TECHNICAL RESEARCH REPORT

Institute for Systems Research

Quantization Over Discrete Noisy Channels Using Rate-One Convolutional Codes

by N. Phamdo and N. Farvardin

The Institute for Systems Research is supported by the National Science Foundation Engineering Research Center Program (NSFD CD 8803012), Industry and the University

TR 93-35

Quantization Over Discrete Noisy Channels Using Rate-One Convolutional Codes †

Nam Phamdo and Nariman Farvardin

Electrical Engineering Department
and
Institute for Systems Research
University of Maryland
College Park, Maryland 20742

Abstract

We consider high-rate scalar quantization of a memoryless source for transmission over a binary symmetric channel. It is assumed that, due to its suboptimality, the quantizer's output is redundant. Our aim is to make use of this redundancy to combat channel noise. A rate-one convolutional code is introduced to convert this natural redundancy into a usable form. At the receiver, a maximum a posteriori decoder is employed. An upper bound on the average distortion of the proposed system is derived. An approximation of this bound is computable and we search for that convolutional code which minimizes the approximate upper bound. Simulation results for a generalized Gaussian source with parameter $\alpha = 0.5$ at rate 4 bits/sample and channel crossover probability 0.005 show improvements of 11.9 dB in signal-to-noise ratio over the Lloyd-Max quantizer and 4.6 dB over Farvardin and Vaishampayan's channel-optimized scalar quantizer.

Index Terms: Convolutional code; non-equiprobable memoryless data; combined source-channel coding.

---

† This work was supported in part by National Science Foundation grants NSFD MIP-91-09109 and CD-88-03012, and in part by NTT Corporation.

I. Introduction

It is known that the performance of a scalar quantizer can be degraded if it is used over a noisy channel. The performance degradation is more severe for high-rate (i.e., high-resolution) quantizers [1].

Thus far, the most successful approach to solving this problem is to take the effect of channel errors into consideration when designing the quantizer – the so-called combined source-channel coding approach [2, 3]. With this approach, the source distribution and channel transition matrix are assumed to be given. The Lloyd-Max formulation [4, 5] for designing a locally-optimal scalar quantizer for a given rate and source distribution is then modified with the effects of channel errors entering the optimization process. This results in a locally-optimal quantization system [2, 3]. A quantizer so designed will be referred to as a channel-optimized scalar quantizer (COSQ). Compare to the Lloyd-Max quantizer (LMQ), which is designed for a noiseless channel, the COSQ provides a better performance when the channel is noisy.

In this paper, we propose a different approach. It is assumed that the source is memoryless and its distribution is sharp-peaked and broad-tailed, i.e., it is highly non-uniform. Also, it is assumed that a LMQ has been designed for this distribution and its output is represented in binary using a natural binary code (NBC). Due to the non-uniform distribution of the input and the optimal structure of the LMQ, the output distribution is also non-uniform. Therefore, the entropy of the quantizer's output is lower than the rate (log of the number of levels) of the quantizer. The difference between these two quantities is called the quantizer's "residual" redundancy [6].

In previous works [6, 7], it was found that the residual redundancy can be used to combat channel noise — providing that maximum a posteriori (MAP) detection is performed at the decoder. However, the redundancy in [6, 7] is due to the quantizer output memory rather than its non-uniform distribution.

When the channel is symmetric (as is assumed here), redundancy at the channel input in the form of memory can be better utilized at the decoder than redundancy in the form of non-uniform distribution. The information-theoretical argument for this is that, for symmetric channels, a uniform input distribution maximizes the mutual information between channel input and output. This means that in sending redundancy-bearing information over a symmetric channel, the redundancy should be in the form of memory rather than non-uniformity.

Therefore, in this work, we do not transmit the quantizer output directly over the channel, as was done in [6, 7]. Instead, the quantizer's output is first encoded by a rate-one convolutional encoder. A rate-one convolutional encoder is one which produces as many bits as it accepts. The purpose of the convolutional encoder is to match the quantizer's output to the channel, i.e., it converts non-uniformly distributed data into uniformly distributed data.

The output of the convolutional encoder is then transmitted over a binary symmetric channel (BSC). At the receiver, a MAP decoder, implemented using the Viterbi algorithm, is employed. The output of the MAP decoder is then fed to the LMQ decoder. The complete system is depicted in Figure 1.

In this paper, an upper bound on the average distortion of the system in Figure 1 is derived. An approximation of this upper bound is computable and the convolutional encoder is chosen as that which minimizes this approximated upper bound. Simulation results are presented. Two other decoding techniques, MAP symbol-by-symbol decoding and minimum mean squared error decoding, are considered. It is then shown that, using an entropy-constrained formulation, the quantizer output redundancy can be adjusted so that the overall system performance can be improved. The performance of this system under channel mismatch conditions is also studied.

The rest of this paper is organized as follows. In Section II, we present rate-one convolutional codes and derive an approximate upper bound on the average distortion of the proposed system. Simulation results are presented in Section III. Finally, conclusions are given in Section IV.

II. Rate-One Convolutional Codes

In this section, we describe a rate-one convolutional code whose function is to "match" the LMQ output to the BSC. The theoretical justification for choosing a convolutional code for this task can be found in the works by Koshelev [8] and Hellman [9]. Koshelev investigated the computational complexity of sequential decoders of convolutional codes used in joint source-channel coding of discrete non-equiprobable sources [8]. Hellman proved that there exist convolutional codes of rate $R_c < C/H$ "that can be reliably decoded when used for joint source-channel coding" provided that MAP decoding is employed [9].

A. System Description

In the following, we will consider the system depicted in Figure 1. The source, $\{X_n\}$, is assumed to be an i.i.d. process with marginal probability density function (p.d.f.) $f(x)$. The p.d.f. is assumed to be symmetric, i.e., $f(x) = f(-x), \ \forall x$. The source is encoded by an LMQ of rate R bits/sample. The LMQ encoder is described by a set of $M-1$ quantization thresholds, $\{T_1, T_2, \ldots, T_{M-1}\}$, where $M = 2^R$. The output of the LMQ encoder is a sequence of binary R-tuples (R is an integer). Assume that the NBC is used, let $\mathcal{J} \triangleq \{0, 1, \ldots, M-1\}$ and $\gamma : \mathcal{J} \mapsto \{0,1\}^R$ be the binary representation of the integers in $\mathcal{J}$. The encoder is then described by

$$U_n = \gamma(i) \qquad \text{if } X_n \in (T_i, T_{i+1}], \qquad i = 0, 1, \ldots, M-1, \tag{1}$$

where it is assumed that $T_M = -T_0 = \infty$, and $T_i < T_{i+1}$ for $i = 0, 1, \ldots, M-1$. The LMQ is also assumed to be symmetric, hence $T_i = -T_{M-i}$ for $i \in \mathcal{J}$. The output of the LMQ encoder is an index, $U_n \in \{0,1\}^R$, representing the quantization level for X_n. $\{U_n\}$ is passed through a rate-one convolutional encoder which produces $\{W_n\}$ which, in turn, is fed into a BSC, the output of which is denoted by $\{V_n\}$. At the receiver, we make a MAP estimate, $\hat{\mathbf{U}} = \{\hat{U}_n\}_{n=1}^N$, of the sequence $\mathbf{U} = \{U_n\}_{n=1}^N$ based upon the observation $\mathbf{V} = \{V_n\}_{n=1}^N$. Then, $\hat{\mathbf{U}}$ is fed to the LMQ decoder; the output of the decoder is an estimate, $\hat{X}_n$, of X_n given by

$$\hat{X}_n = q_{\gamma^{-1}(\hat{U}_n)}, \tag{2}$$

where q_i is the i^{th} reconstruction level of the quantizer ($i \in \mathcal{J}$). We will use the block notation: $\mathbf{U} = g(\mathbf{X})$, $\mathbf{W} = h(\mathbf{U})$, $\hat{\mathbf{U}} = \eta(\mathbf{V})$ and $\hat{\mathbf{X}} = \delta(\hat{\mathbf{U}})$.

B. Distortion Upper Bound

Traditionally, convolutional codes are designed to minimize probability of error (or, equivalently, maximize free distance). Our aim, however, is to design codes which minimize the squared-error distortion,

$$D = E[\frac{1}{N}\sum_{n=1}^{N}(X_n - \hat{X}_n)^2]. \tag{3}$$

To do this, we first derive an (approximate) upper bound on D and then choose that code which minimizes this bound. Let $\tilde{X}_n \triangleq q_{\gamma^{-1}(U_n)}$, i.e., $\tilde{X}_n$ is the reconstruction sample given that $\hat{U}_n = U_n$. It can be shown that (3) can be expressed as [15]

$$D = E[\frac{1}{N}\sum_{n=1}^{N}(X_n - \tilde{X}_n)^2] + E[\frac{1}{N}\sum_{n=1}^{N}(\tilde{X}_n - \hat{X}_n)^2]. \tag{4}$$

The first term in (4), denoted by D_s, is the average distortion induced by the source coding (quantization) operation while the second term, denoted by D_c, is the distortion introduced by channel noise. Our goal now is to minimize D_c.

Let $\mathcal{Q} \triangleq \{q_0, q_1, \ldots, q_{M-1}\}$ be the set of reconstruction levels. Define

$$p_i \triangleq \Pr\{\tilde{X}_n = q_i\} = \Pr\{X_n \in (T_i, T_{i+1}]\}, \quad i \in \mathcal{J}. \tag{5}$$

By symmetry, we have $q_i = -q_{M-1-i}$ and $p_i = p_{M-1-i}$ for $i \in \mathcal{J}$. Since the source distribution is assumed to be sharp-peaked, reconstruction levels closer to the origin are more probable than those on the outskirts. We thus assume that for $i = 0, 1, \ldots, M/2 - 2$, $p_i < p_{i+1}$.

Let $\tilde{\mathbf{X}} = (\tilde{X}_1, \tilde{X}_2, \ldots, \tilde{X}_N)$. We are interested in upper bounds for D_c given the "type" of $\tilde{\mathbf{X}}$ and the number of bit errors.

Suppose $\tilde{\mathbf{X}} \in \{q_{M/2-1}, q_{M/2}\}^N \triangleq \mathcal{T}_0$, i.e., each element of $\tilde{\mathbf{X}}$ is either $q_{M/2-1}$ or $q_{M/2}$. Such a sequence is called a *type-0* sequence and

$$\Pr\{\tilde{\mathbf{X}} \in \mathcal{T}_0\} = 2^N p_{M/2-1}^N \triangleq Q_0. \tag{6}$$

Type-0 sequences are the most probable. If $\mathbf{z} \in \mathcal{T}_0$, then $\Pr\{\tilde{\mathbf{X}} = \mathbf{z}\} = p_{M/2-1}^N \triangleq P_0$. Now define a *type-1* sequence as one in which exactly $N-1$ elements are either $q_{M/2-1}$ or $q_{M/2}$ and the other element is either $q_{M/2-2}$ or $q_{M/2+1}$. The set of all type-1 sequences is denoted by $\mathcal{T}_1$ and

$$\Pr\{\tilde{\mathbf{X}} \in \mathcal{T}_1\} = N 2^N p_{M/2-1}^{N-1} p_{M/2-2} \triangleq Q_1. \tag{7}$$

Type-1 sequences are the second most probable. If $\mathbf{z} \in \mathcal{T}_1$, then $\Pr\{\tilde{\mathbf{X}} = \mathbf{z}\} = p_{M/2-1}^{N-1} p_{M/2-2} \triangleq P_1$. Continuing this way, we get a sequence of types $\{\mathcal{T}_i\}_{i=0}^L$ and two sequences of probabilities, $\{Q_i\}_{i=0}^L$ and $\{P_i\}_{i=0}^L$, where $\cup_{i=0}^L \mathcal{T}_i = \mathcal{Q}^N$, $\sum_{i=0}^L Q_i = 1$ and $P_i \geq P_{i+1}$ for $i = 0, 1, \ldots, L-1$.

Let $\mathbf{W}$ and $\mathbf{V}$ be the transmitted and received bit sequences, respectively, and let $d_H(\mathbf{W}, \mathbf{V})$ be the number of errors occurred during transmission. Define

$$D_{i,j} \triangleq E\left[\frac{1}{N}||\tilde{\mathbf{X}} - \hat{\mathbf{X}}||^2 \,\middle|\, \tilde{\mathbf{X}} \in \mathcal{T}_i, d_H(\mathbf{W}, \mathbf{V}) = j\right], \tag{8}$$

$i = 0, 1, \ldots, L$, $j = 0, 1, \ldots, RN$. D_c can be computed according to

$$D_c = \sum_{i=0}^{L} \sum_{j=0}^{RN} D_{i,j} \Pr\{\tilde{\mathbf{X}} \in \mathcal{T}_i, d_H(\mathbf{W}, \mathbf{V}) = j\}, \tag{9}$$

where

$$\Pr\{\tilde{\mathbf{X}} \in \mathcal{T}_i, d_H(\mathbf{W}, \mathbf{V}) = j\} = Q_i \binom{RN}{j} \epsilon^j (1-\epsilon)^{RN-j}, \tag{10}$$

and ϵ is the channel bit error rate (BER).

Define the *a posterior probability* (APP) of a sequence $\mathbf{u} \in \{0,1\}^{RN}$ given that a sequence $\mathbf{v}$ is received as

$$APP(\mathbf{u}|\mathbf{v}) \triangleq \Pr\{\mathbf{U} = \mathbf{u}|\mathbf{V} = \mathbf{v}\}. \tag{11}$$

The output of the MAP detector is given by

$$\hat{\mathbf{u}} = \eta(\mathbf{v}) = \arg \max_{\mathbf{u} \in \{0,1\}^{RN}} APP(\mathbf{u}|\mathbf{v}). \tag{12}$$

Then, $D_{i,j}$ can be upper bounded by

$$D_{i,j} \leq D_{i,j}^u \triangleq \max_{\substack{\tilde{\mathbf{x}} \in \mathcal{T}_i \\ \mathbf{v} : d_H(h(g(\tilde{\mathbf{x}})), \mathbf{v}) = j}} \frac{1}{N} ||\tilde{\mathbf{x}} - \delta(\eta(\mathbf{v}))||^2. \tag{13}$$

If the above upper bound can be computed, an upper bound on D_c can be computed by

$$D_c^u \triangleq \sum_{i=0}^{L} \sum_{j=0}^{RN} D_{i,j}^u Q_i \begin{pmatrix} RN \\ j \end{pmatrix} \epsilon^j (1-\epsilon)^{RN-j}, \tag{14}$$

which leads to an upper bound on D given by $D^u \triangleq D_s + D_c^u$. Note that the maximization in (13) can be written as

$$D_{i,j}^u = \max_{\tilde{\mathbf{x}} \in \mathcal{T}_i} D_j^u(\tilde{\mathbf{x}}), \tag{15}$$

where

$$D_j^u(\tilde{\mathbf{x}}) = \max_{\mathbf{v} : d_H(h(g(\tilde{\mathbf{x}})), \mathbf{v}) = j} \frac{1}{N} ||\tilde{\mathbf{x}} - \delta(\eta(\mathbf{v}))||^2. \tag{16}$$

The following theorem states that when $i = 0$ the maximization in (15) is unnecessary.

Theorem 1 *$D_j^u(\tilde{\mathbf{x}})$ is constant on $\mathcal{T}_0$, i.e.,*

$$D_j^u(\tilde{\mathbf{x}}_1) = D_j^u(\tilde{\mathbf{x}}_2) \quad \forall \tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2 \in \mathcal{T}_0, \ \forall j = 0, 1, \ldots, RN. \tag{17}$$

The proof of the theorem is provided in [15]. In the following, we provide an algorithmic approach for computing an upper bound on $D_j^u(\tilde{\mathbf{x}})$ for a given $\tilde{\mathbf{x}}$.

Algorithm:

(1) Set $D_l^*(\tilde{\mathbf{x}}) = 0$ for $l = 0, 1, \ldots, RN$. Set $l = 0$.

(2) $l = l + 1$.

(3) Let $\mathbf{u}$ be the binary representation of $\tilde{\mathbf{x}}$, define

$$\mathcal{Z}^l \triangleq \left\{ \mathbf{z} \in \{0,1\}^{RN} : d_H(h(\mathbf{u}), h(\mathbf{z})) = l, APP(\mathbf{z}|h(\mathbf{z})) \geq APP(\mathbf{u}|h(\mathbf{z})) \right\}, \tag{18}$$

and set

$$D_l^*(\tilde{\mathbf{x}}) = \max_{\mathbf{z} \in \mathcal{Z}^l} \frac{1}{N} ||\tilde{\mathbf{x}} - \delta(\mathbf{z})||^2. \tag{19}$$

(4) $\forall \mathbf{z} \in \mathcal{Z}^l$, let j^* be the minimum j for which

$$\Pr\{\mathbf{U} = \mathbf{z}\} \epsilon^{l-j} (1-\epsilon)^{RN-(l-j)} \geq \Pr\{\mathbf{U} = \mathbf{u}\} \epsilon^j (1-\epsilon)^{RN-j}, \tag{20}$$

and set

$$D_{j^*}^*(\tilde{\mathbf{x}}) = \max(D_{j^*}^*(\tilde{\mathbf{x}}), \frac{1}{N} ||\tilde{\mathbf{x}} - \delta(\mathbf{z})||^2). \tag{21}$$

(5) If $l < RN$, go to step **(2)**.

(6) For $l = 0, 1, \ldots, RN$, let $D_l^*(\tilde{\mathbf{x}}) = \max(D_0^*(\tilde{\mathbf{x}}), D_1^*(\tilde{\mathbf{x}}), \ldots, D_l^*(\tilde{\mathbf{x}}))$.

(7) Stop.

Note that in step **(3)** we assume that if we receive $h(\mathbf{z})$ and $APP(\mathbf{z}|h(\mathbf{z})) \geq APP(\mathbf{u}|h(\mathbf{z}))$ then we decode $\mathbf{z}$. In step **(4)**, we note that only j^* errors need to occur for $APP(\mathbf{z}|h(\mathbf{z})) \geq APP(\mathbf{u}|h(\mathbf{z}))$. Finally, in step **(6)**, we force $D_j^*(\tilde{\mathbf{x}})$ to be monotonically non-decreasing. The following theorem establishes that $D_j^*(\tilde{\mathbf{x}})$ upper bounds $D_j^u(\tilde{\mathbf{x}})$.

Theorem 2 *$D_j^*(\tilde{\mathbf{x}}) \geq D_j^u(\tilde{\mathbf{x}})$ for $j = 0, 1, \ldots, RN$.*

The proof of this theorem is also given in [15]. The above provides a yet looser upper bound on the conditional average distortion. This new upper bound is computable. $D_j^*(\tilde{\mathbf{x}})$ in step **(3)** can be computed using a small modification of the algorithm described in [10, 11].

C. Approximations

Even though $D_j^*(\tilde{\mathbf{x}})$ is not constant on $\mathcal{T}_i$ for $i > 0$, we have observed that, in most cases, it does not vary much on $\mathcal{T}_i$. We thus approximate the maximum of $D_j^*(\tilde{\mathbf{x}})$ over the set $\mathcal{T}_i$ by $D_j^*(\tilde{\mathbf{x}})$ for some randomly chosen $\tilde{\mathbf{x}} \in \mathcal{T}_i$. Also, we note that $D_{i,j}$ is bounded (by the maximum distance between any two elements in $\mathcal{Q}$) while $\Pr\{\tilde{\mathbf{X}} \in \mathcal{T}_i, d_H(\mathbf{W}, \mathbf{V}) = j\}$ vanishes for large i and j. Therefore, we approximate the upper bound on D_c (see (14)) by

$$D_c^* = \sum_{i=0}^{i_{max}} \sum_{j=0}^{j_{max}} D_{i,j}^* Q_i \binom{RN}{j} \epsilon^j (1-\epsilon)^{RN-j}, \tag{22}$$

where $D_{i,j}^* = D_j^*(\tilde{\mathbf{x}})$ for some randomly chosen $\tilde{\mathbf{x}} \in \mathcal{T}_i$ and $i_{max} < L$, $j_{max} < RN$ are chosen such that $\Pr\{\tilde{\mathbf{X}} \in \cup_{i=i_{max}+1}^{L} \mathcal{T}_i\}$ and $\Pr\{d_H(\mathbf{W}, \mathbf{V}) > j_{max}\}$ are sufficiently small.

D. Search for Good Codes

The above development provides an (approximate) upper bound on the average distortion for a given convolutional encoder. To find that encoder which minimizes the (approximate) upper bound, the following procedure is adopted. First, an encoder is chosen at random by randomly selecting the tap coefficients. The upper bound for this code is computed. A new encoder is chosen by randomly changing one of the tap coefficients. If this results in a smaller upper bound, it is recorded as "the best code found thus far". The encoder is again changed and the procedure is repeated. We stop the algorithm if no improvement is found after T changes.

A list of good codes found by the above procedure is listed in Table 1. Here, we consider two types of encoders: minimal encoders and encoders with feedback. For minimal encoders, the tap coefficients are listed where we assume that the least significant bits are fed into the encoder first. In the case of feedback, we used the structure depicted in [12], where all additions and multiplications are over $GF(2^R)$. We used the primitive polynomials $p(X) = 1 + X + X^3$ and $p(X) = 1 + X + X^4$ to construct $GF(2^3)$ and $GF(2^4)$, respectively [13]. The tap coefficients of the forward loop are given in the first row while the tap coefficients of the feedback loop are given in the second row. For both minimal and feedback encoders, the coefficients are given in octal for $R = 3$ and hexadecimal for $R = 4$.

IV. Simulation Results

Simulations of the system in Figure 1 for a Generalized Gaussian source with $\alpha = 0.5$ have been performed and the results are given in Table 2. Performances of LMQ with NBC and COSQ and the optimum performance theoretically attainable (OPTA) are also given in Table 2. Note that in almost all cases the convolutionally encoded system beats the COSQ.

A. Other Decoding Techniques

We note that all of the delay of the system in Figure 1 is in the MAP decoder. The MAP decoder must wait for the entire sequence to arrive before it makes a decision on which sequence was transmitted. In this subsection, we consider two other decoding methods which have smaller delays than the MAP Viterbi decoder.

We first consider optimum symbol-by-symbol decoding for a given delay τ. Suppose we are interested in decoding u_n given that we have observed $\mathbf{v}_1^{n+\tau} = (v_1, v_2, \ldots, v_{n+\tau})$. The minimum probability of error decoding rule is given by:

$$\hat{u}_n = \arg \max_{u_n} \Pr\{U_n = u_n | \mathbf{V}_1^{n+\tau} = \mathbf{v}_1^{n+\tau}\}. \tag{23}$$

Note that the above minimizes the symbol error probability whereas the MAP Viterbi decoder minimizes the sequence (or block) error probability. Implementation of this decoder is described in [15].

Note that similar to hard-decision convolutional decoding, there is an irreversible loss of information between the MAP decoder and the LMQ decoder. To fix this problem, we combine these two decoders into a single minimum mean-squared error (MMSE) decoder. The MMSE estimate of $\mathbf{X}_n$ given that $\mathbf{V}_1^{n+\tau} = \mathbf{v}_1^{n+\tau}$ is

$$\hat{\mathbf{x}}_n = E[\mathbf{X}_n | \mathbf{V}_1^{n+\tau} = \mathbf{v}_1^{n+\tau}], \tag{24}$$

which can be expanded to

$$\hat{\mathbf{x}}_n = \sum_{u_n} E[\mathbf{X}_n | U_n = u_n, \mathbf{V}_1^{n+\tau} = \mathbf{v}_1^{n+\tau}] \Pr\{U_n = u_n | \mathbf{V}_1^{n+\tau} = \mathbf{v}_1^{n+\tau}\}. \tag{25}$$

Since the source is memoryless, the above expectation is just $q_{\gamma^{-1}(u_n)}$. The second factor in the above summation can be obtained from the symbol-by-symbol decoder. Simulation results using symbol-by-symbol and MMSE decoding with $\tau = 20$ samples are given in Table 3. In all cases, MMSE decoding is superior to symbol-by-symbol decoding. In Figure 2, we plot the SNR performance as a function of the delay for the case $\alpha = 0.5, R = 3, \epsilon = 0.005$ and constraint length $K = 2$.

B. Entropy-Constrained Scalar Quantizer

We have repeated the simulations for a Generalized Gaussian source with $\alpha = 1.0$, i.e., a Laplacian source. The results are given in Table 4. The codes that have been found for $\alpha = 1.0$ are given in [15]. We note that the results of the convolutionally encoded system for $\alpha = 1.0$ are not as good as those for $\alpha = 0.5$. In fact, for many cases, the SNR performances of the convolutionally encoded system are even worse than that of LMQ-NBC.

The poor performance of the convolutionally encoded system for the Laplacian source ($\alpha = 1.0$) can be attributed to the low level of redundancy at the output of the LMQ. It is possible to increase this redundancy by changing the quantization levels of the quantizer. This may reduce the SNR performance when there is no channel noise. However, the increase in redundancy will lead to additional protection when there is noise.

Here, instead of LMQ, we consider rate- and entropy-constrained scalar quantizers. These are quantizers in which both the number of levels and the entropy are constrained. We used the generalized Lloyd algorithm described in [14] to design these quantizers. The redundancy of a quantizer is defined to be the difference between its rate and its entropy. In Figure 3, we plot the SNR performances of the convolutionally encoded system for $\alpha = 1.0$ using rate- and entropy-constrained scalar quantizers and MAP Viterbi decoding as a function of the quantizer redundancy. Here, we used the feedback codes listed in Table 1. Table 5 lists the best results using rate- and entropy-constrained scalar quantizers.

C. Channel Mismatch

Up to this point, we have been working under the assumption that the channel BER is known. In most practical situations, the BER is not known *a priori* or it is time-varying. In this subsection, we will examine the performance of the convolutionally encoded system under channel mismatch, i.e., when the system is designed for a BER ϵ_d, but applied to a channel whose BER is actually ϵ_a. We consider the case $\alpha = 0.5, R = 4, K = 2$. The simulation results for this case are plotted in Figure 4, where we have chosen the entropy-constrained scalar quantizers which yielded the peak performances reported in Table 5.

V. Conclusions

We have proposed the use of a rate-one convolutional code to exploit the residual redundancy of the LMQ. We derived an approximate upper bound on the average distortion of the proposed system and searched for those codes which minimize the approximate upper bound. Even though its performances are still far from OPTA, the proposed system provides significant improvements in SNR over LMQ-NBC and COSQ.

References

[1] N. S. Jayant and P. Noll, *Digital Coding of Waveforms.* Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1984.

[2] A. Kurtenbach and P. Wintz, "Quantizing for Noisy Channels," *IEEE Trans. Commun. Technology,* Vol. 17, pp. 291-302, Apr. 1969.

[3] N. Farvardin and V. Vaishampayan, "Optimal Quantizer Design for Noisy Channels: An Approach to Combined Source-Channel Coding," *IEEE Trans. Inform. Theory,* Vol. 33, pp. 827-838, Nov. 1987.

[4] S. P. Lloyd, "Least Squares Quantization in PCM," *IEEE Trans. Inform. Theory,* Vol. 28, pp. 129-137, Mar. 1982.

[5] J. Max, "Sufficient Conditions for Achieving Minimum Distortion in a Quantizer," *IEEE Int. Conv. Rec.,* Part 1, pp. 101-111, 1967.

[6] K. Sayood and J. C. Borkenhagen, "Use of Residual Redundancy in the Design of Joint Source/Channel Coders," *IEEE Trans. Commun.*, Vol. 39, pp. 838-846, Jun. 1991.

[7] N. Phamdo and N. Farvardin, "Optimal Detection of Discrete Markov Sources Over Discrete Memoryless Channels — Applications to Combined Source-Channel Coding," *IEEE Trans. Inform. Theory,* accepted for publication, Dec. 1992.

[8] V. N. Koshelev, "Direct Sequential Encoding and Decoding for Discrete Sources," *IEEE Trans. Inform. Theory,* Vol. 19, pp. 340-343 May 1973.

[9] M. E. Hellman, "Convolutional Source Coding," *IEEE Trans. Inform. Theory,* Vol. 21, pp. 651-656 Nov. 1975.

[10] L. R. Bahl, C. D. Cullum, W. D. Frazer and F. Jelinek, "An Efficient Algorithm for Computing Free Distance," *IEEE Trans. Inform. Theory,* Vol. 18, pp. 437-439 May 1972.

[11] K. J. Larsen, "Comments on 'An Efficient Algorithm for Computing Free Distance'," *IEEE Trans. Inform. Theory,* Vol. 19, pp. 577-579 July 1973.

[12] G. D. Forney, Jr., "Convolutional Codes I: Algebraic Structure," *IEEE Trans. Inform. Theory,* Vol. 16, pp. 720-738 Nov. 1970.

[13] S. Lin and D. J. Costello, Jr., *Error Control Coding.* Englewood Cliffs, New Jersey: Prentice-Hall, 1983.

[14] N. Farvardin and J. Modestino, "Optimum Quantizer Performance for a Class of Non-Gaussian Memoryless Sources," *IEEE Trans. Inform. Theory,* Vol. 30, pp. 485-497, Jan. 1984.

[15] N. Phamdo, "Quantization Over Discrete Noisy Channels Under Complexity Constraints," Ph.D. Dissertation, Electrical Engineering Department, University of Maryland, College Park, in preparation.

		Minimal		Feedback	
R	ϵ	$K=1$	$K=2$	$K=1$	$K=2$
3	0.005	4 2 2 1 1 5	4 4 1 2 2 2 1 3 5	4 7 3	7 6 2 1 6
	0.010	4 1 2 4 1 3	4 6 5 2 0 4 1 3 3	4 7 3	7 6 2 1 6
4	0.005	8 1 4 F 2 3 1 7	8 F 8 4 4 D 2 F 2 1 3 E	A D 2	A 8 8 8 7
	0.010	8 5 4 3 2 B 1 2	8 8 2 4 A 1 2 5 D 1 B 8	A D 2	A 8 8 8 7

Table 1: A List of Good Rate-One Convolutional Codes for the Generalized Gaussian Source with $\alpha = 0.5$; K = Constraint Length; 2^{RK} = Number of States.

R	ϵ	LMQ-NBC [3]	COSQ [3]	Convolutional Encoder Minimal K=1	Minimal K=2	Feedback K=1	Feedback K=2	OPTA
3	0.000	10.35	10.35	10.35	10.35	10.35	10.35	21.74
	0.005	5.19	8.35	8.94	9.60	9.12	10.00	20.97
	0.010	2.90	7.23	7.75	8.44	7.21	8.71	20.34
4	0.000	15.69	15.69	15.69	15.69	15.69	15.69	27.79
	0.005	3.49	10.79	11.16	14.56	12.96	15.36	26.95
	0.010	0.64	9.09	9.48	11.60	9.40	12.46	26.05

Table 2: SNR (in dB) Performances of LMQ-NBC, COSQ and Convolutionally Encoded Systems with MAP Viterbi Decoding; Generalized Gaussian Source with $\alpha = 0.5$; K = Constraint Length; 2^{KR} = Number of States.

		MAP Viterbi Delay=∞ Feedback		Sym-By-Sym Delay=20 Feedback		MMSE Delay=20 Feedback	
R	ϵ	$K=1$	$K=2$	$K=1$	$K=2$	$K=1$	$K=2$
3	0.005	9.12	10.00	8.96	9.90	9.33	10.10
	0.010	7.21	8.71	7.59	8.59	8.22	9.03
4	0.005	12.96	15.36	12.37	15.22	13.23	15.33
	0.010	9.40	12.46	9.17	11.85	10.37	12.75

Table 3: SNR (in dB) Performances of Convolutionally Encoded Systems Using Three Different Decoding Techniques; Generalized Gaussian Source with $\alpha = 0.5$; K = Constraint Length; 2^{KR} = Number of States.

R	ϵ	LMQ-NBC [3]	COSQ [3]	Convolutional Encoder Minimal K=1	Minimal K=2	Feedback K=1	Feedback K=2	OPTA
3	0.000	12.64	12.64	12.64	12.64	12.64	12.64	18.68
	0.005	9.30	10.49	8.64	9.50	8.11	9.05	17.87
	0.010	7.44	9.17	6.16	6.16	5.56	5.34	17.23
4	0.000	18.13	18.13	18.13	18.13	18.13	18.13	24.70
	0.005	9.78	12.76	9.64	10.72	8.57	11.02	23.65
	0.010	7.11	11.03	5.59	5.04	5.26	5.49	22.77

Table 4: SNR (in dB) Performances of LMQ-NBC, COSQ and Convolutionally Encoded Systems with MAP Viterbi Decoding; Generalized Gaussian Source with $\alpha = 1.0$; K = Constraint Length; 2^{KR} = Number of States.

α	R	ϵ	LMQ NBC [3]	COSQ [3]	Feedback K=2	OPTA
0.5	3	0.005	5.19	8.35	10.08	20.97
		0.010	2.90	7.23	9.37	20.34
	4	0.005	3.49	10.79	15.39	26.95
		0.010	0.64	9.09	13.91	26.05
1.0	3	0.005	9.30	10.49	10.72	17.87
		0.010	7.44	9.17	9.45	17.23
	4	0.005	9.78	12.76	15.76	23.65
		0.010	7.11	11.03	13.58	22.77

Table 5: SNR (in dB) Performances of LMQ-NBC, COSQ and Convolutionally Encoded Systems with MAP Viterbi Decoding and Entropy-Constrained Scalar Quantizers; K = Constraint Length; 2^{KR} = Number of States.

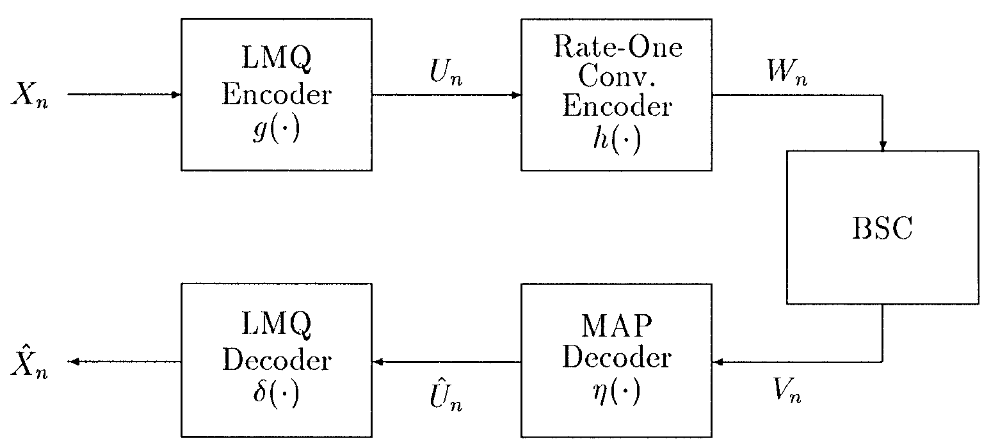

Figure 1: Block Diagram of Convolutionally Encoded System.

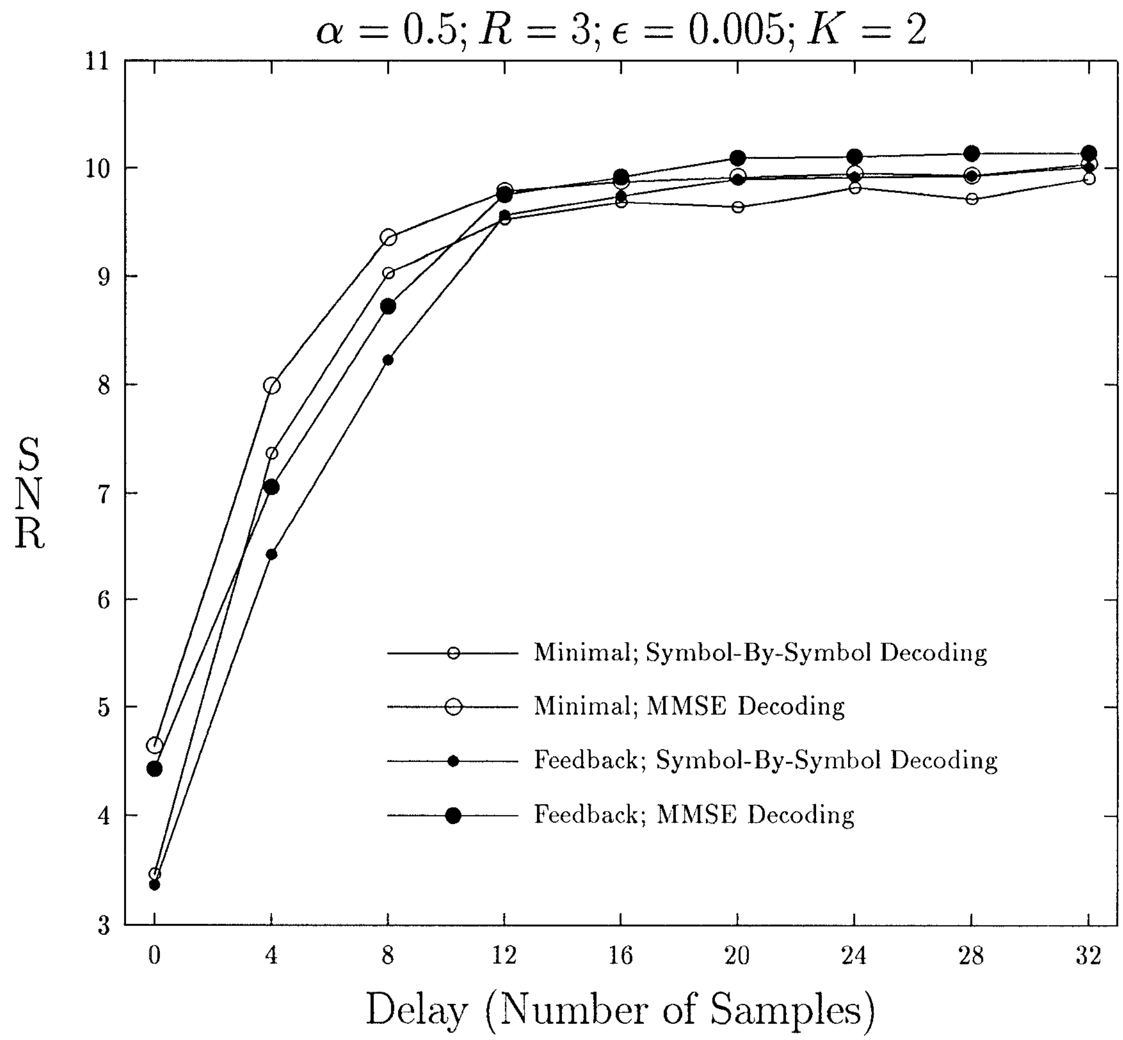

Figure 2: SNR (in dB) Performances of Convolutionally Encoded Systems with Symbol-By-Symbol and MMSE Decoding Vs. Delay; Generalized Gaussian Source with $\alpha = 0.5$; K = Constraint Length; 2^{KR} = Number of States.

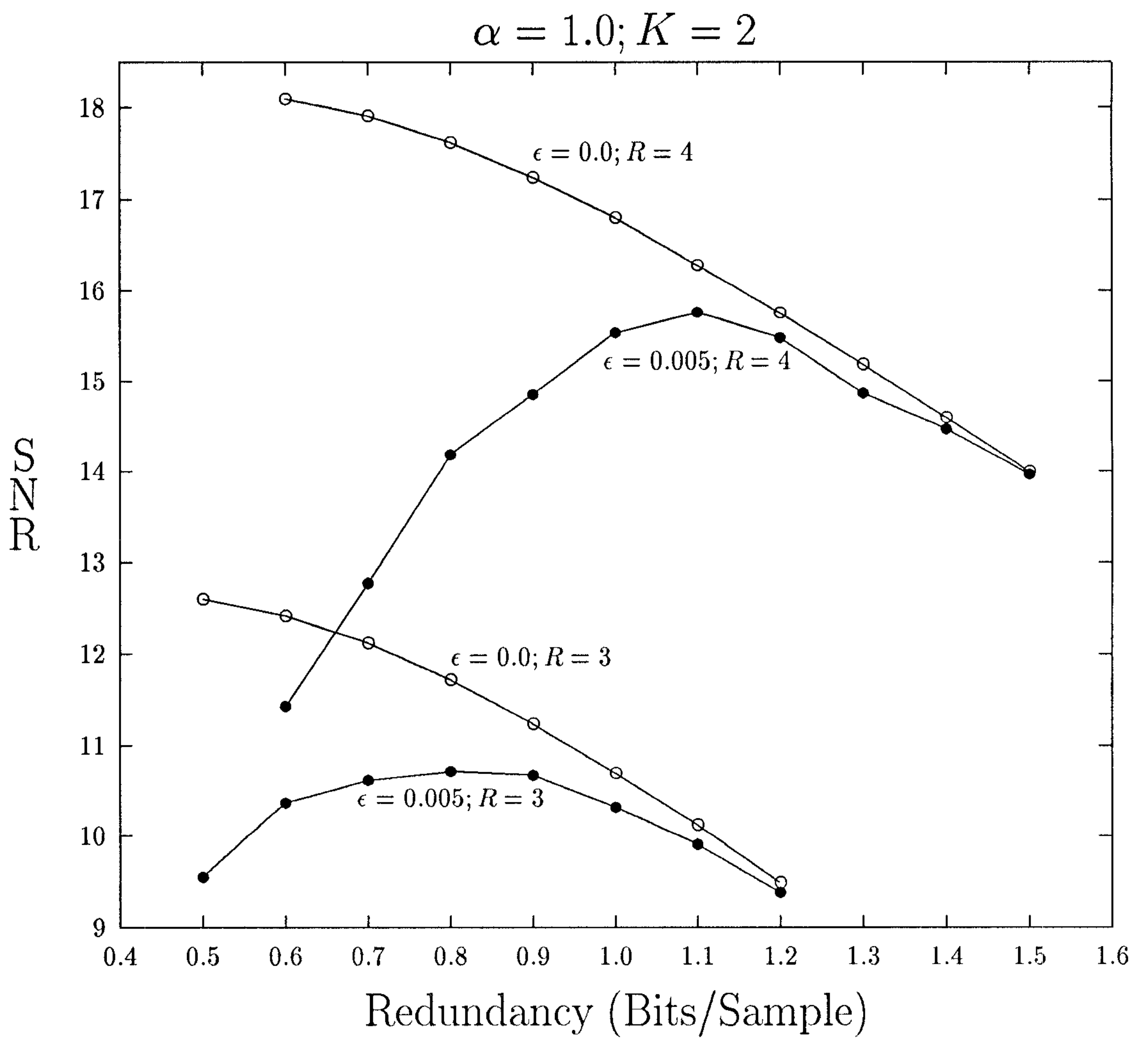

Figure 3: SNR (in dB) Performances of Convolutionally Encoded Systems with MAP Viterbi Decoding Vs. Redundancy of Entropy-Constrained Scalar Quantizer; Generalized Gaussian Source with $\alpha = 1.0$; K = Constraint Length; 2^{KR} = Number of States.

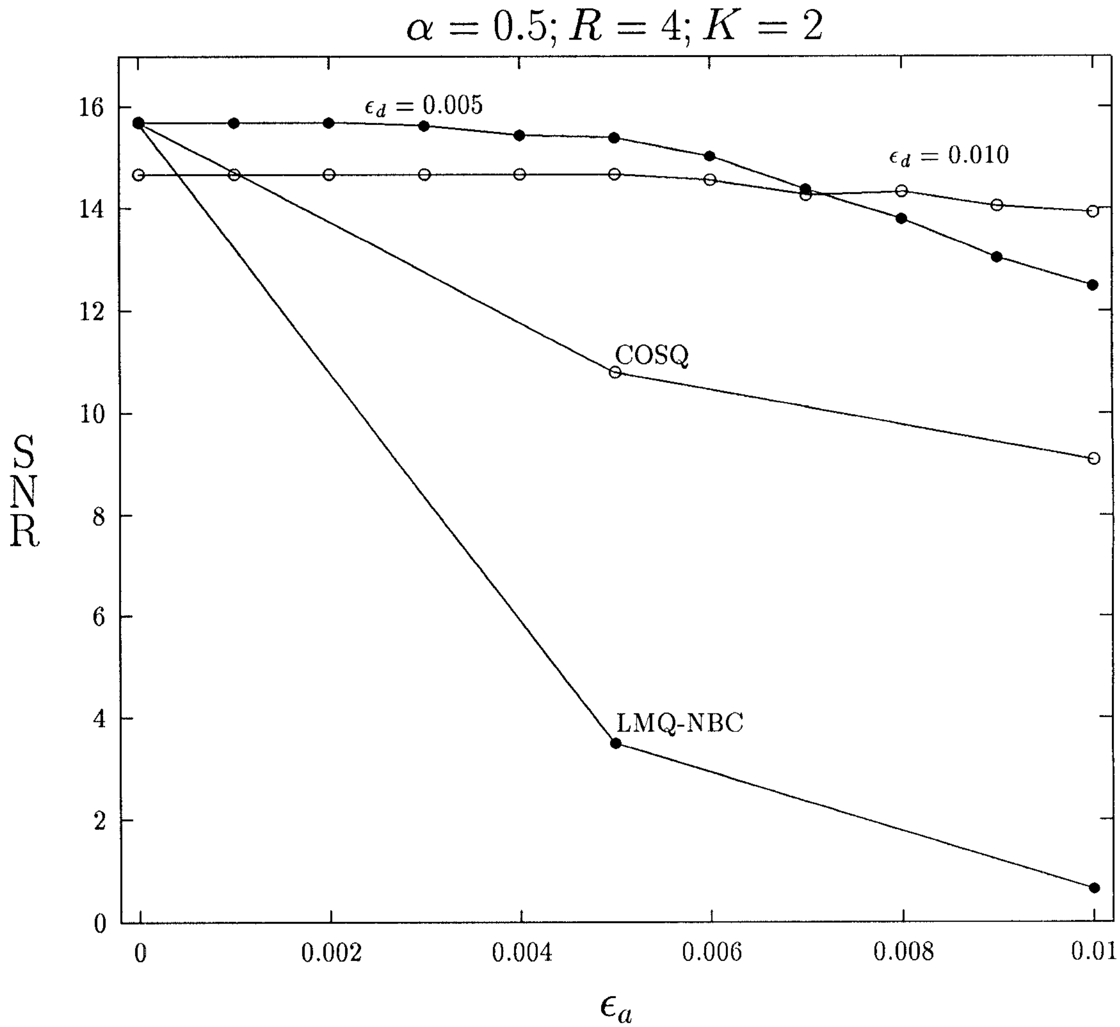

Figure 4: SNR (in dB) Performances of Convolutionally Encoded Systems with MAP Viterbi Decoding Under Channel Mismatch; Generalized Gaussian Source with $\alpha = 0.5$; ϵ_d = Design BER; ϵ_a = Actual Channel BER; K = Constraint Length; 2^{KR} = Number of States.