Joint Design of Fixed-Rate Source Codes and UEP Channel Codes for Fading Channels

Hai-Xin Tie[*], Andrea Goldsmith, and Michelle Effros
California Institute of Technology
{haixin, andrea, effros}@systems.caltech.edu

Abstract

We study an iterative design algorithm that jointly optimizes source and channel codes for image transmission over Rayleigh fading channels. The joint design combines channel-optimized vector quantization (COVQ) for the source code with a family of rate-compatible punctured convolutional codes for the unequal-error-protection (UEP) channel code. We consider both hard-decision and soft-decision decoding for these channel codes. Our objective is to minimize the average end-to-end distortion, averaged over both the source and channel statistics. For a given channel SNR and transmission rate, our joint source and channel code design achieves an optimal allocation of bits between the source and channel coders. This optimal allocation can reduce distortion by up to 7 dB over suboptimal allocations for the source data set considered. Our joint iterative design exhibits a coding gain over COVQ of 1.5 dB for hard-decision decoding and 3 dB for soft-decision decoding.

1 Introduction

Shannon's classical separation theorem [1] states that for stationary and ergodic sources and channels, the design of source and channel codes can be done separately with no loss in optimality. However, this result holds in the limit of infinite source code dimension and infinite channel code block length. For practical systems with finite dimension and block length constraints, a joint source and channel code may reduce distortion, as well as complexity and delay. Indeed, recent work on source and channel coding for the additive white Gaussian noise (AWGN) channel and the Rayleigh fading channel have demonstrated significant performance gains through a joint design [2, 3, 4, and the references therein].

In this work we focus on a joint source and channel code design for fading channels, where the channel amplitude varies with time. Specifically, we propose an iterative code design which jointly optimizes the source and channel codes to minimize end-to-end distortion over fading channels. Our joint code design uses COVQ for the source code and RCPC coding for the channel code. This code design is an extension of the joint design proposed in [2]: in that work the design is optimized for an AWGN channel whereas now we optimize the design for a fading channel.

The COVQ is a vector quantizer (VQ) that has been optimized for a given set of crossover probabilities of the source codeword indices [5]. The codeword indices are generally mapped to binary strings, and the crossover probabilities are then functions of the channel's bit error probability. The COVQ design algorithm optimizes its codewords relative to this error probability: thus, inherent to the COVQ design is a block channel code with hard-decision decoding. In addition to this inherent code, our joint code design uses an additional channel code to reduce the channel's bit-error-probability at the expense of a reduced source rate. Since bit errors in different locations of a binary string cause different amounts of distortion, it is desirable for this additional channel code to provide different levels of error protection for different bits. This is done using an UEP channel code matched to the COVQ design to minimize distortion. UEP channel coding can be implemented using multiplexed convolutional or block codes, trellis codes, or rate-compatible punctured convolutional (RCPC) codes [6]. We use RCPC codes for our channel code, although our design technique is easily extended to other forms of UEP channel coding. We examine both hard-decision (HD) and soft-decision (SD) decoding for the RCPC code. In general, HD decoding has poorer performance in exchange for reduced complexity.

The designs of the COVQ and RCPC codes are not independent. The optimal COVQ is the COVQ matched to the index crossover probabilities determined by the RCPC channel code and the channel. Likewise, the optimal RCPC code is the RCPC code that minimizes the expected distortion of the COVQ. The design algorithm proposed in this work achieves a joint optimization of these source and channel codes using an iterative descent technique reminiscent of the generalized Lloyd algorithm [7]. Our joint design also optimizes the bit allocation between the source and channel codes for a given channel SNR. For the source data set considered, we will see that a suboptimal allocation can increase distortion by up to 7 dB. We will also show that our joint design algorithm exhibits a performance gain relative to standard COVQ (without additional channel coding) of 1.5 dB for HD decoding and 3 dB for SD decoding.

The remainder of this paper is organized as follows. The end-to-end system model is presented in Section 2. The

[*] Supported by NSF CAREER award NCR-9501452 and by a Caltech Summer Undergraduate Research Fellowship (SURF).

0-7803-5148-7/98/$10.00©1998 IEEE

COVQ, RCPC channel codes, and joint iterative code design are described in Section 3. Experimental results appear in Section 4. Soft-decision COVQ is discussed in Section 5. Section 6 summarizes our results and conclusions.

2 System Model

A block diagram of the end-to-end communication system is shown in Figure 1. We assume a discrete-time, real-valued, stationary source. The source encoder maps the set of possible k-dimensional source vectors $\underline{x}$ into a set of binary strings $\underline{v}$. An example of a source vector would be a block of k pixels from an image. The binary strings can be fixed-length or variable-length, corresponding to a fixed-rate or variable-rate source code. Due to finite bandwidth constraints, the source encoder typically introduces some distortion. While variable-rate source codes achieve better compression for a given distortion and source vector dimension, they typically perform poorly when channel errors are introduced. Thus we consider only fixed-rate vector quantizers, which produce fixed-length binary strings of length kR_s for each source vector $\underline{x}$. The number of bits per source symbol, R_s, is a parameter of the source code design. The channel encoder operates on the vector $\underline{v}$ of kR_s bits to obtain $k(R_s+R_c)$ coded bits $\underline{w}$. These coded bits (channel symbols) are interleaved, modulated using binary phase shift keying (BPSK) and then transmitted over the channel at a rate of one channel symbol per T seconds, where T is the channel symbol time.

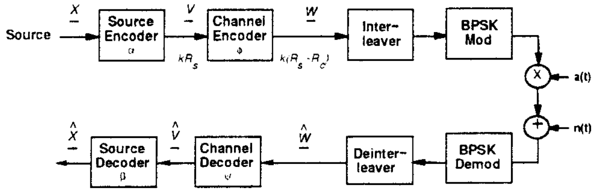

Figure 1: System Model.

The channel has amplitude fading $a(t)$ and AWGN $n(t)$. Amplitude fading arises in wireless channels due to multipath. Multipath components of the transmitted signal reflect off surrounding objects between the transmitter and receiver, and arrive at the receiver delayed in time and shifted in phase with respect to the direct-path signal component [8]. The phase shifts and time delays cause constructive and destructive interference between the multipath and direct-path signal components, causing large variations in the received signal amplitude. We assume that the delay spread of the multipath components is small relative to T, so there is no intersymbol interference from the multipath, only the time-varying amplitude $a(t)$. In our numerical results we assume the amplitude variations follow a Rayleigh distribution: $p(a) = 2a\exp[-a^2], a > 0$. However, our joint code design can be applied to any fading distribution. The interleaver/deinterleaver pair are assumed to have infinite depth so that the amplitude values appear i.i.d. to the channel decoder.

At the receiver, the signal is first demodulated and then deinterleaved, which yields the noisy channel symbols $\hat{\underline{w}}$. These channel symbols are passed through the channel decoder, which may correct some (or all) of the channel errors to obtain a noisy reproduction $\hat{\underline{v}}$ of the original binary string $\underline{v}$. Finally, $\hat{\underline{v}}$ is passed through the source decoder to obtain a noisy reproduction $\hat{\underline{x}}$ of the original source vector $\underline{x}$.

The total end-to-end distortion of the system is the mean-squared-error between the source vector $\underline{x}$ and source reproduction vector $\hat{\underline{x}}$: $d(\underline{x}, \hat{\underline{x}}) =\| \underline{x} - \hat{\underline{x}} \|^2$. For a given R_s, the total distortion decreases as R_c increases. Similarly, for a given R_c, the total distortion decreases as R_s increases. For $R = R_s + R_c$ fixed, there is an optimal way to divide the transmission rate R between the source and channel rates R_s and R_c to minimize total distortion. This optimal allocation depends on the SNR per channel symbol E_s/N_0[1], the fading distribution, the channel code, and the source statistics. For example, channel coding is less important on high-SNR channels, so the R_s value corresponding to the optimal bit allocation will generally increase as a function of E_s/N_0. For our joint code design we consider all (R_s, R_c) pairs such that $R_s + R_c = R$ and choose the pair and corresponding joint code with minimal distortion. In the joint code design for AWGN channels the optimal bit allocation turned out to be the most important aspect of the joint code design [2], and we expect the same to be true for fading channels.

3 Joint Iterative Code Design

The goal of our joint code design is to minimize the expected distortion $D = E[d(\underline{x}, \hat{\underline{x}})]$ of the COVQ and RCPC codes, where the expectation is with respect to the source data training set and the channel statistics. We use an iterative design technique to obtain this minimization. Specifically, for a given R_s value we alternately optimize the COVQ for a given RCPC code and then the RCPC code for the new COVQ. The flow chart for our design algorithm is shown in Figure 2. This iterative approach is guaranteed to converge to a locally optimal solution since each design step gives a global optimum. The design process is repeated for each R_s value in the range $0 \leq R_s \leq R$, where R is the channel transmission rate per source symbol. For $R_s = 0$, the source is represented by a single codeword, so there is no channel distortion and the total distortion corresponds to the source variance. For $R_s = R$, all redundant bits are allocated to the source code, so there is no channel coding. This corresponds to standard COVQ [5]. Intermediate values of R_s correspond to a redundancy tradeoff in the source and channel codes. The joint code design corresponding to the (R_s, R_c) pair with minimal distortion is chosen as the final joint code design. We now describe each of the algorithm steps in Figure 2.

3.1 Step 0: Initialization

We initialize the system such that $\Pr(\hat{\underline{v}} = \underline{v} | \underline{v}) = 1$. This initialization is equivalent to assuming a noiseless channel,

[1] The $E_s/N_0 = PT/N_0$, where P is the received signal power, N_0 the spectral noise density, and T the channel symbol time.

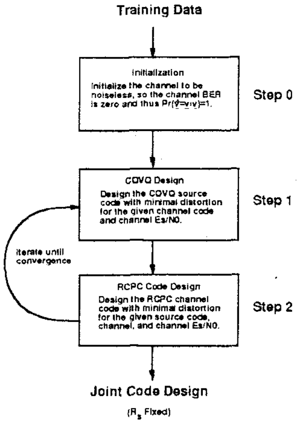
Training Data
Initialization
Initialize the channel to be noiseless, so the channel BER is zero and thus Pr(v̂=v|v)=1.
Step 0
COVQ Design
Design the COVQ source code with minimal distortion for the given channel code and channel Es/N0.
Step 1
Iterate until convergence
RCPC Code Design
Design the RCPC channel code with minimal distortion for the given source code, channel, and channel Es/N0.
Step 2
Joint Code Design
(R_s Fixed)

Figure 2: Iterative Code Design Process.

so that the bit-error-probability for any RCPC channel code is zero.

3.2 Step 1: COVQ Design

In Step 1 of the design algorithm, the COVQ (α,β) is optimized for the bit-error-probability of the given RCPC channel code to minimize the expected distortion $D = E[d(\underline{X},\hat{\underline{X}})]$ between a random source input vector $\underline{X}$ and its reproduction $\hat{\underline{X}}$ at the receiver. The bit-error-probabilities result either from the initialization of Step 0 or from the channel E_s/N_0, the channel fading statistics, and the RCPC channel code (ϕ,ψ) found in Step 2. The optimal α and β are obtained through the COVQ design algorithm described in [5]. For any fixed RCPC channel code, this design technique is itself an iterative algorithm which successively redesigns α for a given β and β for a given α.

Specifically, for a fixed source decoder and channel code the optimal source encoder α^* that minimizes for each $\underline{x}$ the expected distortion between $\underline{x}$ and its reproduction $\hat{\underline{x}}$ at the source decoder is given by $\alpha^*(\underline{x}) = \arg\min_{\underline{v}} \sum_{\hat{\underline{v}}\in\{0,1\}^{kR_s}} \Pr(\hat{\underline{v}} \mid \underline{v}) d(\underline{x},\beta(\hat{\underline{v}}))$, where $Pr(\hat{\underline{v}} \mid \underline{v})$ is the probability that channel errors cause the binary string $\underline{v}$ output by the source encoder to be received as $\hat{\underline{v}}$ by the source decoder. We call $\{\Pr(\hat{\underline{v}} \mid \underline{v})\}$ the index crossover probabilities, which are functions of the RCPC channel code and the channel, as we describe in more detail in Section 3.3 below. Associated with the source encoder $\alpha^*(\underline{x})$ are a set of encoding regions $\{S_{\underline{v}} : \underline{v} \in \{0,1\}^{kR_s}\}$, where $S_{\underline{v}} = \{\underline{x} : \alpha(\underline{x}) = \underline{v}\}$. Notice that the optimal source encoder is no longer the traditional VQ nearest neighbor encoder: the binary string to which α^* maps a given $\underline{x}$ vector may not be the one with the closest reproduction $\beta(\cdot)$, due to the fact that the binary string may be corrupted in transmission.

For a given source encoder α and crossover probabilities $\{\Pr(\hat{\underline{v}} \mid \underline{v})\}$ of a fixed RCPC channel code, the optimal source decoder β^* must have codewords that satisfy $\beta^*(\hat{\underline{v}}) = \arg\min_{\hat{\underline{x}}} E[\Pr(\hat{\underline{v}} \mid \alpha(\underline{X}))d(\underline{X},\hat{\underline{x}})]$ for each $\hat{\underline{v}} \in \{0,1\}^{kR_s}$. Just as the optimal encoding regions have changed due to the channel errors, the optimal codewords of the decoder design are no longer at the centroids of their decoding cells. Instead, each codeword is a weighted average of the centroids of all of the encoding cells. Specifically, in finding the codeword $\beta(\hat{\underline{v}})$ associated with the codeword index $\hat{\underline{v}}$, we weight the centroid of the encoding region $S_{\underline{v}}$ by the probability that the received vector $\hat{\underline{v}}$ came from transmitting $\underline{v}$ through the channel.

The encoding region centroids will be extremely useful to us in optimizing our joint code. For each $\underline{v} \in \{0,1\}^{kR_s}$ define the centroid $\gamma(\underline{v})$ of $S_{\underline{v}}$ by $\gamma(\underline{v}) = E[\underline{X} \mid \underline{x} \in S_{\underline{v}}]$. Then it can be shown [2] that the expected distortion associated with a given source code (α,β) can be written as

$$E[d(\underline{X},\hat{\underline{X}})] = E_x\left[||\underline{X} - \gamma(\alpha(\underline{X}))||^2\right] + E_{\underline{v},\hat{\underline{v}}}\left[||\gamma(\underline{v}) - \beta(\hat{\underline{v}})||^2\right], \tag{1}$$

where $E_{\underline{v},\hat{\underline{v}}}$ denotes the expectation with respect to the joint distribution $\Pr(\underline{v},\hat{\underline{v}}) = \Pr(\underline{v})\Pr(\hat{\underline{v}}|\underline{v})$. In Section 3.3 we will see that Equation (1) is extremely useful in designing the optimal RCPC code, since it greatly reduces the computational expense of evaluating the distortion for each possible channel code. We will also see that this equation allows the joint design algorithm to be easily modified for different channel models and channel code designs.

The COVQ design algorithm for a given RCPC channel code is an iterative descent technique analogous to the generalized Lloyd algorithm. Each iteration proceeds through the following two steps: the encoder is optimized for the given decoder and RCPC channel code, where the optimal decoder is given by α^* above, then the decoder is optimized for the given encoder and RCPC channel code, where the optimal decoder is given by β^* above. Notice that while the algorithm as a whole guarantees only a locally optimal (α,β), each of these two steps finds a globally optimal solution: in the first step we find a globally optimal encoder for the given decoder and RCPC channel code, and in the second step we find a globally optimal decoder for the given encoder and RCPC channel code.

3.3 Step 2: RCPC Design

In Step 2 of the algorithm, the RCPC code with minimal distortion for the COVQ designed in Step 1 is found. We use the set of channel code rates from Table 1 in [6] to obtain our RCPC code candidates. Each candidate RCPC code provides a set of kR_s bit-error-probabilities for the kR_s bits in $\underline{v}$. We represent each candidate code by a vector of length kR_s, where the jth vector element corresponds to the channel code rate applied to the jth bit in $\underline{v}$. For example, for $kR_s = 4$, the RCPC code $(1/2, 1/4, 1, 1)$ applies a rate $1/2$ convolutional code to the first bit in $\underline{v}$, a rate $1/4$ convolutional code to the second bit in $\underline{v}$, and no channel code to the third and fourth bits in $\underline{v}$. Associated with the convolutional code applied to the jth bit is a bit-error-probability P_j which depends on the channel E_s/N_0, the channel fading statistics, and the decoder assumptions (hard or soft decision, channel side information, etc). The bit-error-probability versus E_s/N_0 for each element of an RCPC channel code vector under our BPSK modulation and Rayleigh fading assumptions was derived by Hagenauer: the error probability for HD decoding and no channel side information is given by [6, Eqn. 11] and the error probability for soft-decision decoding with receiver knowledge of the fading amplitude $a(t)$ is given by [6, Eqn. 13]. We determine the performance of our joint source and channel code design for both types of channel decoders, which we denote by HD and SD respectively.

The minimal-distortion RCPC code is obtained as follows. For a given COVQ and corresponding R_s value, we search over all RCPC code vectors that satisfy the transmission rate constraint $R = R_s + R_c$. For a given R_s, an RCPC code vector $(c_1, \ldots, c_{kR_s})$ satisfies this constraint if $\sum_{i=1}^{kR_s}(c_i)^{-1} \leq kR$. We call an RCPC code that satisfies this constraint an eligible channel code. The search for the optimal RCPC channel code must consider all eligible channel codes in the set of $(14)^{R_s}$ possible channel codes.

The optimal RCPC code is the code with the error protection levels that minimize the expected distortion (1). The first term in (1) depends only on the source code and source training data set. Thus, while this term is expensive to calculate, it does not change as a function of the RCPC channel code. Therefore, our search for the optimal RCPC channel code need only find the RCPC code that minimizes the second term of (1):

$$E_{\underline{v},\hat{\underline{v}}}\left[||\gamma(\underline{v}) - \beta(\hat{\underline{v}})||^2\right]. \tag{2}$$

This term is a function of the index crossover probabilities $\{\Pr(\hat{\underline{v}} \mid \underline{v})\} = \prod_{j=1}^{kR_s}[P_j 1(\hat{v}_j \neq v_j) + (1 - P_j)1(\hat{v}_j = v_j)]$, where $1(\cdot)$ denotes the indicator function, $\hat{v}_j$ and v_j denote the jth elements of the vectors $\hat{\underline{v}}$ and $\underline{v}$ respectively, and P_j denotes the bit-error-probability of the jth bit in $\underline{v}$.

Since (2) is not a function of the training set and the $\{P_j\}$ values are typically inexpensive to calculate, we can quickly search through all eligible channel codes to determine which one minimizes (2) and thereby minimizes the total distortion for the given COVQ. We can also easily adapt our joint code design for different channel models and UEP channel codes merely by calculating the corresponding bit-error-probabilities $\{P_j\}$ and using these values to compute (2). After computing (2) for all eligible channel codes, we select the RCPC code with minimal distortion. This code corresponds to the initial channel code (ϕ, ψ) in Figure 1.

3.4 Design Iteration

At the conclusion of Step 2 a new channel code (ϕ, ψ) has been designed for the COVQ obtained in Step 1. The iterative design process then returns to Step 1 to determine the COVQ (α, β) for this new RCPC channel code (ϕ, ψ). Once a new COVQ is obtained, this source code is passed to Step 2 of the algorithm to obtain a new RCPC channel code. Successive application of Steps 1 and 2 results in a sequence of source codes $\{(\alpha, \beta)\}$ and corresponding channel codes $\{(\phi, \psi)\}$ for which the average distortions (1) form a positive nonincreasing sequence which has to converge. At convergence, the source code $(\alpha, \beta)^*_{R_s}$ and channel code $(\phi, \psi)^*_{R_s}$ with minimal distortion for the given R_s value are obtained. The design process is repeated for each R_s value $0 \leq R_s \leq R$, and the source code $(\alpha, \beta)^*$ and channel code $(\phi, \psi)^*$ corresponding to the R_s value with minimal distortion comprise the final joint code.

4 Experimental Results

The joint code design process described in Section 3 was implemented in C^{++} and run for a range of channel E_s/N_0 values. Although the iterative design process is computationally complex (approximately twenty iterations are required for convergence), it is done off-line. Thus, the code design complexity does not impact the system operation. Our experimental results are computed for a test data set of 5 magnetic resonance images (MRIs) applied to the joint code designed from a training data set of 20 other MRIs. We define the ratio of signal power to quantization and channel noise power (SQCNR) as SQCNR (dB) $= 10\log(\sigma^2/D)$, where D is the distortion of the joint code averaged over the test data set and the channel statistics, and σ^2 is the distortion of a rate zero ($R_s = 0$) VQ averaged over the test data set.

In Figure 3 we show a plot of SQCNR for our joint code design as a function of R_s for a channel $E_s/N_0 = 10$ dB. We consider $0 \leq R_s \leq 2$ bits per pixel (bpp) and both HD and SD decoding in the channel code. For typical image compression algorithms R_s ranges from .5 to 1.5 bpp. For this range of R_s we see from Figure 3 that improper choice of the bit allocation can reduce the SQCNR by more than 7 dB for SD decoding and by more than 5.5 dB for HD decoding. This SQCNR penalty is larger than the penalty in AWGN for improper choice of the bit allocation [2]. Thus we see that as for AWGN channels, the optimal bit allocation is the most important aspect of a joint source and channel code design for fading channels, yet it is precisely this step that is ignored in most previous joint code designs. The performance of COVQ without additional channel coding (standard COVQ) is indicated by the SQCNR value for $R_s = 2$. We see from Figure 3 that our joint code design exhibits a performance gain relative to standard COVQ of more than 3 dB for SD decoding and more than 1.5 dB for HD decoding at $E_s/N_o =$ 10 dB. We expect even larger gains at other E_s/N_o values.

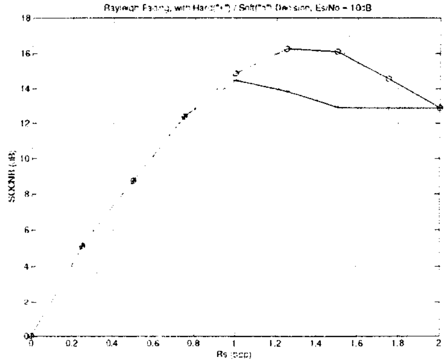

Figure 3: Joint Code SQCNR versus R_s at $E_s/N_0 = 10$ dB.

The distortion D corresponding to the SQCNR in Figure 3 is shown by the solid lines in Figure 4. The dashed lines in Figure 4 show the distortion contribution of the source code alone, based on averaging the distortion of our joint source and channel code over the test data set for a noiseless channel. Note that almost all the distortion at the optimal bit allocation (R_s, R_c) is contributed by the source code. A similar phenomenon was observed for the joint code design applied to an AWGN channel, although the source distortion contribution is even more dramatic for the fading channel. This indicates that the optimal source encoder in our joint code design has codewords so close together that channel errors have little impact on the average distortion.

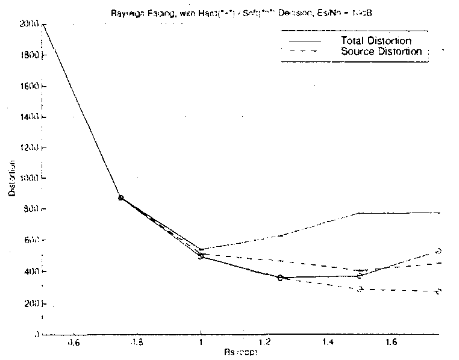

Figure 4: Distortion versus R_s at $E_s/N_0 = 10$ dB.

5 Soft-Decision COVQ

Our experimental results in the previous section indicate that our joint code design with either SD or HD decoding has better performance than COVQ. That is because the RCPC channel code in our joint code design is a more effective channel code than the hard-decision block channel code inherent to the COVQ design. There has been some recent work on modifying COVQ so that its inherent channel code uses soft-decision decoding. This design algorithm is called Soft-Decision COVQ (SD-COVQ).

SD-COVQ is a COVQ where the encoder and decoder are optimized relative to the analog channel output. That is, hard decisions prior to source decoding are eliminated. SD-COVQ has been applied to AWGN channels in [4] and to Rayleigh fading channels in [4, 9]. An alternative to SD-COVQ with somewhat reduced complexity is quantized COVQ [3]. With this technique the analog channel output is first passed through a q bit quantizer before decoding. Thus, an input source codeword of n bits, after passing through the channel and the quantizer, is represented by qn bits at the input to the decoder. Standard COVQ is applied to the joint design of the encoder/decoder pair for the n bit source encoder and qn bit source decoder. SD-COVQ corresponds to quantized COVQ with $q = \infty$. Quantized COVQ can achieve performance only slightly inferior to that of SD-COVQ with the advantage of a reduced decoder complexity and the disadvantage of a higher memory requirement. We are currently investigating the performance of SD-COVQ compared to that of our joint design algorithm. We expect to obtain better performance for our joint code design with SD decoding, since the SD-COVQ channel code is constrained by the source encoder vector dimension whereas the RCPC channel code is not. However, SD-COVQ may outperform our joint code design with HD decoding.

6 Summary

We have described an iterative procedure for jointly optimizing a COVQ source code and an RCPC channel code for Rayleigh fading channels. The procedure optimizes the bit allocation and the joint code design to minimize end-to-end distortion. Although the computational complexity of the code design is quite high, it is done off-line, and does not impact system operation. The most important design aspect of our joint code design is the optimal bit allocation between the source and channel coders. This optimal allocation depends on the source statistics, the channel statistics, the channel code, and the channel E_s/N_0. Our experiments indicate that a suboptimal choice of bit allocation decreases SQCNR by more than 7 dB. Our experiments also show that our joint code design outperform COVQ by up to 3 dB.

References

[1] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE Natl. Conv. Rec, Part 4*, pp. 142–163, 1959.

[2] A. Goldsmith and M. Effros, "Joint design of fixed-rate source codes and multiresolution channel codes." *IEEE Trans. Commun.*, Vol. 46, No. 10, pp. 1301–1312, Oct. 1998.

[3] F.I. Alajaji and N.C. Phamdo, "Soft-decision COVQ for Rayleigh-fading channels," *IEEE Commun. Let.*, Vol. 2, No. 6, pp. 162–164, June 1998.

[4] M. Skoglund, "On soft decoding and robust vector quantization," Ph.D. thesis, Chalmers Univ. of Technol., Goteborg, Sweden, Tech. Rep. 302, March 1997.

[5] N. Farvardin and V. Vaishampayan, "On the performance and complexity of channel-optimized vector quantizers," *IEEE Trans. Inform. Theory*, Vol. 37, pp. 155–160, Jan. 1991.

[6] J. Hagenauer, "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Trans. Commun.*, Vol. 36, No. 4, pp. 389–400, April 1988.

[7] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression.* New York: Kluwer Academic. 1992.

[8] T.S. Rappaport, *Wireless Communications - Principles and Practice,* IEEE Press, 1996.

[9] H. Xiao and B. Vucetic, "Soft input source decoding in low-bit-rate speech transmission," *Proc. IEEE Intl. Symp. Inform. Theory*, p. 443, June 1997.