," in *Proceedings of the 33rd Asilomar Conference on Signals and Systems*, vol. 1, October 1999, pp. 342-346.

[15] C. Berron, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *IEEE Conference on Communications*, May 1993, pp. 1064-1070.

[16] M. Davey and D. MacKay, "Low density parity check codes over GF(q)" *IEEE Communications Letters*, vol 2, no. 6, pp. 165-167, June 1998.

[17] J. Hagenauer, "Rate-compatible punctured convolutional codes and their applications," *IEEE Transactions on Communication*, vol. 36, pp. 389-400, April 1988.

[18] D. P. Bertsekas. *Nonlinear Programming.* Belmont, Massachusetts: Athena Scientific, 1995.

[19] P. Karn, "Forward error correcting codes," August 1999, http://people.qualcomm.com /karn/code/fec/.