# UNEQUAL ERROR PROTECTION FOR TRANSMISSION OF JPEG2000 CODESTREAMS OVER NOISY CHANNELS

*Zhenyu Wu, Ali Bilgin and Michael W. Marcellin*

Dept. of Electrical and Computer Engineering, The University of Arizona, Tucson, AZ 85721

## ABSTRACT

A method for joint source/channel rate allocation for transmission of JPEG2000 (J2K) codestreams over noisy channels is proposed. The rate allocation method uses the BER statistics of different channel codes and the rate-distortion characteristics of the source generated by the J2K encoder. The features of the J2K codestream are utilized to form a multi-layer codestream and unequal amount of protection is applied to each layer. The proposed method is capable of accomodating different channel coding strategies. Experimental results indicate that the presented method compares favorably with methods existing in the literature.

## 1. INTRODUCTION

The new generation of wavelet-based image codecs, such as SPIHT [1] and J2K [2], can deliver efficient compression. However, their bitstreams are very sensitive to errors. Thus, transmission of these bitstreams over noisy channels is challenging. In recent years, numerous works have focused on protecting these bitstreams from the effects of channel noise. These works can be classified into two general groups: equal error protection (EEP) and unequal error protection (UEP). In EEP, the entire bitstream receives the same amount of protection. The UEP schemes, however, apply different amounts of protection to different sections of the bitstream. Since the above mentioned image codecs can generate progressive bitstreams, the importance of the bits at different sections of the bitstream may not be equal. While an EEP scheme assigns the same amount of protection to every bit regardless of its importance, the UEP schemes are capable of reducing the protection of less important sections of the bitstream to provide stronger protection in more important sections. Thus, it is possible to achieve better performance with UEP than EEP.

In [3], rate-compatible punctured convolutional (RCPC) codes [4] are used to protect SPIHT bitstreams over binary symmetric channels (BSC). A channel code rate was carefully chosen for each channel such that it has an extremely low probability of error after channel decoding. Then the entire SPIHT bitstream is equally protected with the selected code. In [5], a UEP scheme for SPIHT that also uses RCPC codes was proposed. An iterative method was used to partition the bitstream into different sections, and to select a channel code rate for each section. This scheme provided an improvement around 0.3 dB over EEP. In [6], a UEP scheme with stronger Turbo codes was proposed to protect J2K bitstreams and the Viterbi algorithm (VA) was used to achieve optimum rate allocation.

In this paper, a method for joint source/channel rate allocation for transmission of J2K codestreams over noisy channels is proposed. The source and channel coding rates are jointly optimized to form a UEP J2K bitstream. By utilizing the features of J2K codestreams, a multi-layer source/channel encoded bitstream is formed at the desired total bit rate. The RCPC codes have been selected to illustrate the method proposed here due to their efficiency and low complexity. However, other channel codes can be easily adopted as well. Experimental results indicate that the proposed scheme compares favorably with others using RCPC channel codes.

The paper is organized as follows: In Section 2, a high level description of J2K is presented. In Section 3, the proposed rate allocation algorithm is developed. Section 4 presents experimental results and Section 5 provides conclusions.

## 2. OVERVIEW OF JPEG2000

JPEG2000 is the latest international standard for image compression. Besides providing state-of-the-art compression performance, it offers a number of functionalities that address the requirements of emerging image applications. Some basic concepts involved in the paper are briefly discussed in this section. For further details, the reader is referred to [7].

In J2K, an input image is first divided into non-overlapping rectangular tiles. If the image has multiple components, the samples of each component that fall into a particular tile are referred to as a *tile-component*. Each tile-component is then transformed using a wavelet transform and the wavelet subbands are partitioned into several different geometric structures. The smallest of such structures is a *codeblock*. Codeblocks are formed by partitioning the wavelet subbands. The codeblocks of particular resolutions

---

This work was supported in part by the National Science Foundation under Grant No. CCR-997310.

are grouped together to form *precincts*. Once the wavelet subbands are quantized, each codeblock is compressed individually using a bitplane coder. The bitplane coder makes three passes over each bitplane of a codeblock. These passes are referred to as *coding passes*. The compressed data from each codeblock can be regarded as an embedded bitstream. The J2K encoder computes and uses the rate-distortion information associated with each coding pass of every codeblock. The codestream is formed by including different numbers of coding passes from each codeblock, depending on a given criterion. Note that if a decoder detects an error within a coding pass, it will discard the current and all future coding passes belonging to that codeblock. In other words, the decoder will decode up to the last coding pass prior to the one that contains the error and stop decoding any further coding passes belonging to this particular codeblock.

For the purpose of forming the codestream, compressed data from each precinct are arranged to form *packets*. Each packet contains a header and a body. The packet header contains information about the contribution of each codeblock in the precinct to the packet. The body of the packet contains coding passes of codeblocks in this precinct. One packet from each precinct of each resolution of each tile-component form a *layer*. So conceptually, layer is a quality increment for an entire tile. Packets that belong to a particular tile are grouped together to form a *tile-stream*, and tile-streams are grouped together to form the J2K codestream. Similar to packets, tile-streams are comprised of a header and a body. There is also a main header at the beginning of the codestream of each image. The header information at various levels is crucial for correct decoding of the codestream.

J2K provides several error resilience tools to combat error propagation along the codestream and keep the synchronization between the encoder and the decoder. J2K provides a mechanism referred to as *packed packet headers*. Using this mechanism, it is possible to extract the packet headers from every packet, and store them within the main header. This can provide significant advantages for error resilience if the main header can be transmitted in an error-free fashion. Experiments show that error resilience tools provided within the standard can improve the performance, especially when they are combined with other techniques such as forward error coding (FEC).

## 3. ALGORITHM DESCRIPTION

The goal of joint source/channel optimization is to find a rate allocation scheme $V$ to minimize the expected distortion subject to a designated bit rate. From Section 2, in the case of no channel noise, it is equivalent to minimizing the distortions contributed from each codeblock while keeping the sum of their bitstream lengths within the designated rate:

$$\min \sum_b D_b \qquad \text{s.t.} \qquad \sum_b L_b \leq L_T \tag{1}$$

where $D_b$ and $L_b$ are the distortion and bitstream length of codeblock $b$ respectively; $L_T$ is the designated file size.

When taking the effects of channel coding and noise into account, for each codeblock $b$, its expected distortion becomes:

$$E[D_b(V_b)] = D_{b,0} - E[\Delta D_b(V_b)] \tag{2}$$

where $D_{b,0}$ is the initial distortion, $V_b$ is the rate allocation scheme for codeblock $b$. $E[\Delta D_b(V_b)]$ is the expected distortion reduction when rate allocation scheme $V_b$ is employed, resulting in a length $L_b(V_b)$ codeblock bitstream.

With Eq. (2), in the case of a noisy channel, Eq. (1) can be written as

$$\min \sum_b (D_{b,0} - E[\Delta D_b(V_b)]) \quad \text{s.t.} \quad \sum_b L_b(V_b) \leq L_T$$

or equivalently,

$$\min \sum_b -E[\Delta D_b(V_b)] \qquad \text{s.t.} \qquad \sum_b L_b(V_b) \leq L_T \tag{3}$$

Adapting the results from [8], we can solve the unconstrained problem

$$\min\{\sum_b -E[\Delta D_b(V_b)] + \lambda \sum_b L_b(V_b)\} \tag{4}$$

by minimizing each term independently. Sweeping $\lambda$ over the range of zero to infinity, sets of $\{V_b\}$ and $\{\sum_b L_b\}$ can be created. If a $(\sum_b L_b)$ happens to equal $L_T$, then a desired solution has been found.

Minimizing each individual term in Eq. (4) corresponds to an optimization task at the coding pass level. For a codeblock $b$, denote $N_c$ as the total number of coding passes it has. For each coding pass $i$, there is an associated distortion reduction $\Delta d_i$ with length $l_i$ bytes, where $i \in [1, N_c]$. Without any channel coding,

$$\Delta D_b = \sum_{i=1}^{N_c} \Delta d_i \qquad L_b = \sum_{i=1}^{N_c} l_i$$

From Eq. (4), it is desired to find a rate allocation scheme $V_b$ which can minimize:

$$-E[\Delta D_b(V_b)] + \lambda L_b(V_b) \tag{5}$$

Denote $r_i$ as the channel coding rate for coding pass $i$ ($0 \leq r_i \leq 1$) and $P(r_i, \frac{l_i}{r_i})$ as the probability that there are one or more uncorrected error in coding pass $i$ if channel coding rate $r_i$ is employed. Finally, let $N_c'$ be the number of coding passes included by $V_b$. Since a J2K decoder can decode all correct coding passes prior to the one containing the first bit error in a codeblock, the expected distortion reduction for $N_c'$ coding passes using channel code rates $r_1, r_2, \ldots, r_{N_c'}$ is

$$E[\Delta d] = \sum_{j=1}^{N_c'-1} (\sum_{k=1}^{j} d_k)(\prod_{k=1}^{j}[1 - P(r_k, \frac{l_k}{r_k})])P(r_{j+1}, \frac{l_{j+1}}{r_{j+1}}) + (\sum_{k=1}^{N_c'} d_k)(\prod_{k=1}^{N_c'}[1 - P(r_k, \frac{l_k}{r_k})]). \qquad (6)$$

So with Eq. (6), given a $\lambda$, Eq. (5) can be evaluated for any rate allocation scheme $V_b = (r_1, r_2, \ldots, r_{N_c'})$. Here, $r_i$ denotes the channel coding rates applied to coding pass $i$, $N_c'$ denotes the last coding pass included from codeblock $b$ ($N_c' \leq N_c$). It is worth noting that the rate allocation must jointly optimize $N_c'$ and the coding rates $r_1, r_2, \ldots, r_{N_c'}$.

One possible way to find the optimal $V_b$ is by exhaustive search. Suppose there are $k$ channel coding rates available. If $N_c'$ ($0 \leq N_c' \leq N_c$) coding passes are included, there are at most $k^{N_c'}$ possible coding rate combinations, each forming a permissible code-rate vector. So for an $N_c$-coding-pass codeblock $b$, there are at most $\sum_{N_c'=1}^{N_c} k^{N_c'}$ permissible code-rate vectors constituting its search space.

Generally, without imposing any constraint, this space is too large to search. Due to the fact that the significance of the bits from a codeblock reduce along the bitstream, in [5], it has been discovered that the optimal protection level reduces along the bitstream. That is, $r_i \leq r_{i+1}$ ($1 \leq i \leq N_c'$). Also from the results in [5, 6], at most $k \leq 3$ protection levels are enough for binary symmetric channels (BSC) with error probabilities $\epsilon \leq 10^{-1}$. With these two conditions, the above search space is tremendously reduced, allowing a simple search scheme.

Based upon the above discussion, the proposed rate allocation method can be divided into two levels. At the lower level, it deals with coding passes within each codeblock. Given a $\lambda \geq 0$, by searching the permissible code-rate vector space using Eq. (5) and Eq. (6), an optimal code-rate vector is found yielding a length $L_b$ for each codeblock $b$. At the higher level, it sweeps $\lambda$ in a reasonable range, collecting all $L_b(\lambda)$s and summing them up. When the sum is equal to a desired file length, then the corresponding optimal rate allocation scheme is found.

## 4. EXPERIMENTAL RESULTS

For the results presented in this section, we choose the binary symmetric channel (BSC) with $\epsilon = 10^{-2}$ as our experimental channel. As mentioned earlier, we use RCPC as our channel code. Thus, our channel encoder/decoder structure is the same as the one used in [3]. In our experiments, we have found that the rate $\frac{2}{3}$ RCPC code is strong enough to yield an extremely small probability of error at the decoder for this channel, and we have selected RCPC code rates ($\frac{8}{9}, \frac{4}{5}, \frac{2}{3}$) from [4] to be used in our experiments. We also allow the case of no channel coding. Thus, we have simulated the error rate statistics for 4 code rates of ($1, \frac{8}{9}, \frac{4}{5}, \frac{2}{3}$) and tabulated them for further computations.

In our experiments using the proposed rate allocation method, the rate $\frac{4}{5}$ and $\frac{8}{9}$ codes were always preferred over the other two codes. In other words, a certain number of coding passes of each codeblock were selected to be protected using the rate $\frac{4}{5}$ code, and other coding passes were selected to be protected by the rate $\frac{8}{9}$ code.

This situation can be easily addressed using the layer functionality offered by J2K. By providing the J2K encoder with the numbers of coding passes of each codeblock in each layer, a 2-layer bitstream can be generated. As discussed in section 2, packed packet headers can be used to store the packet headers within the main header which is then strongly protected using the rate $\frac{2}{3}$ code.

The proposed rate allocation method is always able to achieve a bit rate that is very close to the target rate in our simulations. It is possible to achieve the target rate exactly by some simple adjustments. If the target rate is above the achieved rate, it is possible to protect more of the bitstream with the stronger code. If the target rate is below the achieved rate, a small number of less significant coding passes can be dropped.

The Lenna and Barbara (512x512) images were used as our test images. Simulations were done at total bitrates of 0.10, 0.25, 0.50, 0.75, 1.00 (bpp) with 1000 trials for each case. Our results were compared with the EEP SPIHT scheme [3] and the EEP J2K scheme using the same channel codes. These results are presented in Tables 1 and 2.

It should be noted that the average noisy PSNRs obtained using both the EEP SPIHT and the EEP J2K schemes are very close to those of the noise-free cases. This is due to the strong protection provided by the channel code used in these cases. The source coding efficiency of these coders are comparable for the Lenna image. For the Barbara image, the J2K encoder performs significantly better with increasing rate. The proposed UEP method provides an average PSNR improvement of 0.26 and 0.47 dB for the Lenna and Barbara images, respectively. These results are similar to those obtained in [5] where the authors mention that their UEP scheme provided an average PSNR improvement of 0.3 dB on the Lenna image compared to the EEP scheme. It should be noted however that the UEP gain can be as large as 0.72 dB in some cases. This gain is usually smaller at low rates and increases with rate. This is expected, since at lower rates the range of the rate-distortion slopes of the coding passes that are included in the bitstream is smaller compared to the higher rates. Thus, the EEP scheme does not yield a large over-protected section that can be exploited using UEP.

The results in Tables 1 and 2 also include the percentage of simulation cases where the PSNR achieved by the UEP J2K is greater than those of SPIHT EEP and J2K EEP. It can be observed that, even when the average PSNRs of UEP and EEP schemes are close, the UEP scheme provides results with larger PSNR values in most cases.

## 5. CONCLUSIONS

In this paper, a joint source/channel rate allocation method that provides unequal error protection for J2K encoded bitstreams is proposed. By optimizing the rate allocation using the bit error rate statistics of the channel codes and the rate-distortion information provided by the encoder, the overall performance of the system is improved. The method is applicable when different types of channel codes are used.

## 6. REFERENCES

[1] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. on Cir. and Syst. for Video Tech.*, vol. 6, pp. 243–250, June 1996.

[2] *JPEG 2000 Part I Final Draft International Standard*, ISO/IEC JTC 1/SC 29/WG1, Doc. No. N1855, 2000.

[3] P. G. Sherwood and K. Zeger, "Progressive image coding for noisy channels," *IEEE Signal Processing Letters*, vol. 4, pp. 189–191, July 1997.

[4] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. on Comm.*, vol. 36, pp. 389–400, Apr. 1988.

[5] M. Zhao and A. N. Akansu, "Optimization of dynamic UEP schemes for embedded image sources in noisy channels," in *Proc. of Int. Conf. on Image Processing*, 2000, vol. 1, pp. 383–386.

[6] B. A. Banister, B. Belzer, and T. R. Fischer, "Robust image transmission using JPEG2000 and turbo-codes," in *Proc. of Int. Conf. on Image Processing*, 2000, vol. 1, pp. 375–378.

[7] D. S. Taubman and M. W. Marcellin, *JPEG2000: Image Compression Fundamentals, Practice and Standards*, Kluwer Academic Publishers, Massachusetts, 2002.

[8] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. on Acous. and Sig. Proc.*, vol. 36, pp. 1445–1453, Sept. 1988.

| Bit Rate (bpp) | 0.1 | 0.25 | 0.50 | 0.75 | 1.00 |
|---|---|---|---|---|---|
| $\frac{2}{3}, \frac{4}{5}, \frac{8}{9}$ rates bitstream length ratio | 1: 1.33: 4.17 | 1: 8.94: 4.24 | 1: 13.88: 5.46 | 1: 19.90: 4.03 | 1: 23.22: 6 |
| UEP J2K source rate / noise-free PSNR | 0.07/ 28.45 | 0.18/ 32.37 | 0.37/ 35.70 | 0.55/ 37.53 | 0.74/ 38.84 |
| EEP SPIHT Source Rate / Av. noisy PSNR | 0.06/ 28.21 | 0.15/ 31.92 | 0.30/ 35.00 | 0.45/ 36.79 | 0.60/ 38.07 |
| EEP J2K Source Rate / Av. noisy PSNR | 0.06/ 27.93 | 0.15/ 31.75 | 0.30/ 34.92 | 0.45/ 36.78 | 0.60/ 38.06 |
| UEP J2K Av. noisy PSNR | 27.87 | 32.00 | 35.32 | 37.17 | 38.39 |
| Percentage PSNR(UEP J2K) ≥PSNR(SPIHT) | 46.80 | 80.40 | 94.20 | 94.10 | 94.00 |
| Percentage PSNR(UEP J2K) ≥PSNR(EEP J2K) | 58.80 | 89.00 | 94.50 | 94.40 | 94.10 |

**Table 1**. Lenna 512 x 512

| Bit Rate (bpp) | 0.1 | 0.25 | 0.50 | 0.75 | 1.00 |
|---|---|---|---|---|---|
| $\frac{2}{3}, \frac{4}{5}, \frac{8}{9}$ rates bitstream length ratio | 1: 3.33: 4.17 | 1: 8.38: 5.86 | 1: 11.52: 4.45 | 1: 25: 3.79 | 1: 27.79: 4.10 |
| UEP J2K source rate / noise-free PSNR | 0.08/ 23.93 | 0.19/ 26.75 | 0.37/ 30.17 | 0.55/ 32.66 | 0.73/ 34.52 |
| EEP SPIHT Source Rate / Av. noisy PSNR | 0.06/ 23.28 | 0.15/ 25.68 | 0.30/ 28.57 | 0.45/ 30.77 | 0.60/ 32.57 |
| EEP J2K Source Rate / Av. noisy PSNR | 0.06/ 23.38 | 0.15/ 26.04 | 0.30/ 29.29 | 0.45/ 31.57 | 0.60/ 33.39 |
| UEP J2K Av. noisy PSNR | 23.38 | 26.45 | 29.84 | 32.25 | 34.11 |
| Percentage PSNR(UEP J2K) ≥PSNR(SPIHT) | 70.20 | 98.10 | 97.70 | 96.60 | 95.70 |
| Percentage PSNR(UEP J2K) ≥PSNR(EEP J2K) | 66.10 | 94.10 | 95.30 | 94.70 | 93.90 |

**Table 2**. Barbara 512 x 512